Microsoft Security

SC-300

Tag 3

Guten Morgen!

Microsoft Identity and Access Administrator

SC-300 Agenda

LP1: Implement an Identity Management Solution

LP2: Implement an Authentication and Access Management Solution

LP3: Implement Access Management for Apps

MDCA
App Registration
Enterprise App
Service Principal

LP4: Plan and Implement an Identity Governance Strategy

Microsoft Security

Implement Access Management for Apps

Outline

On Prem GCP AWS Azure

VM

Multi Cloud

ARC Server
ARC K8S
ARC SQL

Sub
RG
R
Pol

Plan and design the integration of Enterprise Apps for Single Sign-On (SSO)

Implement, and monitor the integration of Enterprise Apps

Implement app registrations

Labs

Pizza

IaaS	PaaS	SaaS
Customer	—	—
Azure		

VM vnet → App service

App Registration

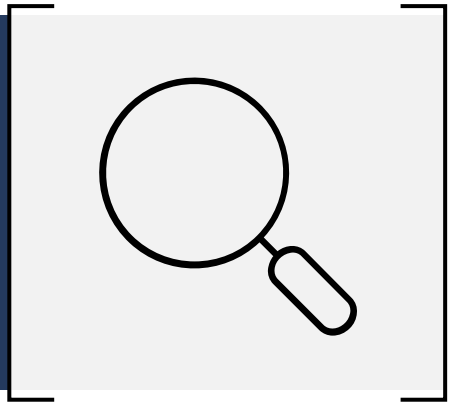
Microsoft Security

Plan and Design the Integration of Enterprise Apps for SSO

Objectives

- Discover apps by using MDCA or ADFS app report
- Configure app connectors in MDCA
- Design and implement access management for apps
- Design and implement app management roles
- Configure pre-integrated (gallery) SaaS apps
- Implement and manage policies for OAuth apps (in MDCA)

Discover apps by using MDCA or ADFS app report



What is CASB and Microsoft Defender for Cloud Apps (MDCA)

CASB – Cloud Access Security Broker

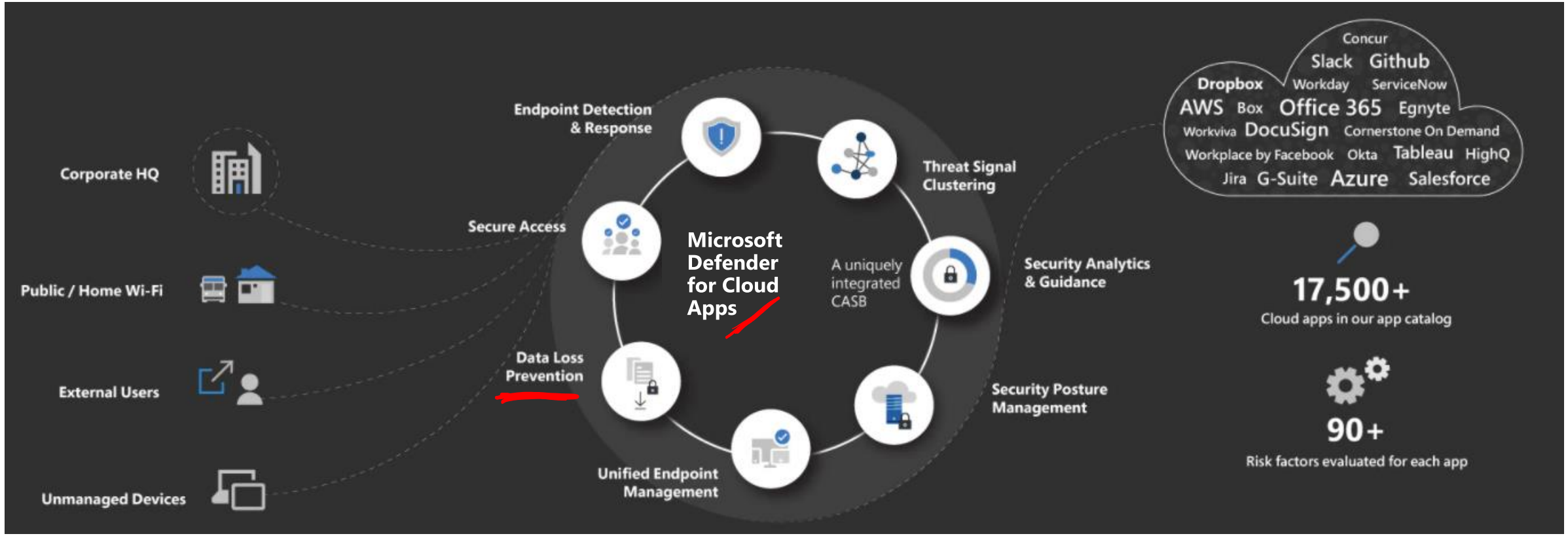
A security tool placed between a cloud service (like an app) and the user to interject enterprise security policies before the cloud-based resource is accessed.

MDCA – Microsoft Defender for Cloud Apps (formerly Cloud App Security)

Microsoft implementation of a CASB service to protect data, services, and applications with enterprise policies. It provides supplemental reporting and analytics services

Microsoft Defender for Cloud Apps - Process Flow

SIEM
SOAR > Sentinel



Microsoft Defender for Cloud Apps Architecture

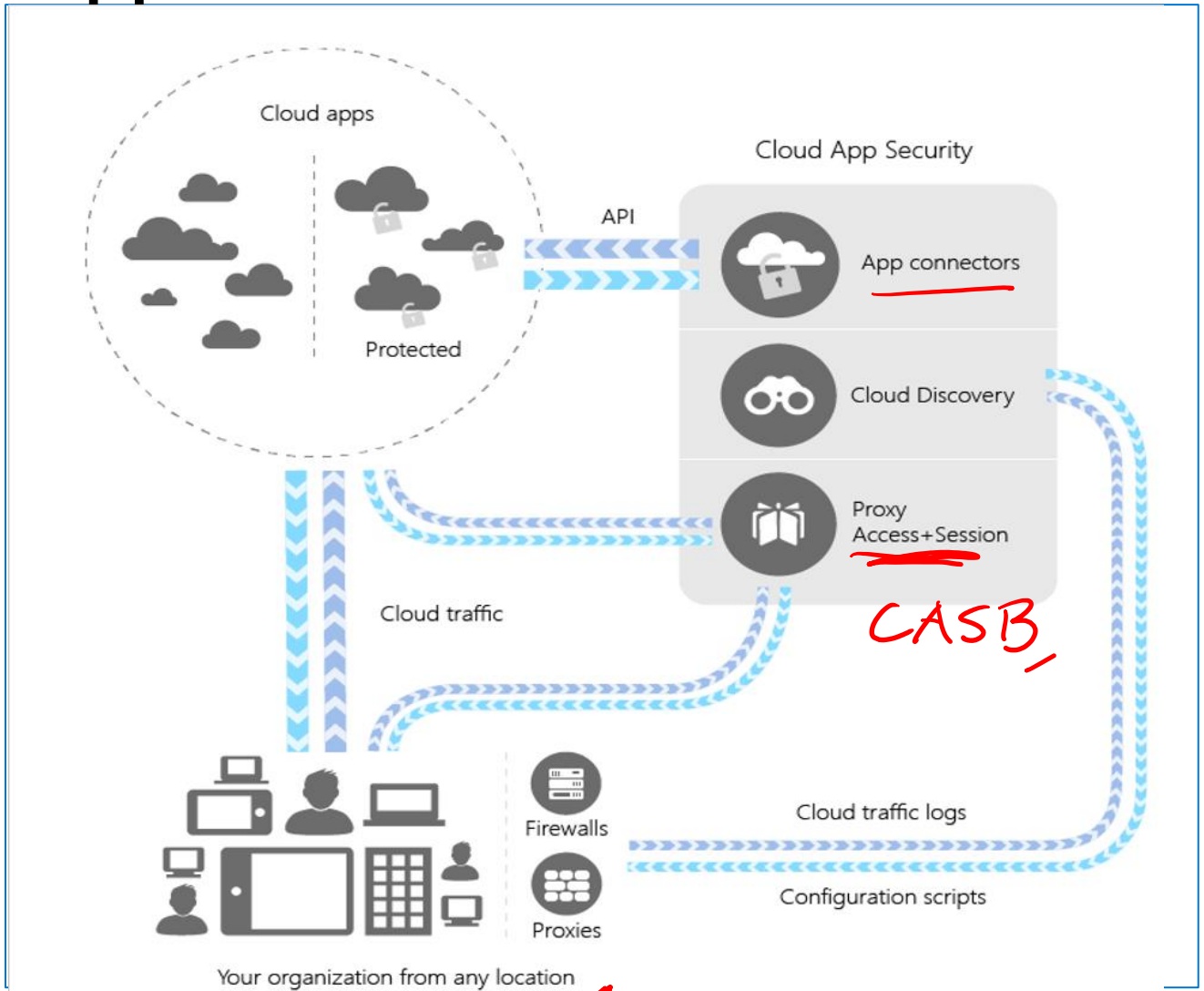
Cloud Discovery – Find apps

Sanctioning – Allow / Deny apps

Connectors – extend protection into the app with APIs

Conditional Access – set access requirements

Policy Control – define user behavior with apps



MDCA Capabilities

Shadow IT Discovery – find an manage cloud apps

Information Protection – protect information as it travels

Threat Protection – look for unusual behavior

Compliance Assessment – assess against regulatory requirements

Microsoft Defender for Cloud Apps

Cloud Access Security Broker (CASB)

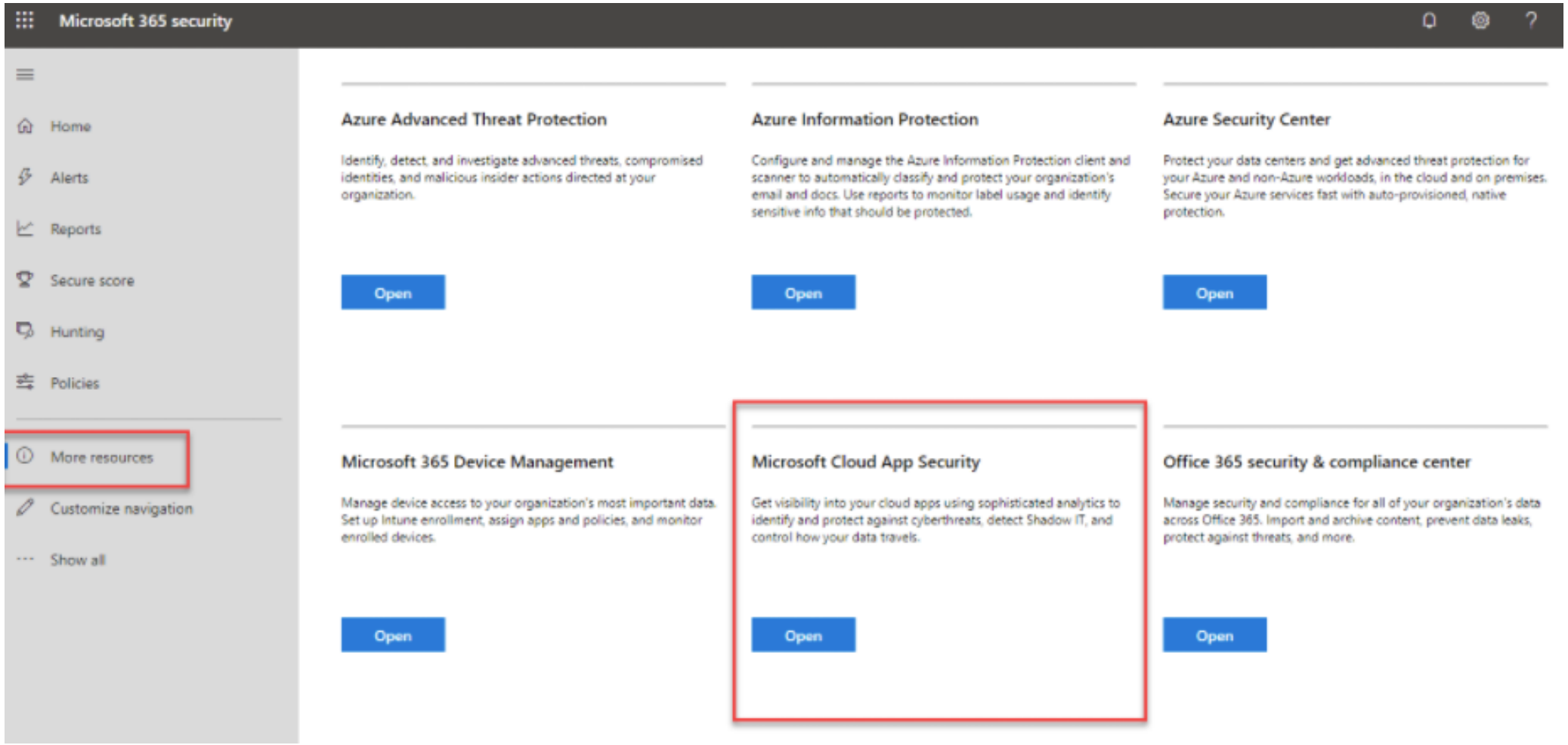
Several different deployment modes:

