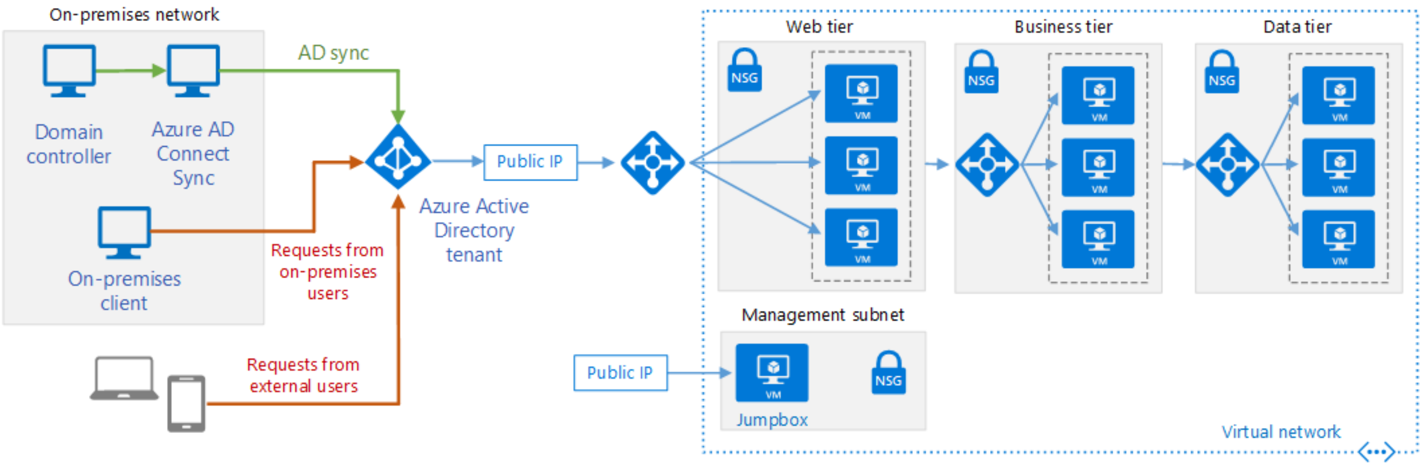
Integrate on-premises Active Directory domains with Azure Active Directory

11/28/2016 • 16 minutes to read • Contributors all

Azure Active Directory (Azure AD) is a cloud-based multi-tenant directory and identity service. This reference architecture shows best practices for integrating on-premises Active Directory domains with Azure AD to provide cloud-based identity authentication. Deploy this solution.



Download a Visio file of this architecture.

> ⓘ **Note**
>
> For simplicity, this diagram only shows the connections directly related to Azure AD, and not protocol-related traffic that may occur as part of authentication and identity federation. For example, a web application may redirect the web browser to authenticate the request through Azure AD. Once authenticated, the request can be passed back to the web application, with the appropriate identity information.

Typical uses for this reference architecture include:

- Web applications deployed in Azure that provide access to remote users who belong to your organization.
- Implementing self-service capabilities for end-users, such as resetting their passwords, and delegating group management. This requires Azure AD Premium edition.
- Architectures in which the on-premises network and the application's Azure VNet are not connected using a VPN tunnel or ExpressRoute circuit.

> ⓘ **Note**

Azure AD can authenticate the identity of users and applications that exist in an organization’s directory. Some applications and services, such as SQL Server, may require computer authentication, in which case this solution is not appropriate.

For additional considerations, see Choose a solution for integrating on-premises Active Directory with Azure.

Architecture

The architecture has the following components.

- **Azure AD tenant**. An instance of Azure AD created by your organization. It acts as a directory service for cloud applications by storing objects copied from the on-premises Active Directory and provides identity services.
- **Web tier subnet**. This subnet holds VMs that run a web application. Azure AD can act as an identity broker for this application.
- **On-premises AD DS server**. An on-premises directory and identity service. The AD DS directory can be synchronized with Azure AD to enable it to authenticate on-premises users.
- **Azure AD Connect sync server**. An on-premises computer that runs the Azure AD Connect sync service. This service synchronizes information held in the on-premises Active Directory to Azure AD. For example, if you provision or deprovision groups and users on-premises, these changes propagate to Azure AD.

  > ⓘ **Note**
  >
  > For security reasons, Azure AD stores user's passwords as a hash. If a user requires a password reset, this must be performed on-premises and the new hash must be sent to Azure AD. Azure AD Premium editions include features that can automate this task to enable users to reset their own passwords.

- **VMs for N-tier application**. The deployment includes infrastructure for an N-tier application. For more information about these resources, see Run VMs for an N-tier architecture.

Recommendations

The following recommendations apply for most scenarios. Follow these recommendations unless you have a specific requirement that overrides them.

