Extend On-Premises Active Directory to Microsoft Azure

An Overview

However, there IS a special order in which things need to happen in order to make it work properly.  The first order of business is the deployment of a virtual network in Azure.  Once the virtual network is deployed in Azure and secured with a Network Security Group, a VPN connection needs to be made between the on-prem network and the Azure virtual network.  With the VPN connectivity in place, the DNS configuration of the virtual network needs to be configured to point back to the on-prem DNS servers (since they know how to resolve the local Active Directory domain).

After the configuration of the virtual network and VPN in Azure are complete, a virtual machine needs to be provisioned in Azure.  Because the virtual network points to the on-prem DNS, the virtual machine will inherit those DNS servers when it comes up.  The virtual machine should be deployed with no public IP address and it should have a separate 20-40GB data drive with caching turned off.  This is a Microsoft-supported configuration and prevents USN Rollbacks.

Once the virtual machine is provisioned, it should be joined to the on-prem Active Directory just like any other member server.  After making the virtual machine a member server, Active Directory Domain Services should be installed.  The server can then be promoted to a domain controller for the on-prem AD.  Active Directory should be stored on the data disk that was provisioned, not the system drive.

At this point, Active Directory Sites and Services should be configured to reflect the new Azure network as a new site.  Once this configuration is complete, the on-prem Active Directory is successfully extended to Azure.

## Extending On-Prem Active Directory to Azure: Step By Step

**Deploy and Secure the Azure Virtual Network**

Go into your ARM portal and deploy a virtual network with an address space that does not overlap your on-premises network space.  For example, you might deploy an address space of 10.0.0.0/16 in Azure if your on-prem address space is 192.168.0.0/16.  While deploying the virtual network in Azure, you will be prompted for a default subnet and address range for it.  Do not use the entire address space when creating your subnet.  Only use what you need.  In this example, you might use 10.0.1.0/24 as your subnet range.  Once the virtual network is deployed, be sure to deploy a Network Security Group in Azure that contains an inbound rule that allows RDP traffic only from your network.

**Establish VPN Connectivity**

Depending on your on-prem VPN device, you might need to deploy a route-based VPN or a policy-based VPN in Azure.  If your on-prem VPN device supports IKE2, you will want to deploy a route-based VPN.  Otherwise, deploy a policy-based VPN.  Most devices that I have encountered to date only support IKE1.

Deploying an Azure VPN requires a few different pieces.  It requires a Gateway Subnet on the Azure virtual network, a Virtual Network Gateway with a public IP address, a Local Gateway that represents the on-prem network device, and a Connection.

Start your VPN deployment by provisioning a Gateway Subnet on your virtual network.  A Gateway Subnet should consist of a /27 range.  To make things easy for yourself, click here to download the Gateway Subnet Calculator.  The Gateway Subnet Calculator is a PowerShell script that takes your virtual network address space and uses it to automatically calculate an address range for your Gateway Subnet.  It is a pretty handy tool.  Just follow the instructions that come with it.  If you stick with the examples from earlier in this article, the calculator will produce a Gateway Subnet range of xxx.xxx.xxx.xxx.

Once you have the Gateway Subnet configured, go ahead and deploy a Virtual Network Gateway and specify your Gateway Subnet during deployment.  During configuration, specify the address space of your Azure virtual network in the "address spaces" section so your domain controller is accessible over the VPN when it is complete.

The deployment of the Virtual Network Gateway will often take up to 45 minutes to complete once it has been submitted so go find something productive to do while that is happening.  Eventually the Virtual Network Gateway deployment will complete and it will show a public IP address.  Jot it down because you will need it to provision the VPN shortly.

With the Virtual Network Gateway provisioned, go ahead and provision a Local Network Gateway.  Specify the public IP address of your on-prem VPN device when configuring the gateway.  Additionally, add any additional underlying address spaces in the "address spaces" section.  These address spaces reflect networks that you wish to access over the VPN.  Talk to your network admin if you are not sure of what address spaces are in use in your environment.