- Log collection
- API connectors
- Reverse proxy

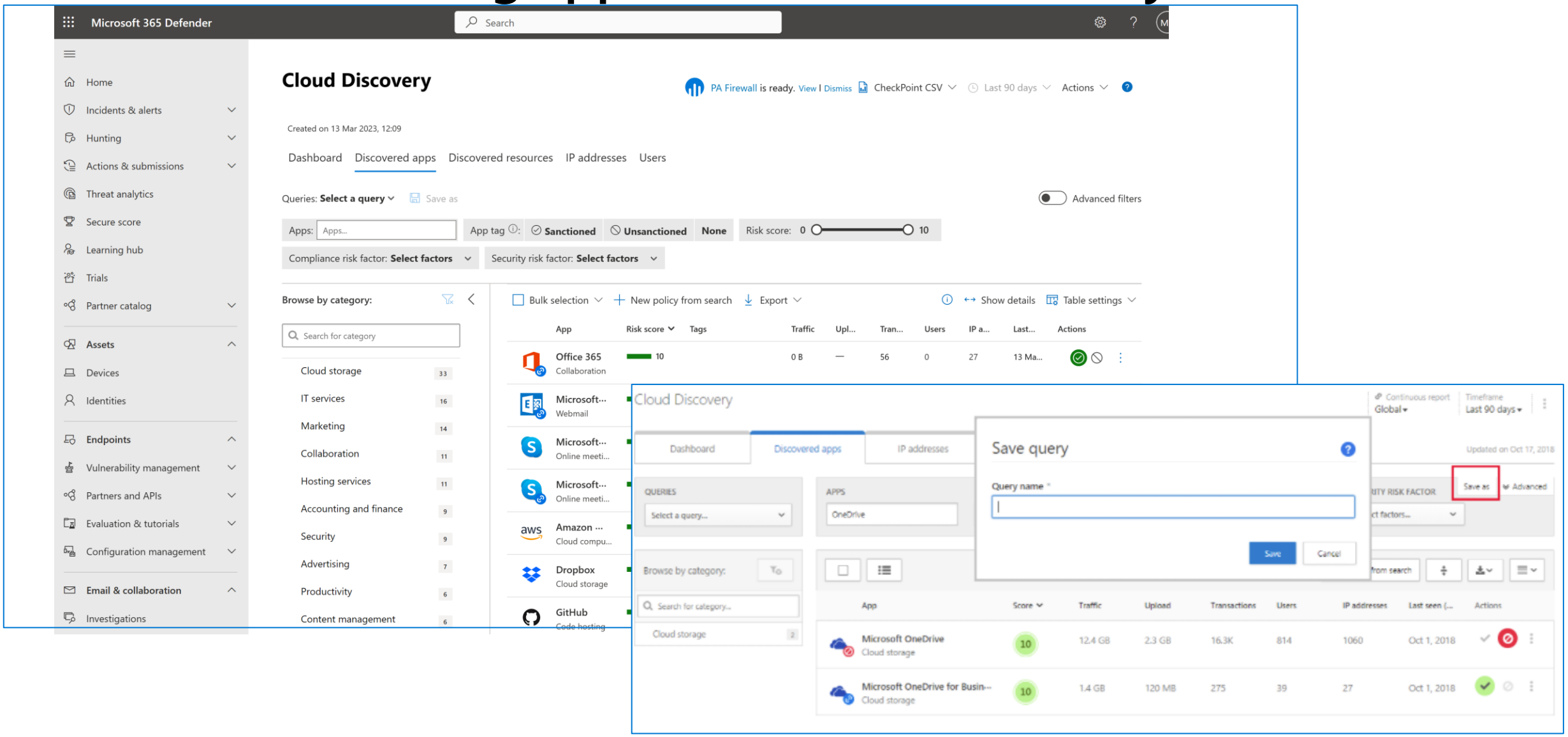
Providing admins with:

- Rich visibility
- Data control
- Sophisticated analytics
- Identification of cyberthreats

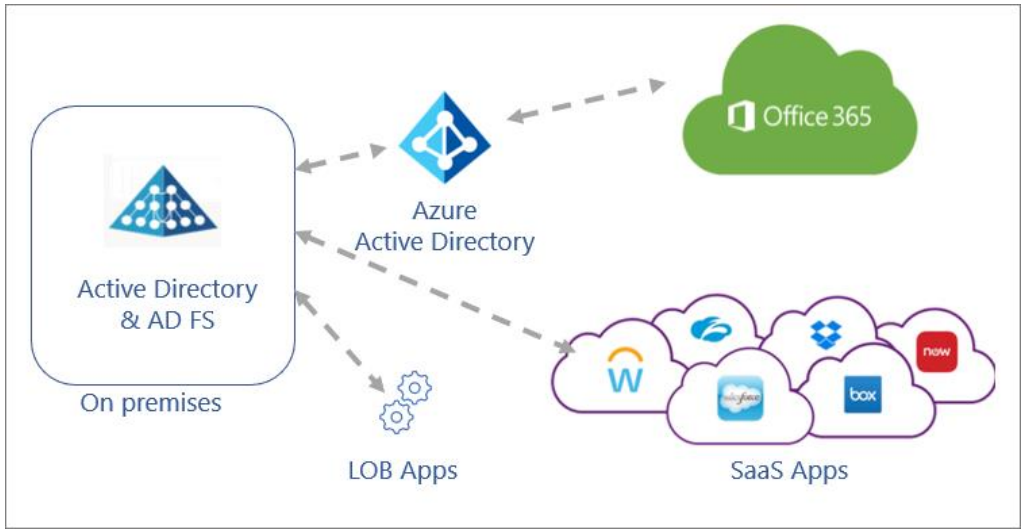
Set up Cloud Discovery with Microsoft Defender for Cloud Apps



MDCA – Discovering apps with Cloud Discovery

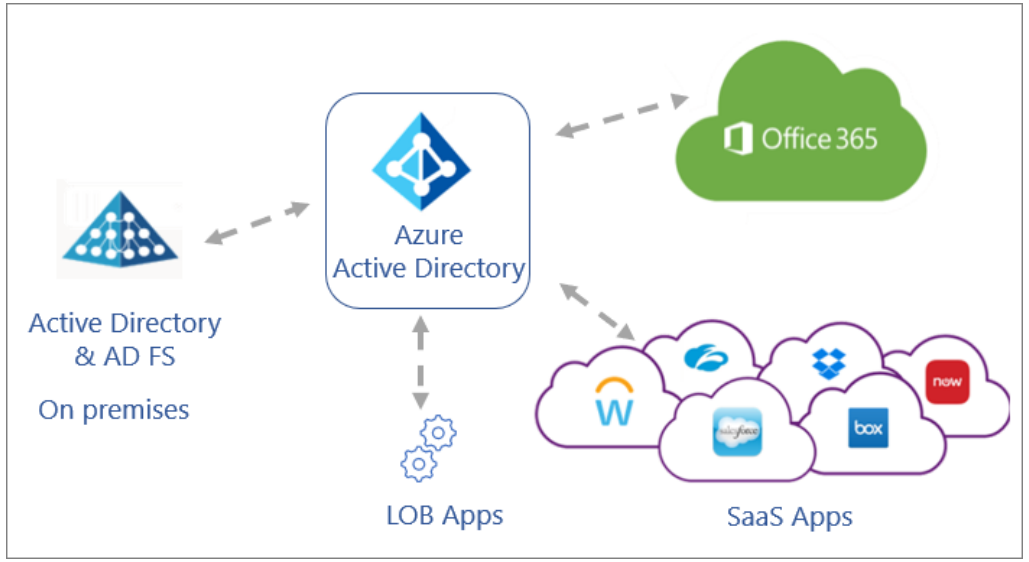


Active Directory Federation Services



AD FS extends single sign-on (SSO) functionality between trusted business partners without requiring users to sign in separately to each application.

To increase application security, your goal is to have a single set of access controls and policies across your on-premises and cloud environments.



Discover apps that can be migrated

There are two types of applications to migrate

- SaaS applications – procured by the organization
- Line-of-business applications – developed by the organization

Home > Usage & insights

Usage & insights | AD FS application activity ···

Usage & insights

Azure AD application activity (Pr...

AD FS application activity

Download Refresh

Date range

30 days

Application Identifier		Unique User Count
CL	clearforce_sp_ms	17
CV	cvent	6
FI	financialknowledge.net	26
GE	getabstract	79
OA	http://identity.office.net	31
AR	http://mssource.sourcing.ariba.com	32
WS	http://nebulaservices/tenantsite	31
EB	http://shibboleth.ebscohost.com	1
AA	https://appmapadmintool.cloudapp.net	1
AP	https://apportal.osdinfra.net	2294

Configure connectors to apps in MDCA

What is an app connector in Defender for Cloud Apps?

Capability	Apps with MDCA connectors
Connect to API provided by the app creator.	Connectors: • Atlassian • Azure • AWS • Box • DocuSign • Dropbox • GitHub • Google Workspace • Many others
Enables greater visibility into the apps.	
All communication over secure HTTPS.	
Common connector API limitations: • Throttling • API limits • Dynamic time-shifting • API windows	
Services vary by app	

How app connectors work in MDCA

Defender for Cloud Apps is deployed with system admin privileges to allow full access to all objects in your environment.

- The App Connector flow is as follows:
- Defender for Cloud Apps scans and saves authentication permissions.
- Defender for Cloud Apps requests the user list. The first time the request is done, it may take some time until the scan completes. After the user scan is over, Defender for Cloud Apps moves on to activities and files. As soon as the scan starts, some activities will be available in Defender for Cloud Apps.
- After completion of the user request, Defender for Cloud Apps periodically scans users, groups, activities, and files. All activities will be available after the first full scan.

Common services offered by app connector

Connections may take some time depending on the size of the tenant, the number of users, and the size and number of files that need to be scanned. 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.

Design and implement access management for apps

Azure Active Directory – Enterprise Applications

Azure AD → Enterprise Applications

Gallery of thousands of pre-integrated applications

- Many of the applications your organization uses are already in the gallery
- Add your own business apps

After an application is added to your Azure AD tenant, you can:

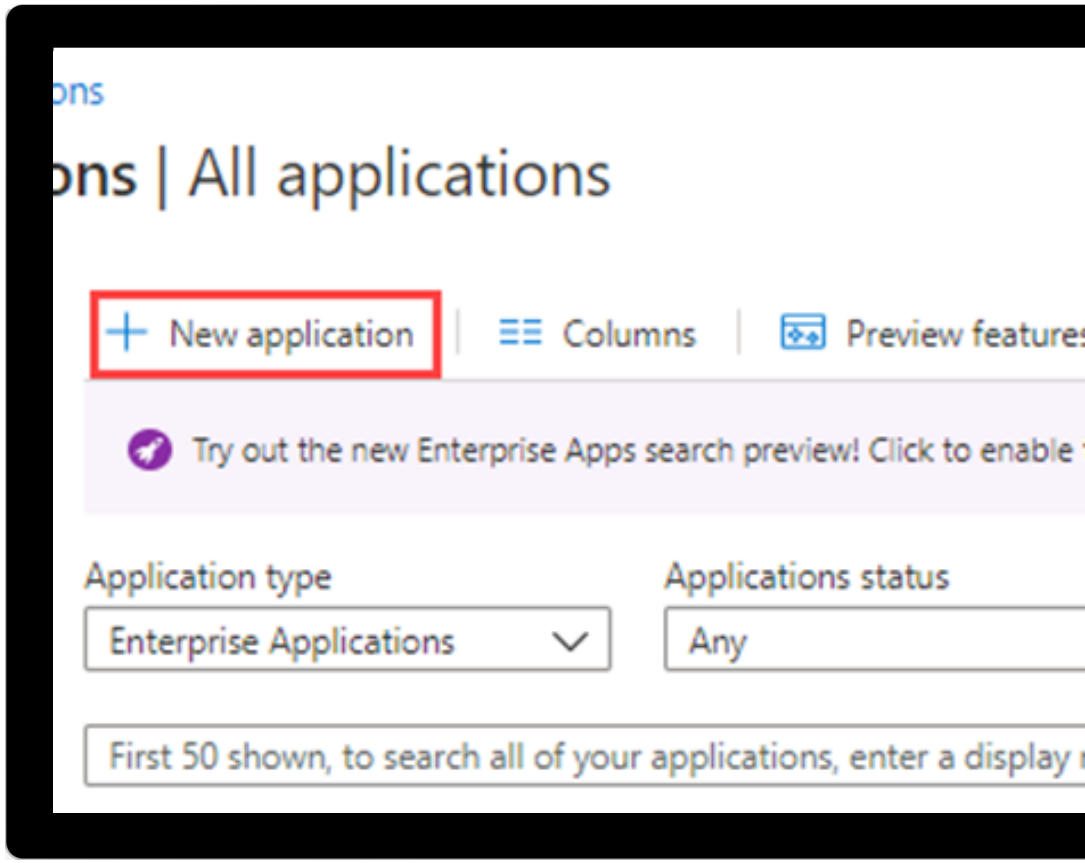
- Configure properties for the app
- Manage user access to the app with a Conditional Access policy
- Configure single sign-on