Azure AD Connect sync service

The Azure AD Connect sync service ensures that identity information stored in the cloud is consistent with that held on-premises. You install this service using the Azure AD Connect software.

Before implementing Azure AD Connect sync, determine the synchronization requirements of your organization. For example, what to synchronize, from which domains, and how frequently. For more information, see Determine directory synchronization requirements.

You can run the Azure AD Connect sync service on a VM or a computer hosted on-premises. Depending on the volatility of the information in your Active Directory directory, the load on the Azure AD Connect sync service is unlikely to be high after the initial synchronization with Azure AD. Running the service on a VM makes it easier to scale the server if needed. Monitor the activity on the VM as described in the Monitoring considerations section to determine whether scaling is necessary.

If you have multiple on-premises domains in a forest, we recommend storing and synchronizing information for the entire forest to a single Azure AD tenant. Filter information for identities that occur in more than one domain, so that each identity appears only once in Azure AD, rather than being duplicated. Duplication can lead to inconsistencies when data is synchronized. For more information, see the Topology section below.

Use filtering so that only necessary data is stored in Azure AD. For example, your organization might not want to store information about inactive accounts in Azure AD. Filtering can be group-based, domain-based, organization unit (OU)-based, or attribute-based. You can combine filters to generate more complex rules. For example, you could synchronize objects held in a domain that have a specific value in a selected attribute. For detailed information, see Azure AD Connect sync: Configure Filtering.

To implement high availability for the AD Connect sync service, run a secondary staging server. For more information, see the Topology recommendations section.

Security recommendations

User password management. The Azure AD Premium editions support password writeback, enabling your on-premises users to perform self-service password resets from within the Azure portal. This feature should be enabled only after reviewing your organization's password security policy. For example, you can restrict which users can change their passwords, and you can tailor the password management experience. For more information, see Customizing Password Management to fit your organization's needs.

Protect on-premises applications that can be accessed externally. Use the Azure AD Application Proxy to provide controlled access to on-premises web applications for external users through Azure AD. Only users that have valid credentials in your Azure directory have permission to use the application. For more information, see the article Enable Application Proxy in the Azure portal.

Actively monitor Azure AD for signs of suspicious activity. Consider using Azure AD Premium P2 edition, which includes Azure AD Identity Protection. Identity Protection uses adaptive machine learning algorithms and heuristics to detect anomalies and risk events that may indicate that an identity has been compromised. For example, it can detect potentially unusual activity such as irregular sign-in activities, sign-ins from unknown sources or from IP addresses with suspicious activity, or sign-ins from devices that may be infected. Using this data, Identity Protection generates reports and alerts that enables you to investigate these risk events and take appropriate action. For more information, see Azure Active Directory Identity Protection.

You can use the reporting feature of Azure AD in the Azure portal to monitor security-related activities occurring in your system. For more information about using these reports, see Azure Active Directory Reporting Guide.

Topology recommendations

Configure Azure AD Connect to implement a topology that most closely matches the requirements of your organization. Topologies that Azure AD Connect supports include:

- **Single forest, single Azure AD directory**. In this topology, Azure AD Connect synchronizes objects and identity information from one or more domains in a single on-premises forest into a single Azure AD tenant. This is the default topology implemented by the express installation of Azure AD Connect.

  > ⓘ Note
  >
  > Don't use multiple Azure AD Connect sync servers to connect different domains in the same on-premises forest to the same Azure AD tenant, unless you are running a server in staging mode, described below.

- **Multiple forests, single Azure AD directory**. In this topology, Azure AD Connect synchronizes objects and identity information from multiple forests into a single Azure AD tenant. Use this topology if your organization has more than one on-premises forest. You can consolidate identity information so that each unique user is represented once in the Azure AD directory, even if the same user exists in more than one forest. All forests use the same

Azure AD Connect sync server. The Azure AD Connect sync server does not have to be part of any domain, but it must be reachable from all forests.

> ⓘ Note
>
> In this topology, don't use separate Azure AD Connect sync servers to connect each on-premises forest to a single Azure AD tenant. This can result in duplicated identity information in Azure AD if users are present in more than one forest.

- **Multiple forests, separate topologies**. This topology merges identity information from separate forests into a single Azure AD tenant, treating all forests as separate entities. This topology is useful if you are combining forests from different organizations and the identity information for each user is held in only one forest.

> ⓘ Note
>
> If the global address lists (GAL) in each forest are synchronized, a user in one forest may be present in another as a contact. This can occur if your organization has implemented GALSync with Forefront Identity manager 2010 or Microsoft Identity Manager 2016. In this scenario, you can specify that users should be identified by their *Mail* attribute. You can also match identities using the *ObjectSID* and *msExchMasterAccountSID* attributes. This is useful if you have one or more resource forests with disabled accounts.