At this point, you should have your Virtual Network Gateway configured, along with your Local Network Gateway.  With both gateways configured, you can deploy the actual connection by clicking the "+" sign in your ARM dashboard and searching for "Connection".  You will be prompted for a Virtual Gateway, a Local Gateway, and a shared key.  Select the gateways you have provisioned and enter a shared key (your network admin will need to use the same key).  Once you do this, the configuration on the Azure side is complete.  Go ahead and provide the public IP of your Virtual Network Gateway to your network administrator so he can configure the VPN on his side of the house (the local VPN device).  He/she will also need the address space of your virtual network and the shared key you chose so provide that info as well.

Once the local VPN device is configured, the status of the VPN in Azure should switch to Succeeded/Connected

**Deploy the Virtual Machine**

Now that the connectivity between Azure and your on-prem network is up and running, you can go ahead and deploy the virtual machine that will eventually become your domain controller in Azure.  Unless you are upgrading your AD at the same time (not recommended), be sure that the OS you choose matches your current domain controllers.  When deploying the VM, go with a reasonable size.  I have found Standard A2\_V2 size to be sufficient in most cases for small environments.

While deploying the virtual machine, you will be prompted to assign a public IP.  You can go ahead and assign it a public IP for now, but you will eventually need to remove it since you don't want a DC to be accessible publicly.  However, assuming your VPN connectivity is good, you should really just forego the assignment of a public IP and configure the VM via RDP from your local network over the VPN using its internal IP address instead.  Additionally, choose the option to deploy an Availability Set as well.  Doing so will allow you to later add a second domain controller into the set to ensure at least one of the DCs in Azure is always available.  Change the assigned internal IP address from dynamic to static.  After deploying the virtual machine, provision and attach a data disk and set the Host Cache Preference to None.  This data disk will house Active Directory.

**Configure the Virtual Machine**

Once the new virtual machine comes up, RDP to it and configure it as you normally would.  Apply any needed patches, set the time zone, etc.  If necessary, go into disk manager on the virtual machine, scan for the data disk that you provisioned earlier, and make sure it is formatted and a volume exists on it.

After configuring the server, reboot if necessary and then join it to the on-prem Active Directory just as you would any other server.  You can then go ahead and install Active Directory Domain Services and promote it to a Domain Controller.  Make sure you've removed the public IP first.  Be sure to specify the data disk that you provisioned as the path for Active Directory.  The installation process is going to complain about your VM having a NIC with a DHCP-assigned address.  You can ignore this.  Yes, your OS sees a DHCP configuration; however, the IP is never going to change because it was set to static in Azure.

**Monitoring and reporting in Azure**

Azure offers many services that together provide a comprehensive solution for collecting, analyzing, and acting on telemetry from your applications and the Azure resources that support them. These services can also monitor critical on-premises resources to provide a hybrid monitoring environment.

**Azure Monitor** provides a single unified hub for all monitoring and diagnostics data in Azure. You can use it to get visibility across your resources. With Azure Monitor, you can find and fix problems, optimize performance, and understand customer behavior.

* **Data collection**: Azure Monitor collects data from [various data sources](https://learn.microsoft.com/en-us/azure/azure-monitor/overview#what-data-can-azure-monitor-collect), including: Application, Container, Guest operating system, Azure resource, Azure subscription, Azure tenant, and Azure resource changes. Additionally, Azure Monitor can collect log data from any REST client using the [Data Collector API](https://learn.microsoft.com/en-us/azure/azure-monitor/logs/data-collector-api).
* **Insights**: Availability, performance, usage, and health of your web applications are monitored using [Application Insights](https://learn.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview). Further, you can use the insights features of Azure Monitor to monitor your:
  + [Applications](https://learn.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview)
  + [Containers](https://learn.microsoft.com/en-us/azure/azure-monitor/containers/container-insights-overview)
  + [Virtual machines](https://learn.microsoft.com/en-us/azure/azure-monitor/vm/vminsights-overview)
  + [Networks](https://learn.microsoft.com/en-us/azure/networking/network-monitoring-overview)
* **Visualization**: Visualizing your monitoring data will greatly help you get an overview of the current posture of your cloud real estate. Make use of visualizations with built-in or custom charts and tables, workbooks, dashboards, or Power BI.
  + Read more about [best practices for analyzing and visualizing data](https://learn.microsoft.com/en-us/azure/azure-monitor/best-practices-analysis).
* **Response**: An effective monitoring strategy often requires an actionable response to critical events in the collected data. You can automate actions by using the built-in [Alerts](https://learn.microsoft.com/en-us/azure/azure-monitor/overview#alerts) or [Autoscale](https://learn.microsoft.com/en-us/azure/azure-monitor/overview" \l "autoscale) capabilities.