Exercise: Implement access management for apps

Add an app to your Azure AD tenant:

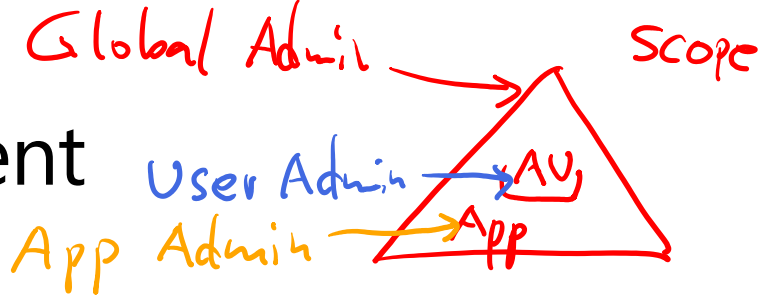
Add an Enterprise app and assign your administrator account

[Launch this Exercise in GitHub]



Design and implement app management roles

Delegate application register and management



- By restricting who can register applications and manage them
- By assigning one or more owners to an application
- By assigning a built-in administrative role that grants access to manage configuration in Azure AD for all applications
- By creating a custom role defining specific permissions, and assigning it

Built in admin application roles

Application Administrator

Includes the ability to manage all aspects of enterprise applications; including registrations and application proxy settings

Cloud Application Administrator

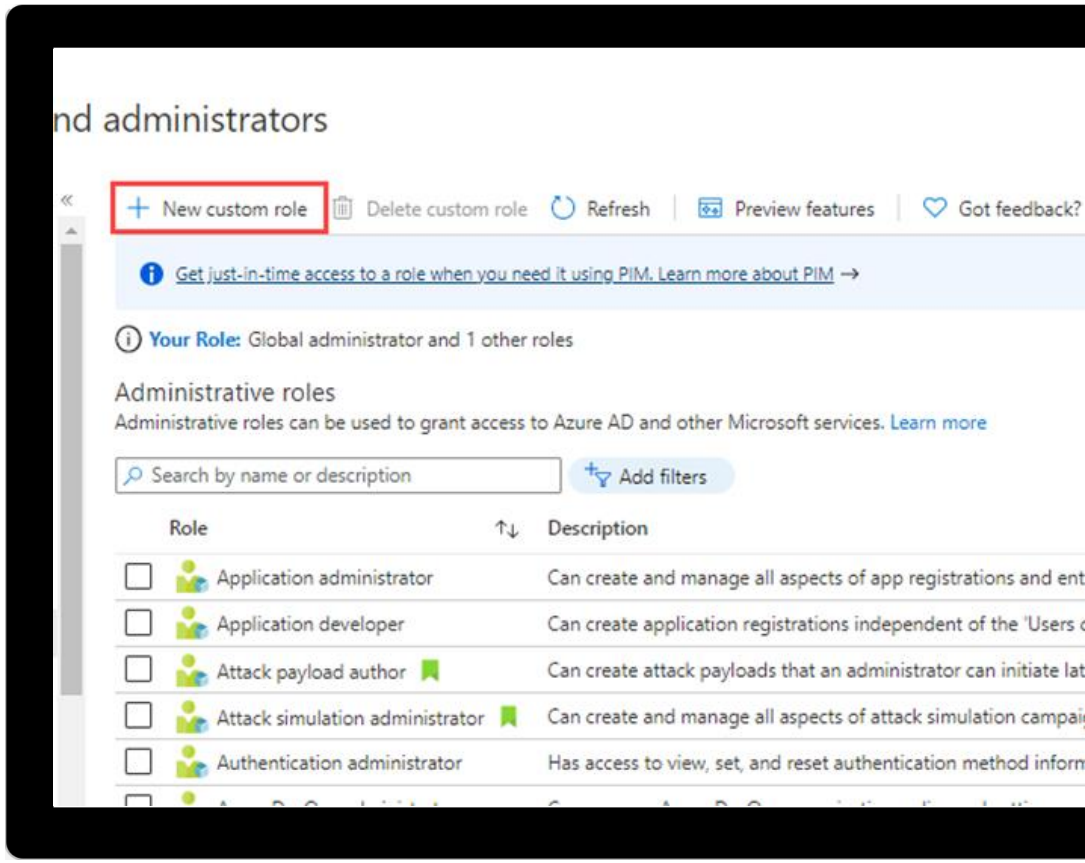
Includes the ability to manage most aspects of enterprise applications, but **excludes the ability to manage application proxy settings**

Exercise: Create a new custom role to grant access to manage app registrations

A custom role can be assigned at organization-wide scope or at the scope of a single Azure AD object.

Create a new custom role that can be used to grant access to manage app registrations.

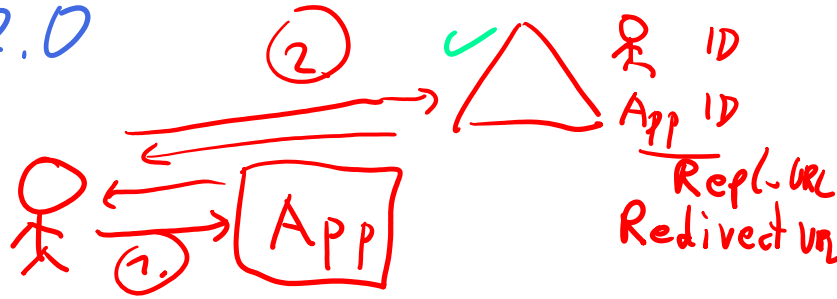
[Launch this Exercise in GitHub]



Configure pre-integrated (gallery) SaaS apps

Enterprise Application Properties

OAuth 2.0



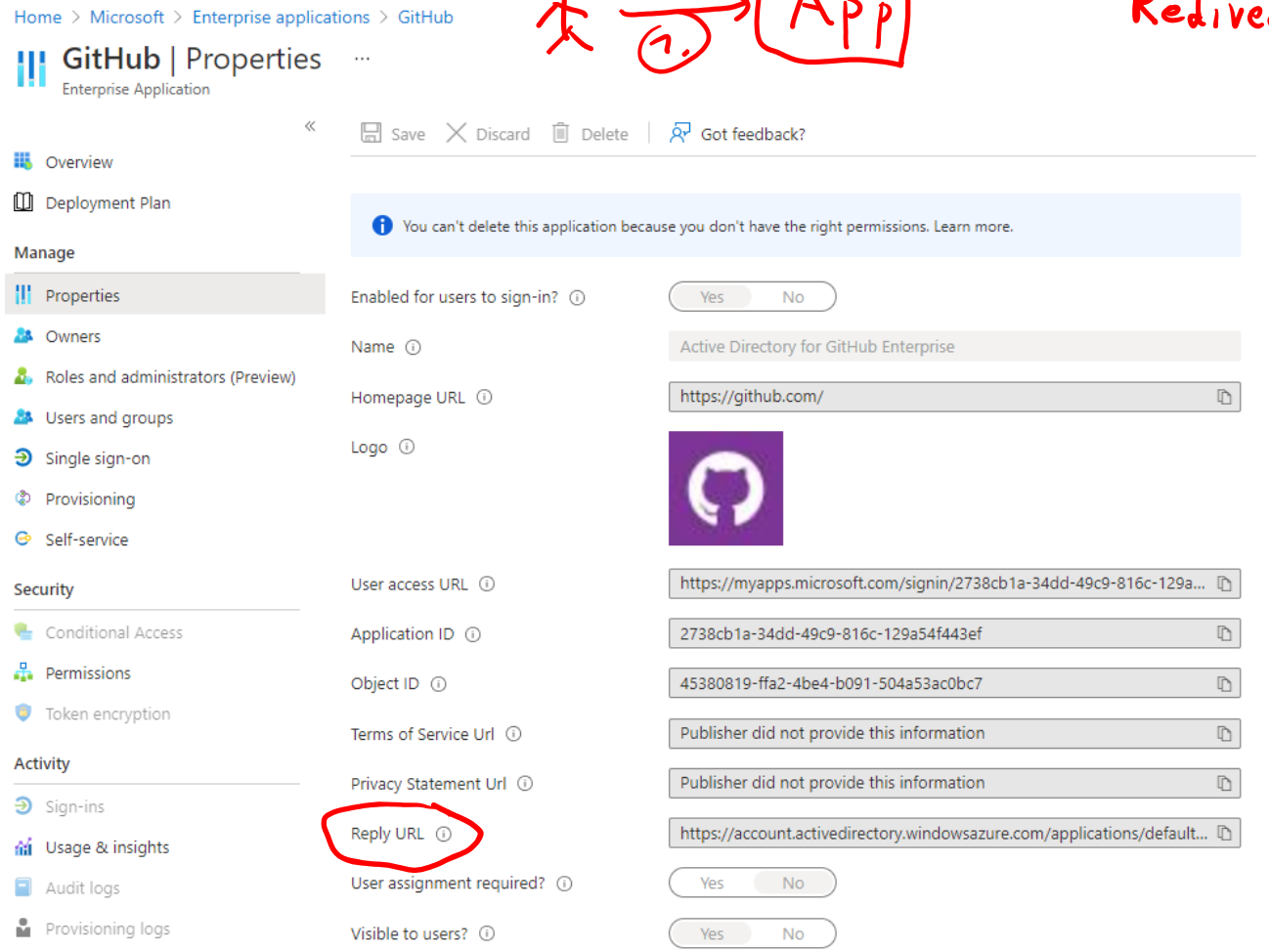
Give the application a name

Pick the URL that opens for users

Name / Homepage URL

ApplicationID / ObjectID

Terms of Service / Privacy Statement



Configure app properties

Enabled for users to sign in?

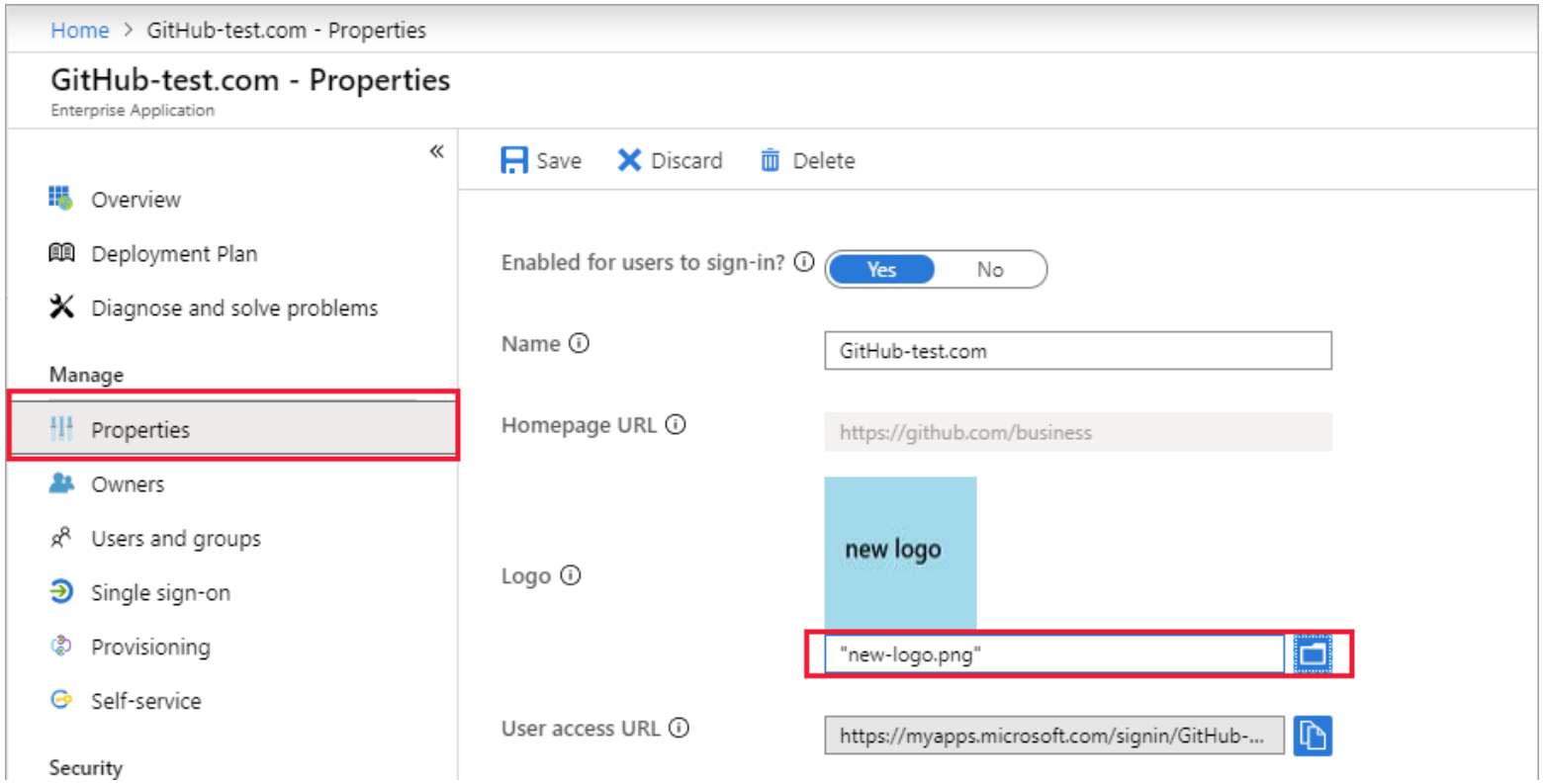
User assignment required?