- **Staging server**. In this configuration, you run a second instance of the Azure AD Connect sync server in parallel with the first. This structure supports scenarios such as:
  - High availability.
  - Testing and deploying a new configuration of the Azure AD Connect sync server.
  - Introducing a new server and decommissioning an old configuration.

    In these scenarios, the second instance runs in *staging mode*. The server records imported objects and synchronization data in its database, but does not pass the data to Azure AD. If you disable staging mode, the server starts writing data to Azure AD, and also starts performing password write-backs into the on-premises directories where appropriate. For more information, see Azure AD Connect sync: Operational tasks and considerations.

- **Multiple Azure AD directories**. It is recommended that you create a single Azure AD directory for an organization, but there may be situations where you need to partition information across separate Azure AD directories. In this case, avoid synchronization and password write-back issues by ensuring that each object from the on-premises forest appears in only one Azure AD directory. To implement this scenario, configure separate Azure AD Connect sync servers for each Azure AD directory, and use filtering so that each Azure AD Connect sync server operates on a mutually exclusive set of objects.

For more information about these topologies, see Topologies for Azure AD Connect.

User authentication

By default, the Azure AD Connect sync server configures password hash synchronization between the on-premises domain and Azure AD, and the Azure AD service assumes that users authenticate by providing the same password that they use on-premises. For many organizations, this is appropriate, but you should consider your organization's existing policies and infrastructure. For example:

- The security policy of your organization may prohibit synchronizing password hashes to the cloud. In this case, your organization should consider pass-through authentication.

- You might require that users experience seamless single sign-on (SSO) when accessing cloud resources from domain-joined machines on the corporate network.
- Your organization might already have Active Directory Federation Services (AD FS) or a third-party federation provider deployed. You can configure Azure AD to use this infrastructure to implement authentication and SSO rather than by using password information held in the cloud.

For more information, see Azure AD Connect User Sign-on options.

Azure AD application proxy

Use Azure AD to provide access to on-premises applications.

Expose your on-premises web applications using application proxy connectors managed by the Azure AD application proxy component. The application proxy connector opens an outbound network connection to the Azure AD application proxy, and remote users' requests are routed back from Azure AD through this connection to the web apps. This removes the need to open inbound ports in the on-premises firewall and reduces the attack surface exposed by your organization.

For more information, see Publish applications using Azure AD Application proxy.

Object synchronization

Azure AD Connect's default configuration synchronizes objects from your local Active Directory directory based on the rules specified in the article Azure AD Connect sync: Understanding the default configuration. Objects that satisfy these rules are synchronized while all other objects are ignored. Some example rules:

- User objects must have a unique *sourceAnchor* attribute and the *accountEnabled* attribute must be populated.
- User objects must have a *sAMAccountName* attribute and cannot start with the text *Azure AD_* or *MSOL_*.

Azure AD Connect applies several rules to User, Contact, Group, ForeignSecurityPrincipal, and Computer objects. Use the Synchronization Rules Editor installed with Azure AD Connect if you need to modify the default set of rules. For more information, see Azure AD Connect sync: Understanding the default configuration).

You can also define your own filters to limit the objects to be synchronized by domain or OU. Alternatively, you can implement more complex custom filtering such as that described in Azure AD Connect sync: Configure Filtering.

Monitoring

Health monitoring is performed by the following agents installed on-premises:

- Azure AD Connect installs an agent that captures information about synchronization operations. Use the Azure AD Connect Health blade in the Azure portal to monitor its health and performance. For more information, see Using Azure AD Connect Health for sync.
- To monitor the health of the AD DS domains and directories from Azure, install the Azure AD Connect Health for AD DS agent on a machine within the on-premises domain. Use the Azure Active Directory Connect Health blade in the Azure portal for health monitoring. For more information, see Using Azure AD Connect Health with AD DS
- Install the Azure AD Connect Health for AD FS agent to monitor the health of services running on on-premises, and use the Azure Active Directory Connect Health blade in the Azure portal to monitor AD FS. For more information, see Using Azure AD Connect Health with AD FS

For more information on installing the AD Connect Health agents and their requirements, see Azure AD Connect Health Agent Installation.

Scalability considerations

The Azure AD service supports scalability based on replicas, with a single primary replica that handles write operations plus multiple read-only secondary replicas. Azure AD transparently redirects attempted writes made against secondary replicas to the primary replica and provides eventual consistency. All changes made to the primary replica are propagated to the secondary replicas. This architecture scales well because most operations against Azure AD are reads rather than writes. For more information, see Azure AD: Under the hood of our geo-redundant, highly available, distributed cloud directory.

For the Azure AD Connect sync server, determine how many objects you are likely to synchronize from your local directory. If you have less than 100,000 objects, you can use the default SQL Server Express LocalDB software provided with Azure AD Connect. If you have a larger number of objects, you should install a production version of SQL Server and perform a custom installation of Azure AD Connect, specifying that it should use an existing instance of SQL Server.