**Azure Service Health** provides a personalized view of the health of your Azure services and regions. Information about active issues is posted to Azure Service Health to help you understand the impact on your resources. Regular updates keep you informed as problems are resolved.

Set up Service Health alerts to notify you when service issues, planned maintenance, or other changes might affect the Azure services and regions you use. You can also view planned maintenance events that could affect the availability of your resources.

Azure Service Health includes:

* **Azure status:** A global view of the health of Azure services.
* **Service health:** A personalized view of the health of your Azure services.
* **Resource health:** A deeper view of the health of each of your resources.

**Azure Advisor** is a free, personalized cloud consultant that helps you follow and implement best practices for Azure deployments. Azure Advisor analyzes your resource configuration and usage telemetry and then recommends solutions that can help optimize your environment. The recommendations are divided into the following categories:

* **Reliability:** To improve the continuity of your business-critical applications. Recommendations might include adding virtual machines to an availability set or adding geo-redundant endpoints.
* **Security:** To detect threats and vulnerabilities that might lead to security breaches. Recommendations might include applying Azure Disk Encryption or enabling network security groups.
* **Performance:** To improve the speed of your applications. Recommendations might include boosting SQL query performance by creating indexes or reconfiguring your Traffic Manager settings.
* **Cost:** To optimize and reduce your overall Azure spending. Recommendations might include resizing or shutting down underused virtual machines or switching to Azure Reservations to lower the total cost of ownership.
* **Operational excellence:** To improve process and workflow efficiency and manageability. Recommendations might include setting up and enforcing Azure Policy rules, repairing invalid log alert rules, and configuring Service Health alerts.

Recommendations in Azure Advisor are based on the resources you deploy and your actions in Azure. You can check Advisor regularly for the latest recommendations.

Additionally, the five categories in Azure Advisor directly relate to the pillars of the [Azure Well-Architected Framework](https://learn.microsoft.com/en-us/azure/architecture/framework/), a set of guiding tenets to optimize and improve the quality of your workloads.

**Microsoft Defender** for Cloud also plays an integral part in your monitoring strategy. Defender for Cloud can help you monitor the security of your machines, networks, storage, data services, and applications. It provides advanced threat detection using machine learning and behavioral analytics to help identify active threats targeting your Azure resources. Defender for Cloud also provides threat protection that blocks malware or other unwanted code and reduces the surface area exposed to brute force and other network attacks.

When Defender for Cloud identifies a threat, it triggers a security alert with steps you need to take to respond to an attack. It also provides a report with information about the threat that was detected.

Microsoft Defender for Cloud is offered in two flavors:

* **Free**. Offers the secure score and related tools.
* **Enhanced security features**. Enables all of the Microsoft Defender plans to provide a range of security benefits for all your resources in Azure, hybrid, and multicloud environments.

**Microsoft Sentinel** delivers intelligent security analytics and threat intelligence across the enterprise. You can use the capabilities from Microsoft Sentinel to monitor security events at scale. Utilize sophisticated workbooks to build customized reports and dashboards with the correct data at your fingertips.

Microsoft Sentinel provides two critical cloud-native solutions:

* Security information and event management (SIEM)
* Security orchestration, automation, and response (SOAR)

By connecting data sources to Microsoft Sentinel using the built-in connectors, you can also widen your monitoring lens to a broader security and application ecosystem for non-Microsoft solutions.

**Get started:**

1. Review the [Microsoft Sentinel pricing](https://azure.microsoft.com/pricing/details/microsoft-sentinel/) to understand the cost impact.
2. Learn how to [onboard Microsoft Sentinel](https://learn.microsoft.com/en-us/azure/sentinel/quickstart-onboard) and set up data connectors to monitor and protect your environment.

After you've connected data sources using the data connectors, you can immediately start using the platform's capabilities.