Visible to users?

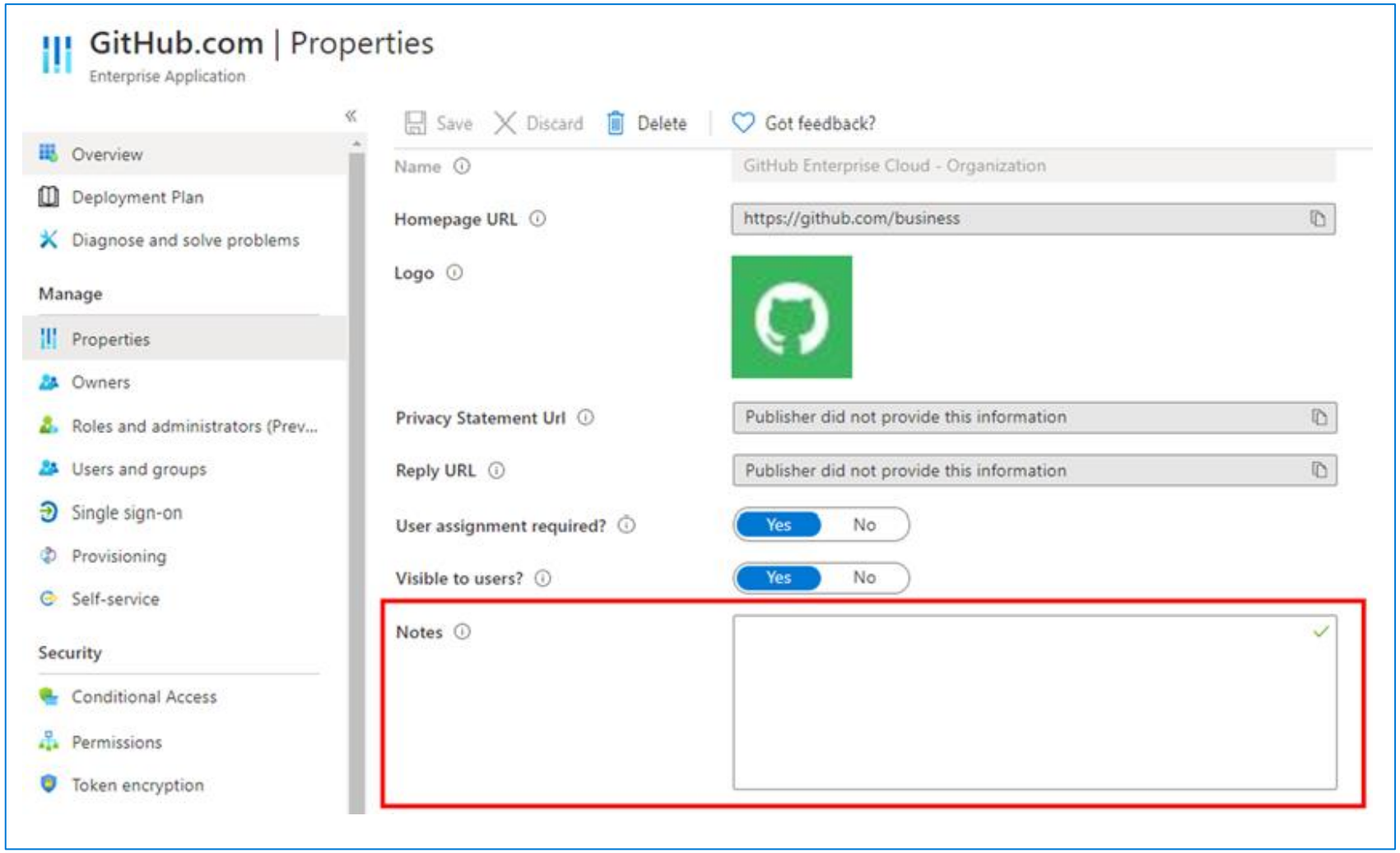
Behavior based on option choices

Enabled for users to sign in?	User assignment required?	Visible to users?	Behavior for users who have either been assigned to the app or not.
Yes	Yes	Yes	• Assigned users can see the app and sign in. • Unassigned users cannot see the app and cannot sign in.
Yes	Yes	No	• Assigned users cannot see the app but they can sign in. • Unassigned users cannot see the app and cannot sign in.
Yes	No	Yes	• Assigned users can see the app and sign in. • Unassigned users cannot see the app but can sign in.
Yes	No	No	• Assigned users cannot see the app but can sign in. • Unassigned users cannot see the app but can sign in.
No	Yes	Yes	• Assigned users cannot see the app and cannot sign in. • Unassigned users cannot see the app and cannot sign in.
No	Yes	No	• Assigned users cannot see the app and cannot sign in. • Unassigned users cannot see the app and cannot sign in.
No	No	Yes	• Assigned users cannot see the app and cannot sign in. • Unassigned users cannot see the app and cannot sign in.
No	No	No	• Assigned users cannot see the app and cannot sign in. • Unassigned users cannot see the app and cannot sign in.

Custom Logo



Add Notes

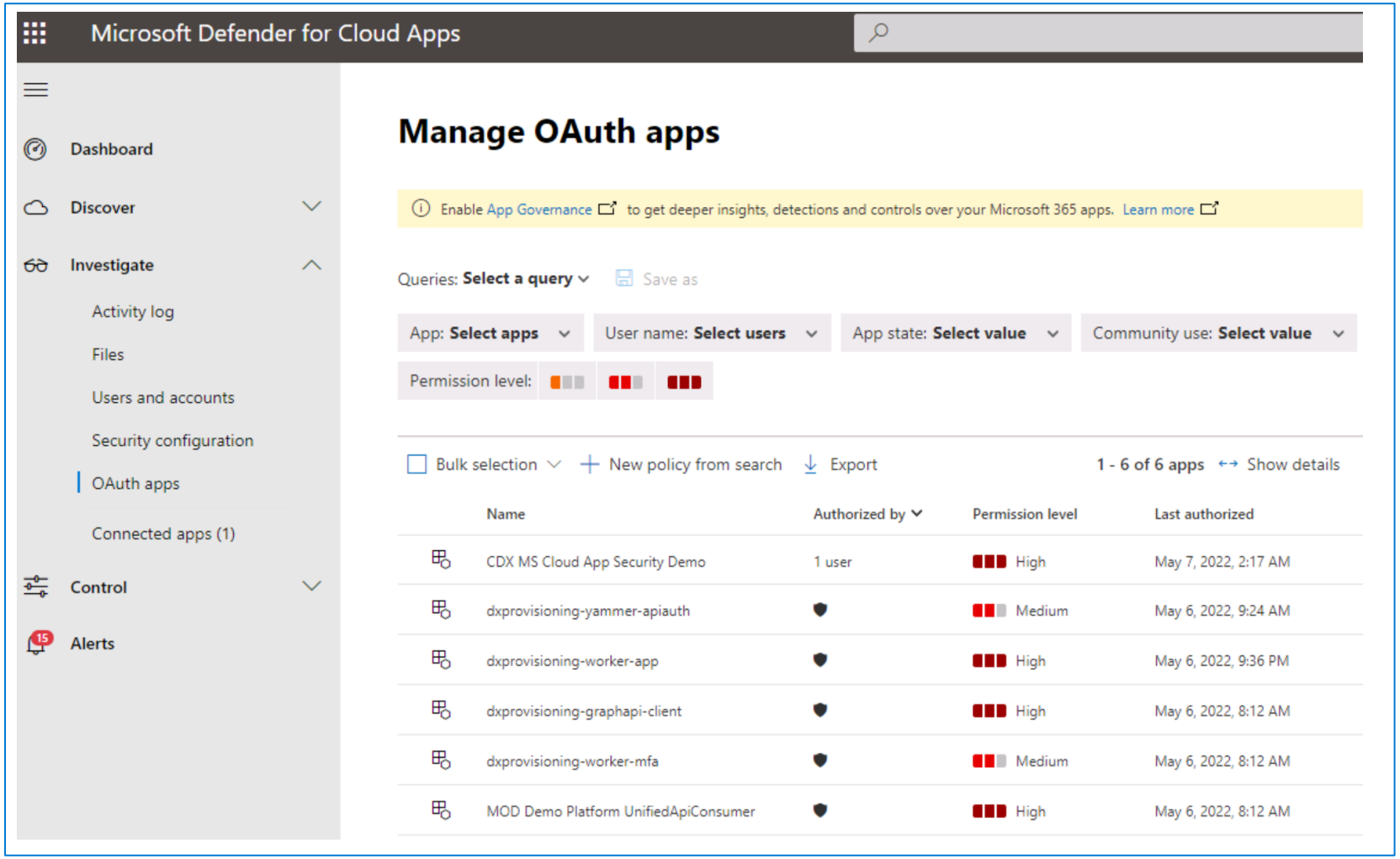


Add any information that is relevant for the management of the application

Implement and manage policies for OAuth apps

Create a new OAuth app policy

1. Launch **Microsoft Defender for Cloud Apps** at https://www.cloudappsecurity.com.
2. Under **Investigate**, select **OAuth apps**.
3. Filter the apps according to your needs.
    1. For example, you can view all apps that request Permission to Modify calendars in your mailbox.
4. Select the **New policy** from search button.



Summary

In this section, you learned how to:

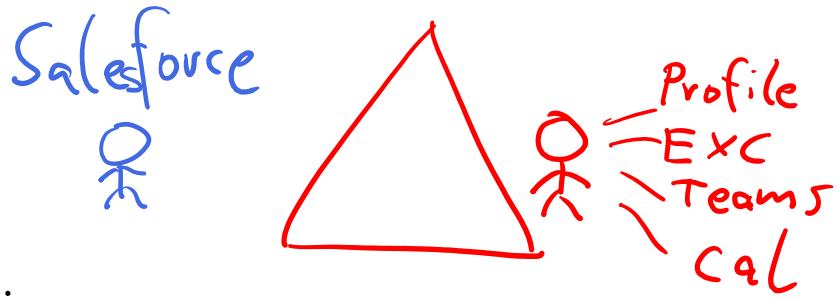
- Discover apps by using MDCA or ADFS app report
- Design and implement access management for apps
- Design and implement app management roles
- Configure pre-integrated (gallery) SaaS apps

Microsoft Security

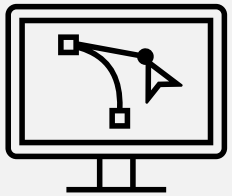
Implement and monitor the integration of enterprise apps for SSO