Availability considerations

The Azure AD service is geo-distributed and runs in multiple datacenters spread around the world with automated failover. If a datacenter becomes unavailable, Azure AD ensures that your directory data is available for instance access in at least two more regionally dispersed datacenters.

> ⓘ **Note**
>
> The service level agreement (SLA) for Azure AD Basic and Premium services guarantees at least 99.9% availability. There is no SLA for the Free tier of Azure AD. For more information, see **SLA for Azure Active Directory**.

Consider provisioning a second instance of Azure AD Connect sync server in staging mode to increase availability, as discussed in the topology recommendations section.

If you are not using the SQL Server Express LocalDB instance that comes with Azure AD Connect, consider using SQL clustering to achieve high availability. Solutions such as mirroring and Always On are not supported by Azure AD Connect.

For additional considerations about achieving high availability of the Azure AD Connect sync server and also how to recover after a failure, see Azure AD Connect sync: Operational tasks and considerations - Disaster Recovery.

Manageability considerations

There are two aspects to managing Azure AD:

- Administering Azure AD in the cloud.
- Maintaining the Azure AD Connect sync servers.

Azure AD provides the following options for managing domains and directories in the cloud:

- **Azure Active Directory PowerShell Module**. Use this module if you need to script common Azure AD administrative tasks such as user management, domain management, and configuring single sign-on.
- **Azure AD management blade in the Azure portal**. This blade provides an interactive management view of the directory, and enables you to control and configure most aspects of Azure AD.

Azure AD Connect installs the following tools to maintain Azure AD Connect sync services from your on-premises machines:

- **Microsoft Azure Active Directory Connect console**. This tool enables you to modify the configuration of the Azure AD Sync server, customize how synchronization occurs, enable or disable staging mode, and switch the user sign-in mode. You can enable Active Directory FS sign-in using your on-premises infrastructure.

- **Synchronization Service Manager**. Use the *Operations* tab in this tool to manage the synchronization process and detect whether any parts of the process have failed. You can trigger synchronizations manually using this tool. The *Connectors* tab enables you to control the connections for the domains that the synchronization engine is attached to.
- **Synchronization Rules Editor**. Use this tool to customize the way objects are transformed when they are copied between an on-premises directory and Azure AD. This tool enables you to specify additional attributes and objects for synchronization, then executes filters to determine which objects should or should not be synchronized. For more information, see the Synchronization Rule Editor section in the document Azure AD Connect sync: Understanding the default configuration.

For more information and tips for managing Azure AD Connect, see Azure AD Connect sync: Best practices for changing the default configuration.

Security considerations

Use conditional access control to deny authentication requests from unexpected sources:

- Trigger Azure Multi-Factor Authentication (MFA) if a user attempts to connect from a nontrusted location such as across the Internet instead of a trusted network.

- Use the device platform type of the user (iOS, Android, Windows Mobile, Windows) to determine access policy to applications and features.

- Record the enabled/disabled state of users' devices, and incorporate this information into the access policy checks. For example, if a user's phone is lost or stolen it should be recorded as disabled to prevent it from being used to gain access.

- Control user access to resources based on group membership. Use Azure AD dynamic membership rules to simplify group administration. For a brief overview of how this works, see Introduction to Dynamic Memberships for Groups.

- Use conditional access risk policies with Azure AD Identity Protection to provide advanced protection based on unusual sign-in activities or other events.

For more information, see Azure Active Directory conditional access.

Deploy the solution

A deployment for a reference architecture that implements these recommendations and considerations is available on GitHub. This reference architecture deploys a simulated on-premises network in Azure that you can use to test and experiment. The reference architecture can be deployed with either with Windows or Linux VMs by following the directions below:

1. Click the link below to deploy the solution.

   Deploy to Azure

2. Once the link has opened in the Azure portal, you must enter values for some of the settings:

   - The **Resource group** name is already defined in the parameter file, so select **Create New** and enter `ra-aad-onpremise-rg` in the text box.
   - Select the region from the **Location** drop-down box.
   - Do not edit the **Template Root Uri** or the **Parameter Root Uri** text boxes.
   - Select **windows** or **linux** in the **Os Type** the drop-down box.
   - Review the terms and conditions, then click the **I agree to the terms and conditions stated above** checkbox.

- Click the **Purchase** button.

3. Wait for the deployment to complete.

4. The parameter files include hard-coded administrator user names and passwords, and you should immediately change both on all the VMs. Click each VM in the Azure portal then click on **Reset password** in the **Support + troubleshooting** blade. Select **Reset password** in the **Mode** drop down box, then select a new **User name** and **Password**. Click the **Update** button to persist the new user name and password.