Learning Objectives

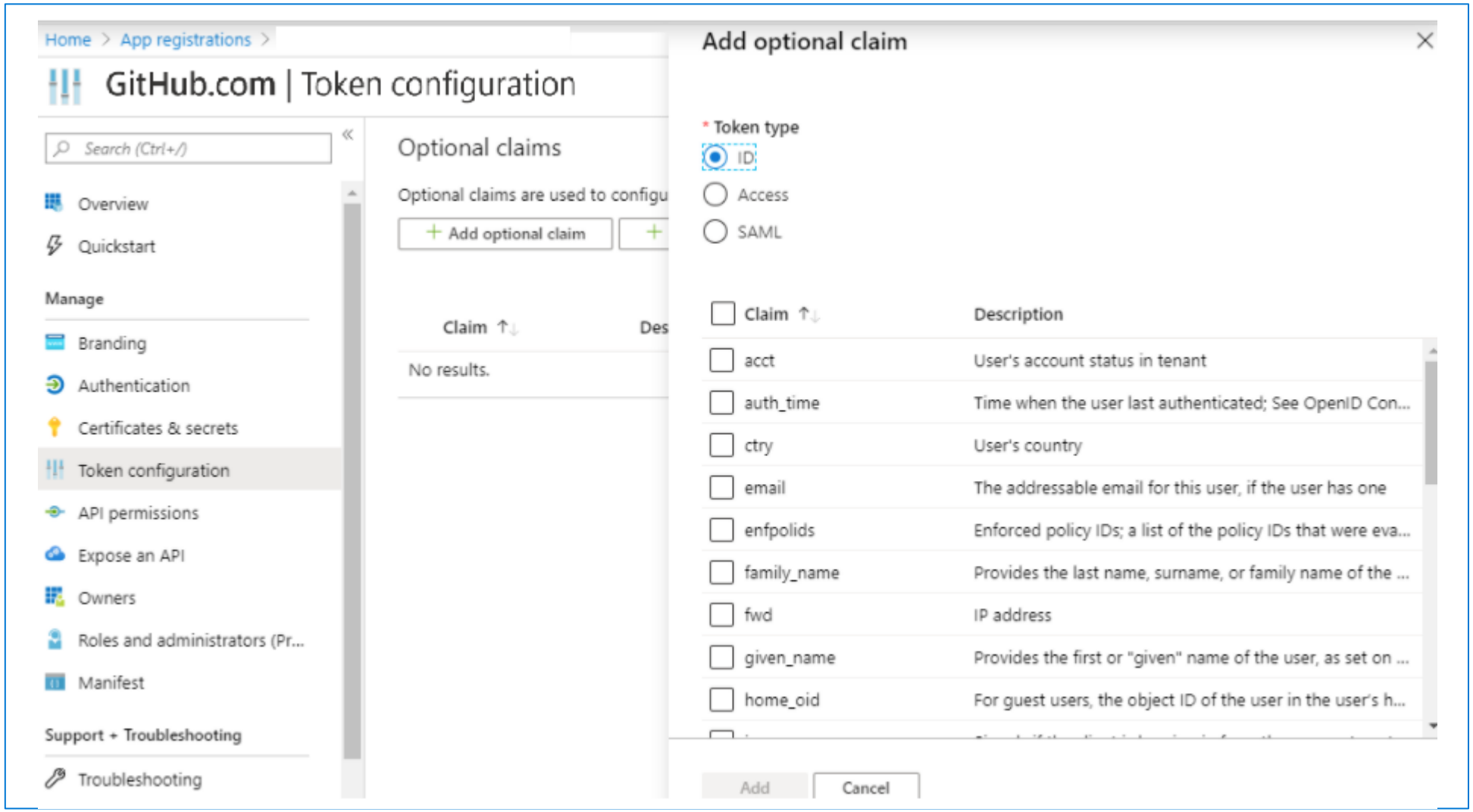
- Implement token customizations
- Implement and configure consent settings
- Integrate on-premises apps by using Azure AD application proxy
- Integrate custom SaaS apps for SSO
- Implement application user provisioning
- Monitor and audit access/Sign-On to Azure Active Directory integrated enterprise applications
- Create and manage application collections (in My Apps)



Implement token customizations



Token Configuration – Claims – SAML based SSO



Customize Tokens for Azure AD

JWT

Get-AzAccessToken
~60min

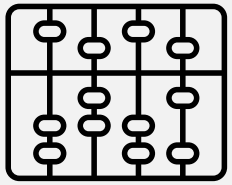
Access and ID token lifetimes (minutes)

The lifetime of the OAuth 2.0 bearer token and ID tokens

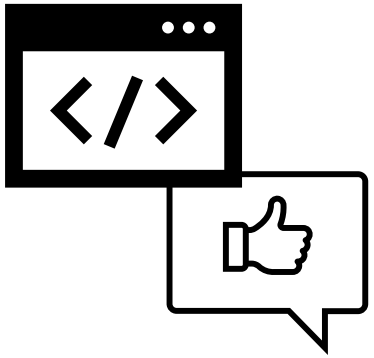
Lifetime length (days)

After this time period elapses, the user is forced to reauthenticate

Implement and configure consent settings



Why is consent important?



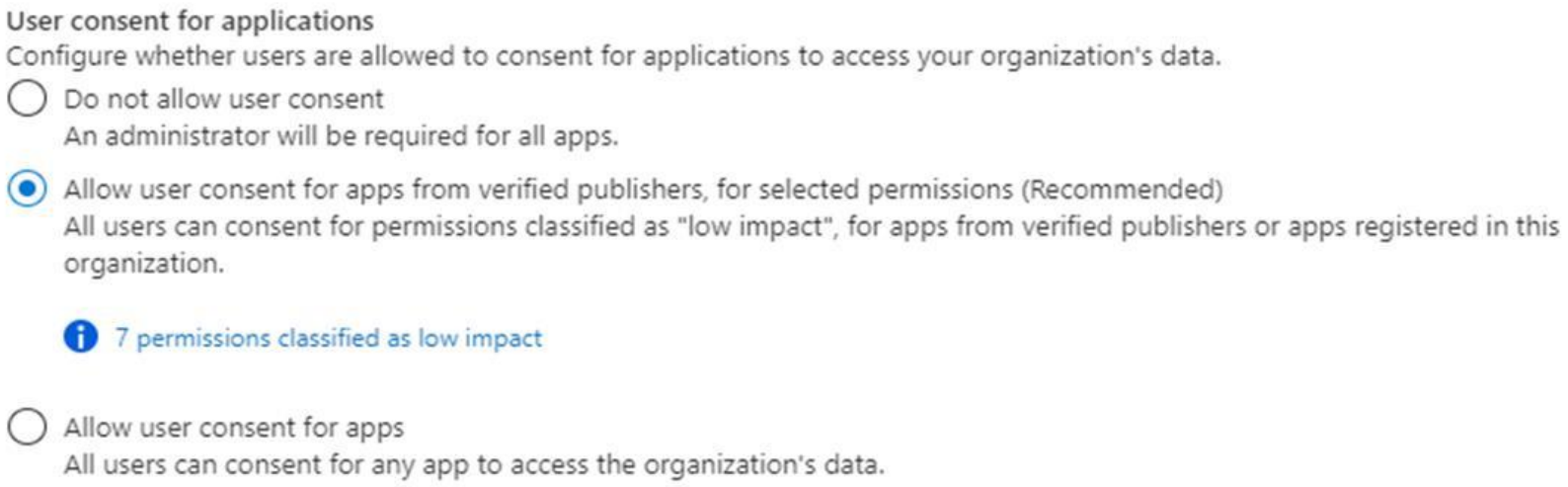
A user or admin must grant permissions to an app before it can access company data.

Users can allow apps access to specific information, like a Mailbox; but not access to organization servers.

Users may not think through ramifications of granting access; they just want to use an app to do a task

What are Consent Settings

- Before an application can access the organization's data, a user must grant the application permissions to do so
- All users can consent to applications for permissions that do not require administrator consent
- By allowing users to grant apps access to data, users can acquire useful applications and be productive



User consent settings

Disable user consent

Users cannot grant permissions to applications. Requires an admin to grant.

Users can consent to apps from verified publishers

Users can only consent to apps that were published by a verified publisher

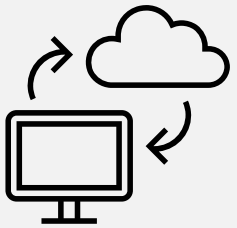
Users can consent to all apps

Users can consent to any permission

Custom app consent policy

Users can consent to custom app consent policies

Integrate on-premises apps by using Azure AD application proxy

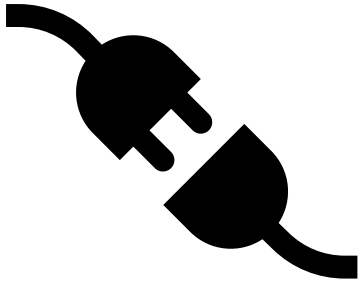


What is Application Proxy?

Feature to allow user to access on-premises application

Proxy service runs in the cloud and has an App Proxy Connector running on-premises

Securely passes sign-on tokens from Azure AD to the application



Value of Application Proxy

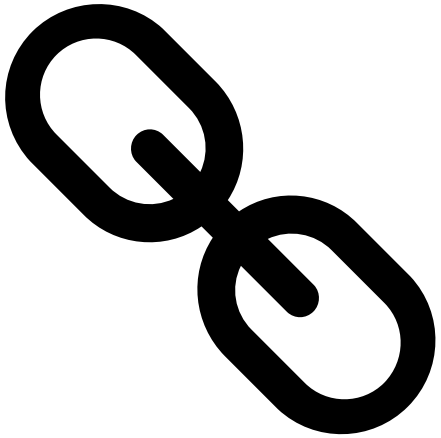
Protocol translation to / from Modern Authentication

Example - Convert Kerberos token to a modern auth token

Use seamless single-sign-on to remove user action to log in multiple times

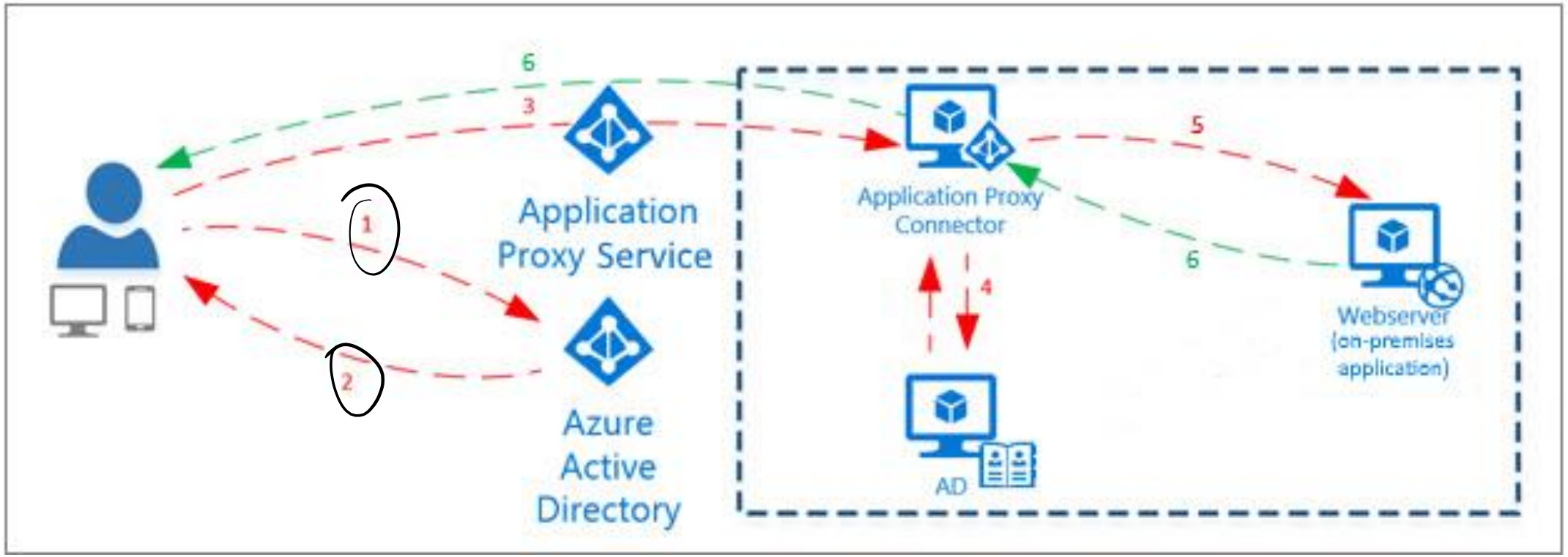
Kerberos!

Allows apps to stay on-premises (for whatever reason), but still be securely available to the user



Application Proxy

Application Proxy is a feature of Azure AD that enables users to access on-premises web applications from a remote client

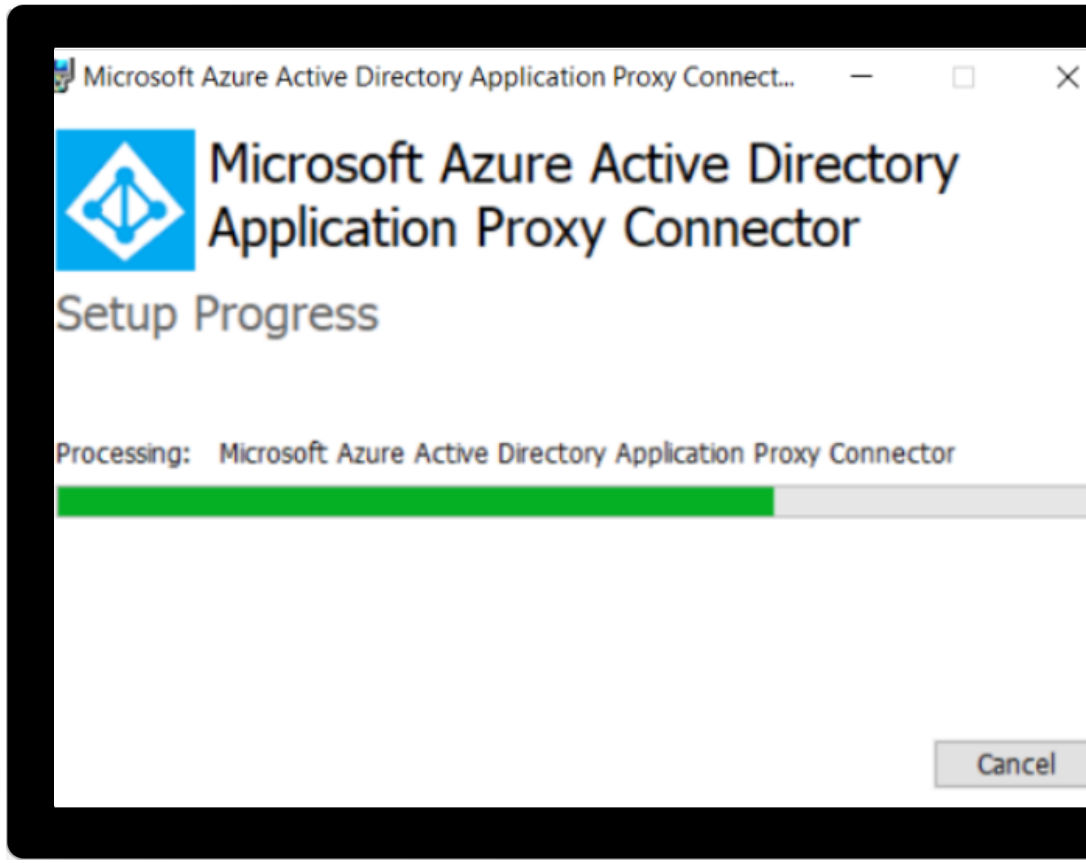


Exercise: Add an on-premises application for remote access through Application Proxy in Azure Active Directory

Interactive Guide

Enable integrated windows authentication to on-premises applications with Azure AD application proxy

[Visit this Interactive Guide in Microsoft Learn]

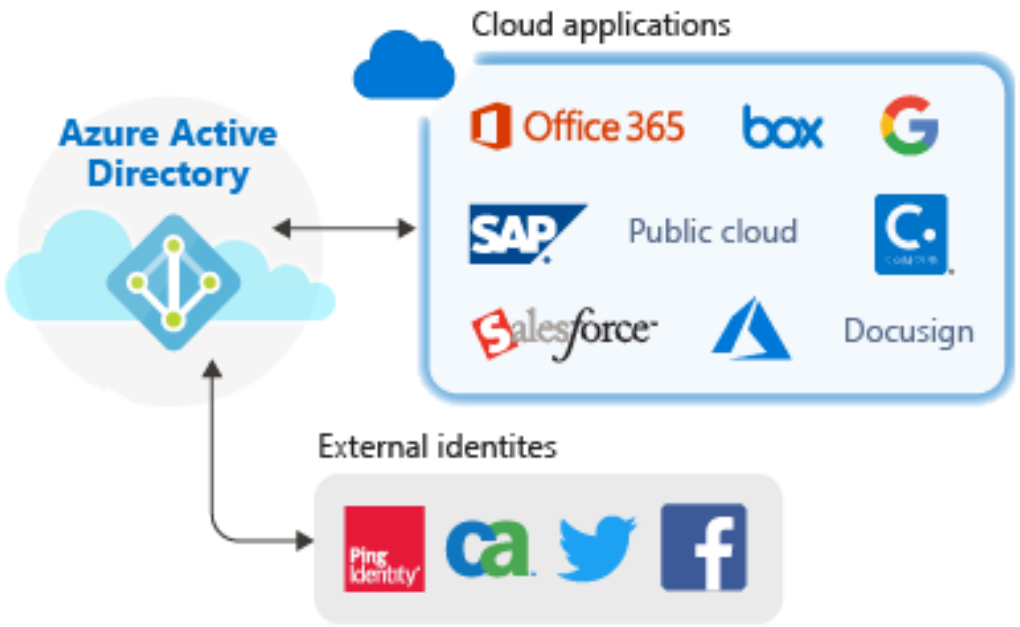


Integrate custom SaaS apps for SSO

SSO for SaaS apps

You can use Azure AD as your identity system for just about any app. Many apps are already pre-configured and can be set up with minimal effort. These pre-configured apps are published in the Azure AD App Gallery.

You can manually configure most apps for single sign-on if they aren't already in the gallery. Azure AD provides several SSO options. SAML-based SSO and OIDC-based SSO.



SaaS App Integration Tutorials

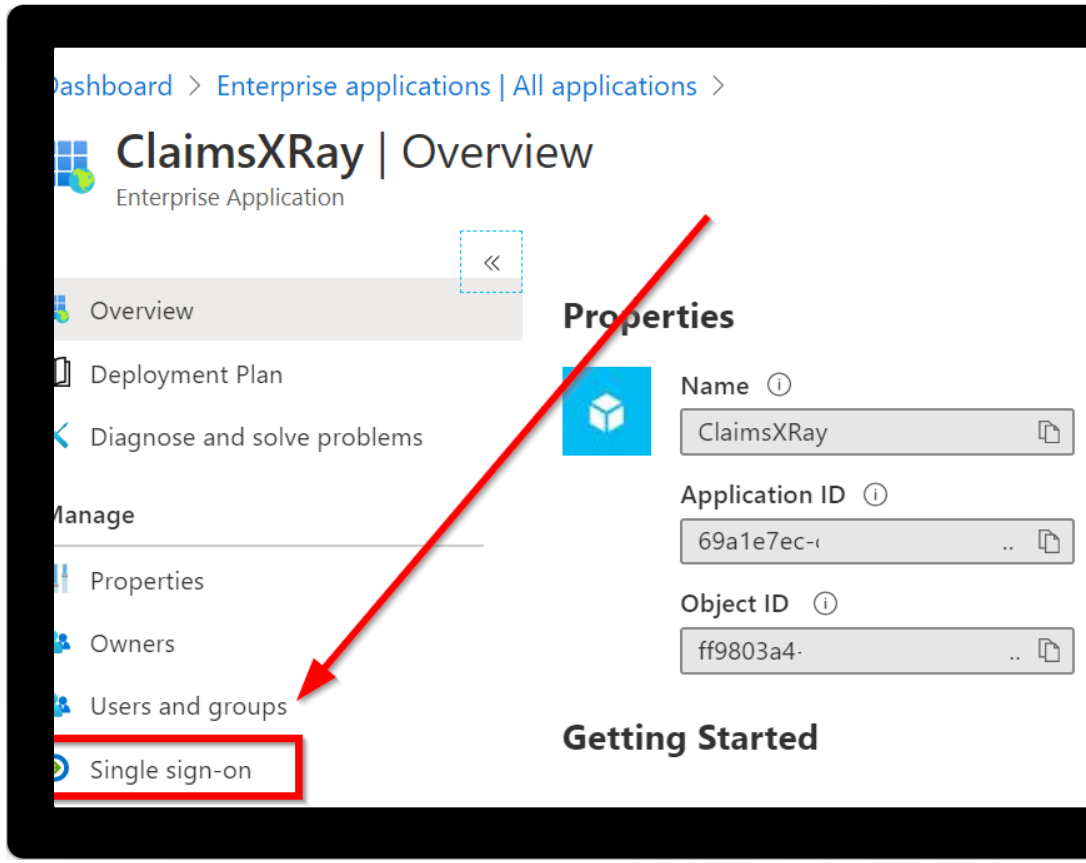
https://docs.microsoft.com/en-us/azure/active-directory/saas-apps/tutorial-list

Exercise: Troubleshoot SAML single sign-on for custom SaaS apps

Interactive Guide

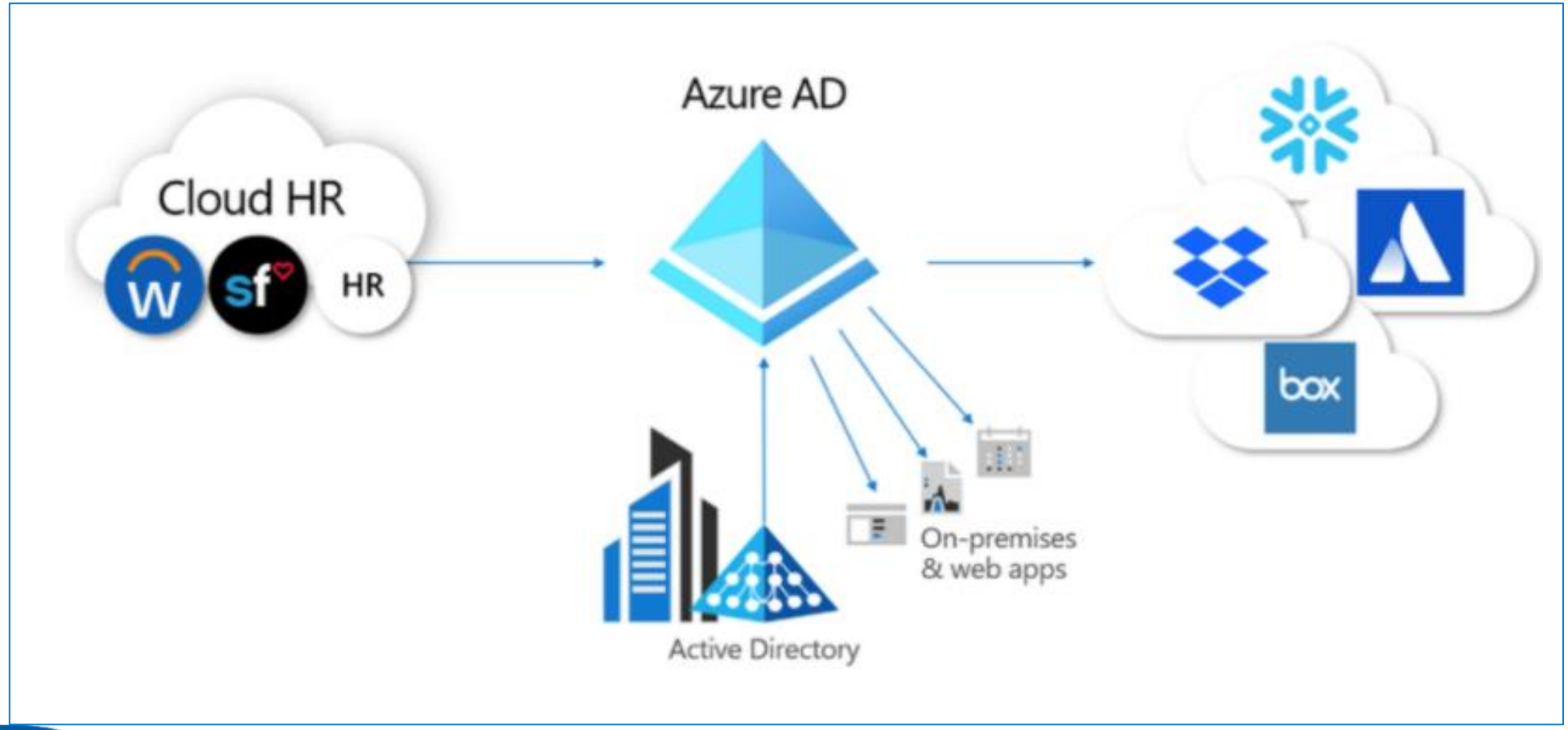
Integrate an application in Azure AD providing the single sign-on experience

[Visit this Interactive Guide]



Implement application user provisioning

Application user provisioning



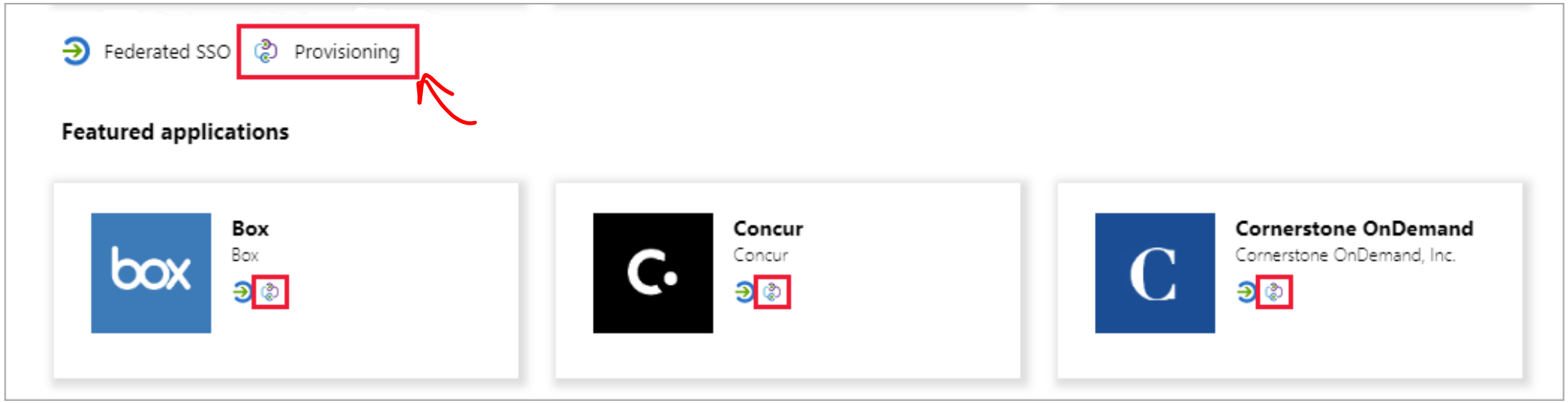
Manual vs. Automatic provisioning

Manual provisioning

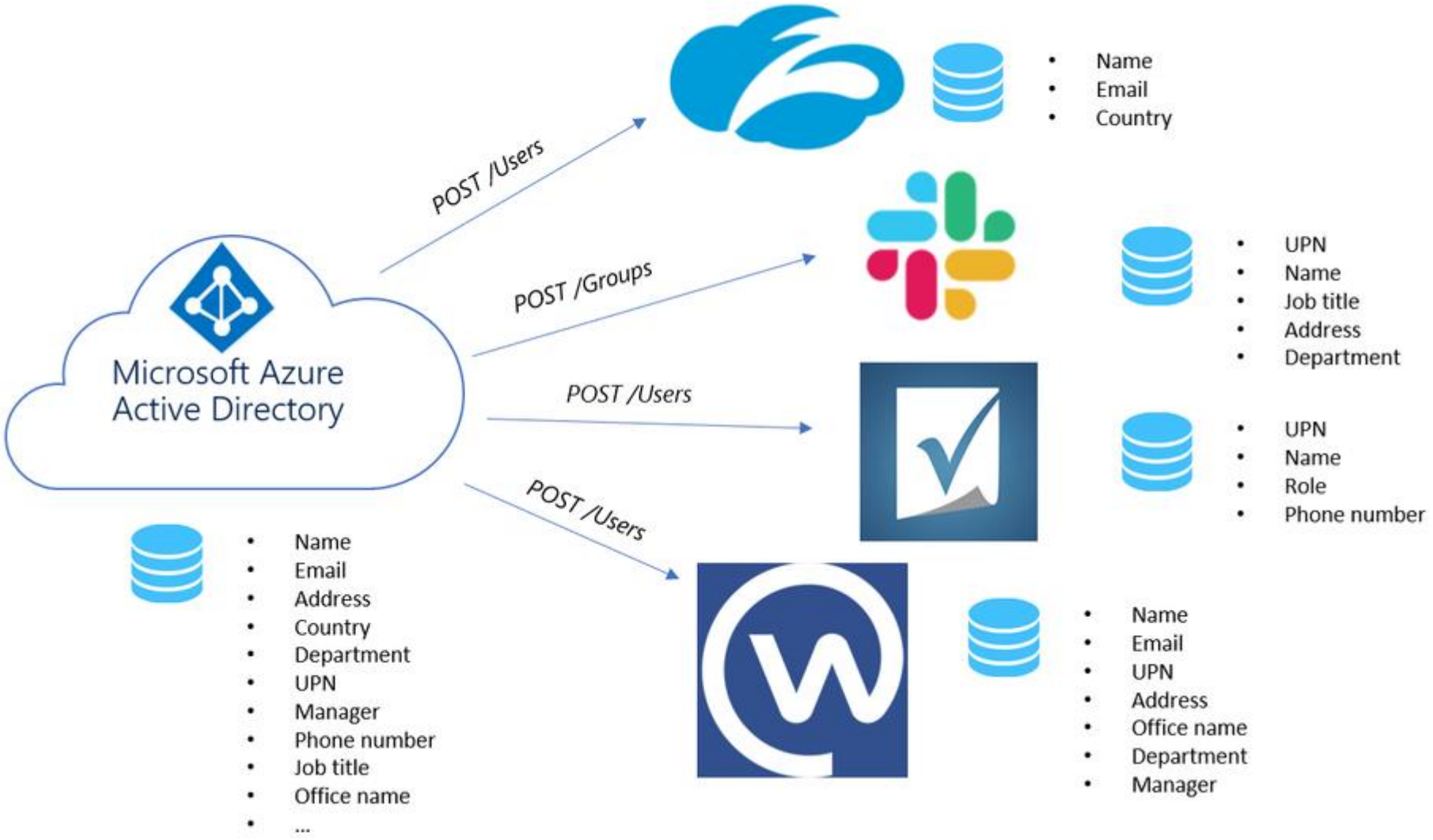
there is no automatic Azure AD provisioning connector for the app yet. User accounts must be created manually

Automatic provisioning

an Azure AD provisioning connector has been developed for this application.



SCIM provisioning overview



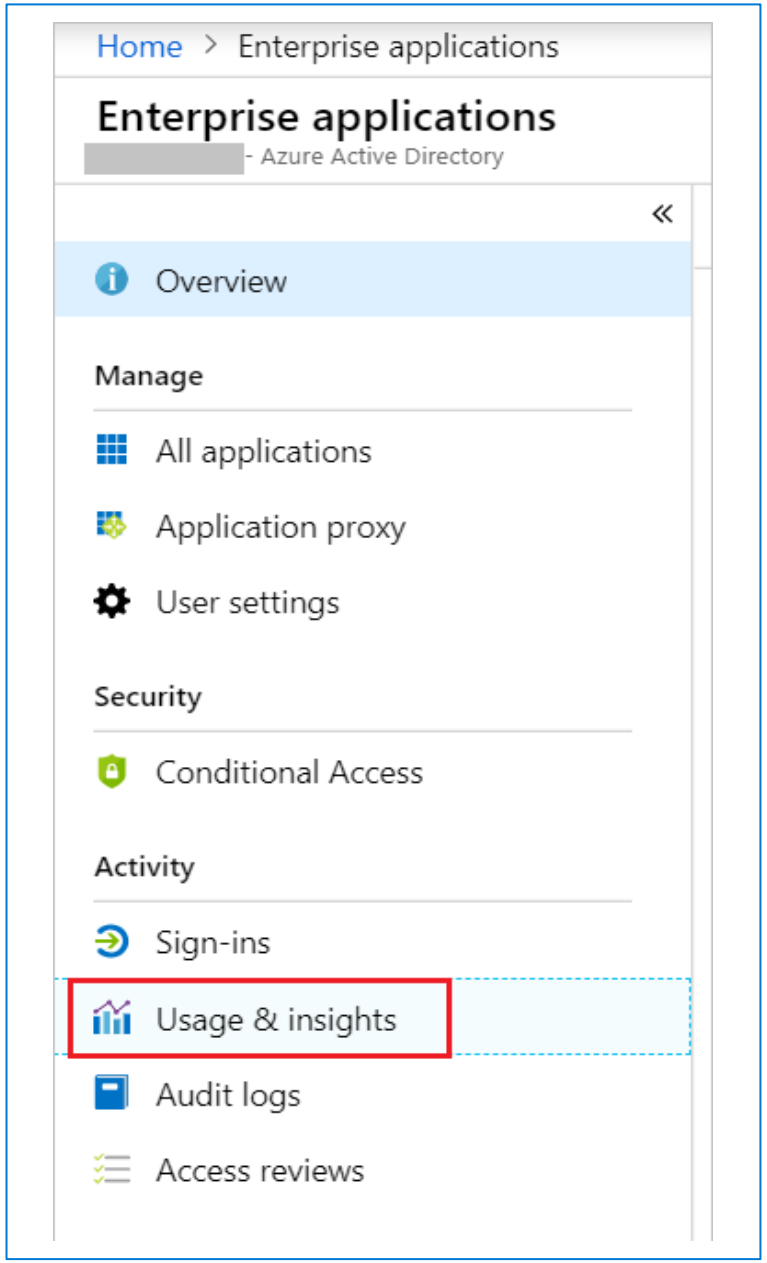
Monitor and audit access/Sign-On to Azure Active Directory integrated enterprise applications

Usage and Insight Reports

What are the top used applications in the organization?

What applications have the most failed sign-ins?

What are the top sign-in errors for each application?



Audit Logs (in Azure AD)

Record of system activities for compliance

- the date and time of the occurrence
- the service that logged the occurrence
- the category and name of the activity (what)
- the status of the activity (success or failure)
- the initiator/actor (who) of an activity

Monitoring

- Sign-ins
- Audit logs
- Provisioning logs
- Logs
- Diagnostic settings
- Workbooks

Download | Export Data Settings | Refresh | Columns | Got feedback?

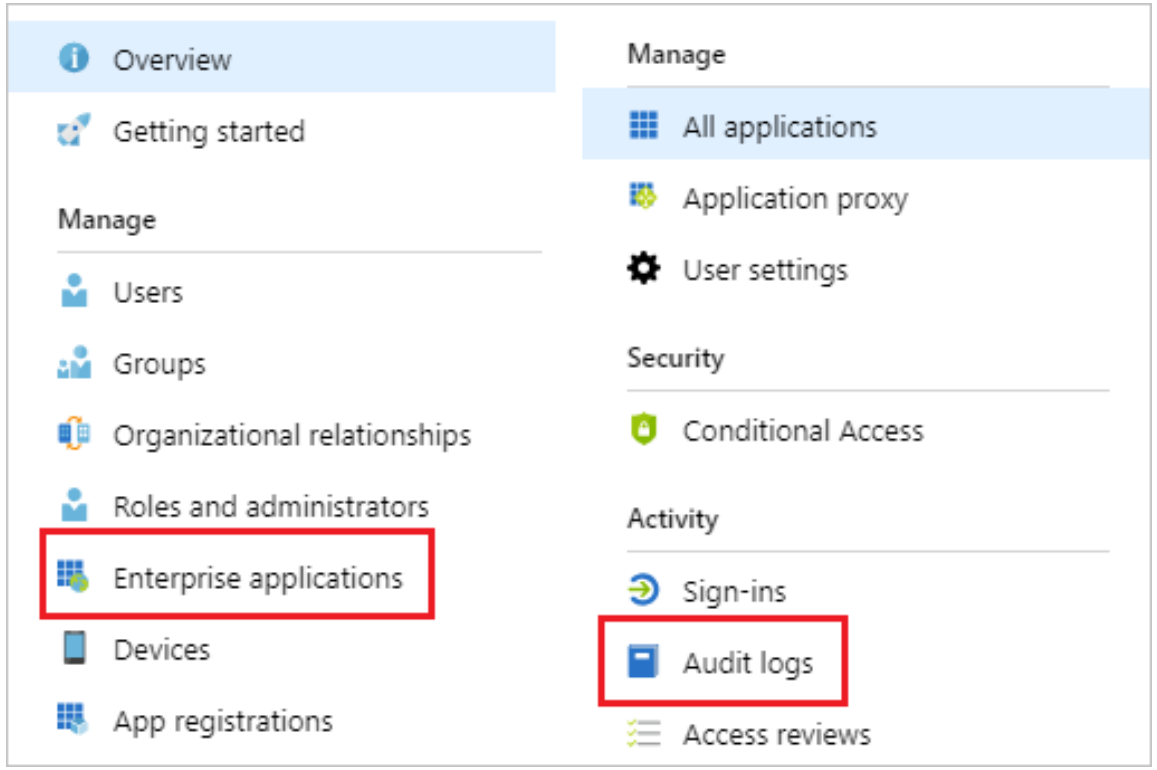
Date : **Last 24 hours** | Show dates as : **Local** | Service : **All** | Category : **All** | Activity : **All** | Add filters

Date	Service	Category	Activity	Status	Status reason	Target(s)	Initiated by (actor)
7/9/2021, 9:38:48 AM	Core Directory	ApplicationManagement	Update service principal	Success		Zoom	
7/9/2021, 9:38:48 AM	Core Directory	ApplicationManagement	Update service principal	Failure	Microsoft.Online.Directory...	Zoom	AAD App Management
7/9/2021, 9:38:48 AM	Core Directory	ApplicationManagement	Update service principal	Success		Zoom	AAD App Management
7/9/2021, 9:36:10 AM	Core Directory	ApplicationManagement	Update application	Success		Zoom	AAD App Management
7/9/2021, 9:36:10 AM	Core Directory	ApplicationManagement	Update service principal	Success		Zoom	AAD App Management
7/9/2021, 9:36:09 AM	Core Directory	ApplicationManagement	Add service principal	Success		Zoom	AAD App Management
7/9/2021, 9:36:09 AM	Core Directory	ApplicationManagement	Add application	Success		Zoom	AAD App Management
7/9/2021, 9:28:02 AM	Core Directory	ApplicationManagement	Add service principal	Success		AAD App Management	

Enterprise applications audit logs

Application-based audit reports

- What applications have been added or updated?
- What applications have been removed?
- Has a service principal for an application changed?
- Have the names of applications been changed?
- Who gave consent to an application?



Create and manage application collections

Create app collections

Create and admin application collection	Create a collection using the My Apps portal
1. Go to Azure Active Directory then select Enterprise Applications. 2. Under Manage, select App Launchers. 3. Select New collection. 4. In the New collection page, enter a Name and Description. 5. Select the Applications tab. Select + Add application to open the Add applications page. 6. Select all the applications you want to add. 7. When you're finished adding applications, select Add. 8. Select the Owners tab. Select + Add users and groups. 9. Select Review + Create. The properties for the new collection appear.	1. Open the My Apps portal. 2. Select the ellipsis (...) on the apps screen. 3. Choose Manage collections. 4. Select Create collection. 5. Select the + Add apps option to add all the apps you want in the collection. 6. After picking your apps, select the Add selected apps button. 7. Give the collection a name and choose Create collection.

Summary

In this section, you learned how to:

- Implement Token Customizations
- Implement and configure consent settings
- Integrate on-premises apps by using Azure AD application proxy
- Integrate custom SaaS apps for SSO
- Implement application user provisioning
- Monitor and audit access/Sign-On to Azure Active Directory integrated enterprise applications

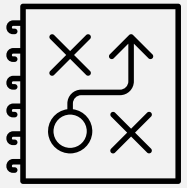
Microsoft Security

Implement app registrations

Learning Objectives

- Plan your line of business application registration strategy
- Implement application registrations
- Configure application permissions
- Implement application authorization
- Manage and monitor applications with App Governance

Plan your line of business application registration strategy



Why do applications integrate with Azure AD?

Add applications to Azure AD to leverage one or more of the services it provides, including:

- Application authentication and authorization
- User authentication and authorization
- Single sign-on (SSO) using federation or password
- User provisioning and synchronization
- OAuth authorization services
- Application publishing and proxy
- Directory schema extension attributes
- Role based access control

MS Graph
Consent

RBAC
Azure

App Permissions

Application objects and Service principals

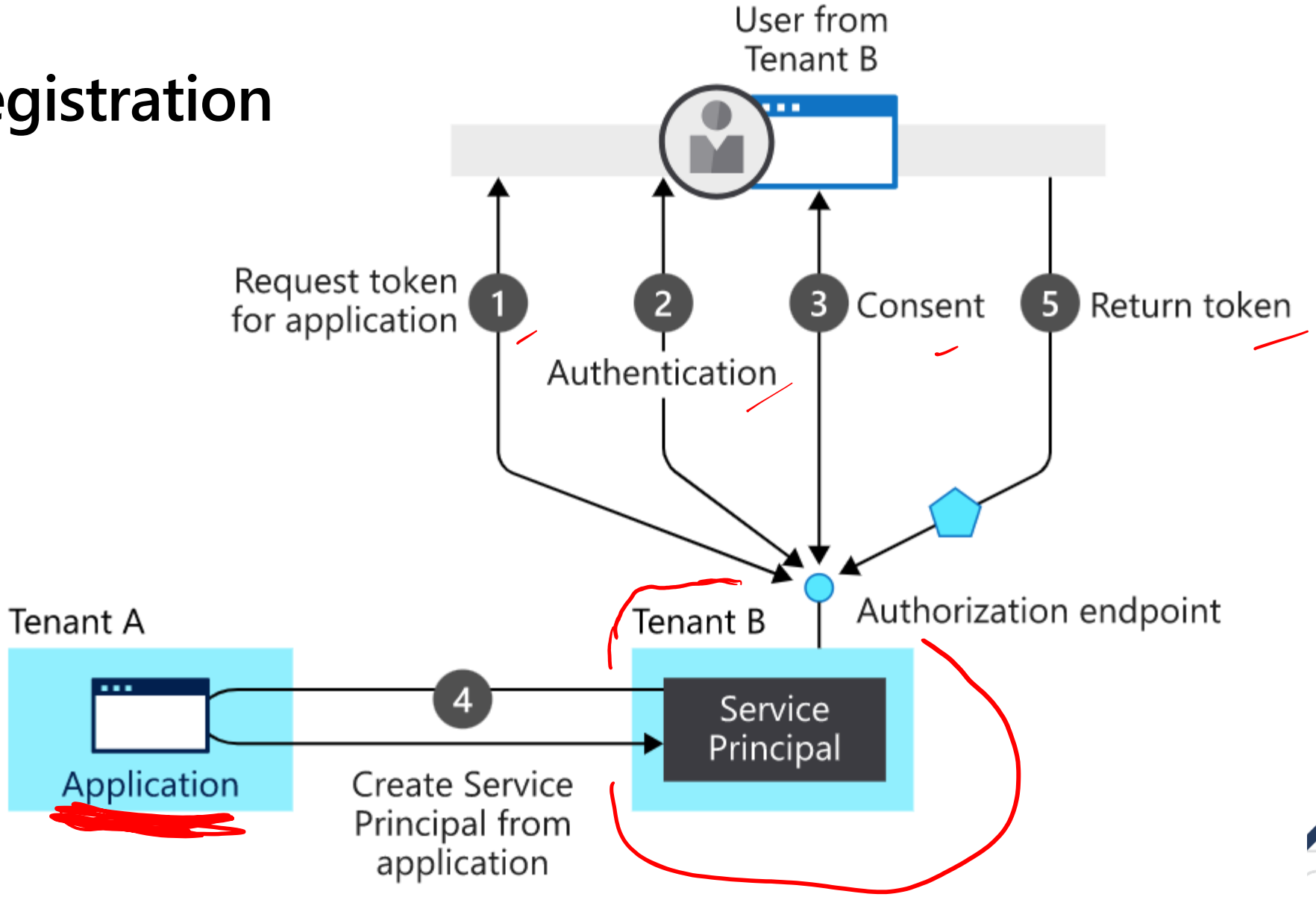
Application Objects:

- Define and describe the application to Azure AD, enabling it to know how to issue tokens based on its settings
- Will only exist in their tenant

Service principals

- Govern an application connecting to Azure AD
- Can be considered the instance of the application in your tenant

New app registration



Who has permission to add applications to my Azure AD instance?

- By default, all users in your directory have rights to register application objects they are developing, and they have discretion over which applications they share or give access to their organizational data through consent
- When the first user in your directory signs into an application and grants consent, that will create a service principal in your tenant; otherwise, the consent grant information will be stored on the existing service principal

Tenancy in Azure Active Directory

Configure your app to be single-tenant or multi-tenant

WHO CAN SIGN IN TO YOUR APP?

Audience	Single/multi-tenant	Who can sign in
Accounts in this directory only	Single tenant	All user and guest accounts in your directory can use your application or API.
Accounts in any Azure AD directory	Multi-tenant	All users and guests with a work or school account from Microsoft can use your application or API. This includes schools and businesses that use Microsoft 365.
Accounts in any Azure AD directory and personal Microsoft accounts (such as Skype, Xbox, Outlook.com)	Multi-tenant	All users with a work, school, or personal Microsoft account can use your application or API. It includes schools and businesses that use Microsoft 365, as well as personal accounts that are used to sign into services like Xbox and Skype.

Implement application registrations

Demo – Register and application

After your app is registered:

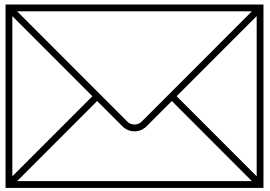
1. Add a redirect URI
2. Configure platform settings (je workflow OAuth)
3. Add credentials
4. Add a certificate and a client secret
5. Register the web API
6. Add a scope User.Read.All
   → Permission

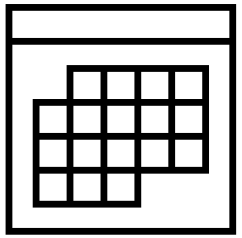
Configure application permissions

Application Permissions

Applications that integrate with Microsoft identity platform follow an authorization model that gives users and administrators control over how data can be accessed. Permissions for tasks like these can be controlled:

- Read a user's calendar
- Write to a user's calendar
- Send mail as a user





Permissions and Consent: Permission types

Delegated permissions

- Used by apps that have a signed-in user present
- Either the user or an administrator consents to the permissions that the app requests

User.Read

Application permissions

- Used by apps that run without a signed-in user present
- Only an administrator can consent to application permissions

User.Read.All

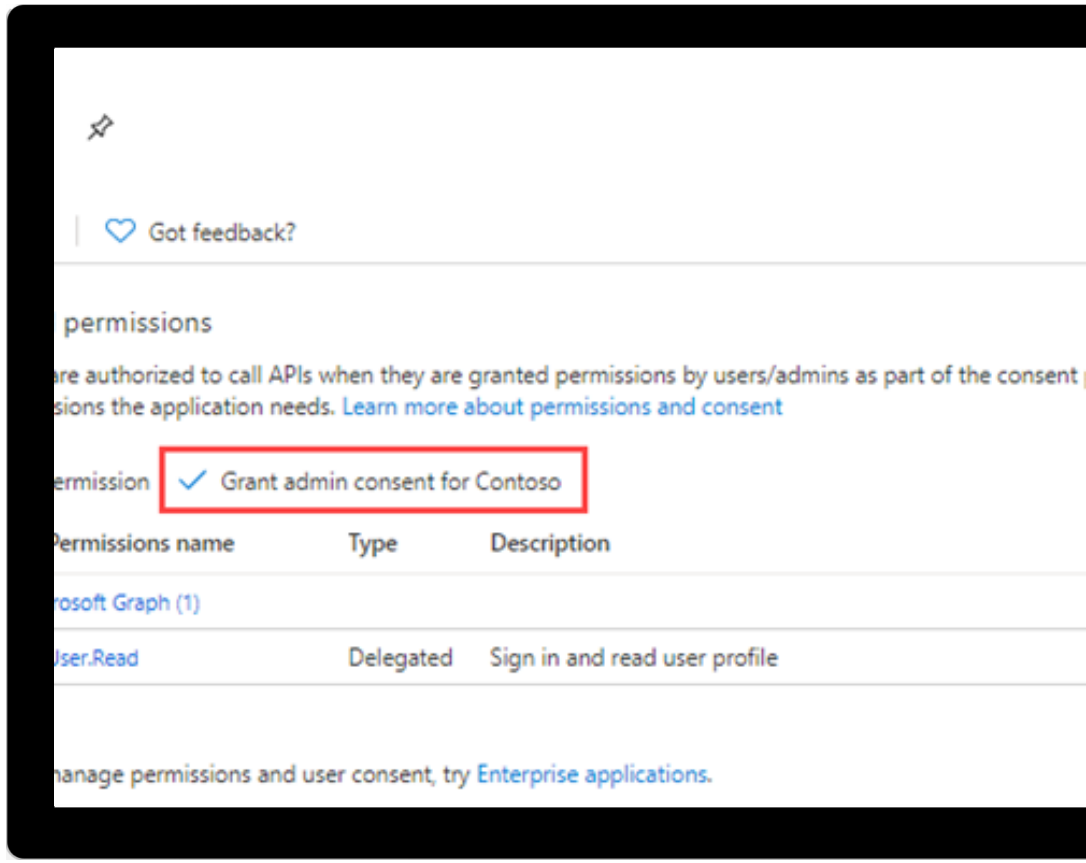
OpenID Connect Scopes

Exercise: Grant tenant-wide admin consent to an application

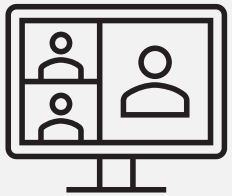
Grant admin consent in app registrations

For applications your organization has developed or for those that are registered directly in your Azure AD tenant, you can grant tenant-wide admin consent from App registrations in the Azure portal.

[Launch this Exercise in GitHub]



Implement application authorization



Application Roles

Application roles are used to assign permissions to users. You define app roles by using the Azure portal. When a user signs into the application, Azure AD emits a roles claim for each role that the user has been granted individually to the user and from their group membership.

There are two ways to declare app roles by using the Azure portal:

- App roles UI
  - Found on the App Registration / App menu
- App manifest editor

Demo – Add app roles to an application

Manage and monitor applications with App Governance

What does App Governance provide

- **Insights**: See a view of all the third-party apps for the Microsoft 365 platform in your tenant on a single dashboard. You can see all the apps' status and alert activities and react or respond to them.

- **Governance**: Create proactive or reactive policies for app and user patterns and behaviors and protect your users from using non-compliant or malicious apps and limiting the access of risky apps to your data.

- **Detection**: Be alerted and notified when there are anomalies in app activity and when non-compliant, malicious, or risky apps are used.

- **Remediation**: Along with automatic remediation capabilities, use remediation controls in a timely manner to respond to anomalous app activity detections.

Enable App Governance (Microsoft 365 Defender)

1. Ensure Office 365 is connected in Defender for Cloud Apps.
2. Ensure Office 365 Azure AD apps are enabled.
3. Go to your Defender for Cloud Apps portal – https://security.microsoft.com
4. Under Cloud app, select App Governance.
5. Select “Start trial,” and then select Save.

Verify integration with Defender for Cloud Apps is active, look for the app governance policies listed below to appear in Defender for Cloud Apps:

- Microsoft 365 OAuth app Reputation
- Microsoft 365 OAuth Phishing Detection
- Microsoft 365 OAuth App Governance

Summary

Now that you have reviewed this section, you should be able to:

- Plan your line of business application registration strategy
- Implement application registrations
- Configure application permissions
- Implement application authorization

Summary

Plan and Design Single Sign-on for Apps

- MDCA and ADFS application location
- App discover
- App management roles
- Add on-premises app management

Implement App Registration

- Design and App Registration strategy
- Register your applications
- Configure app permissions
- Assign app authorization

Roles Azure

MS Graph Perm

Implement and Monitor Enterprise Apps

- Consent settings
- Monitor enterprise applications
- Application collections
- Add on-premises app management

Connect-MgGraph MSAL

New-MgUser

MS Graph API

M365

New-AzureADUser ADAL deprecated

Labs

^3

Lab	Brief description	Length
17. App Discovery	Use Defender for Cloud Apps application discovery and enforce a restriction	15 minutes
18. App Access Policies	Configure app access policies in Defender for Cloud Apps	10 minutes
19. Register and application	Registering your application establishes a trust relationship between your app and the Microsoft identity platform.	10 minutes
20. Implement access management for apps.	Add an Enterprise app and assign your administrator account.	5 minutes
21. Grant tenant wide access to an app	For applications registered directly in your Azure AD tenant, grant tenant-wide admin consent from App registrations in the Azure portal.	10 minutes

End of presentation