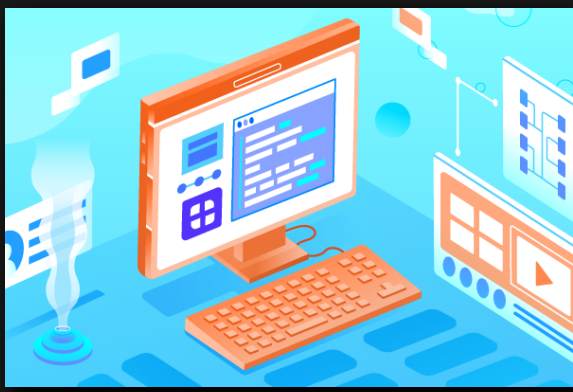
Azure Administrator Associate (AZ-103)

Course Navigation

- Manage Azure Subscriptions and Resources
- Implement and Manage Storage
- Deploy and Manage Virtual Machines
- Configure and Manage Virtual Networks
- Manage Identities
- Exam Preparation



Linux Academy

Manage Azure Subscriptions and Resources

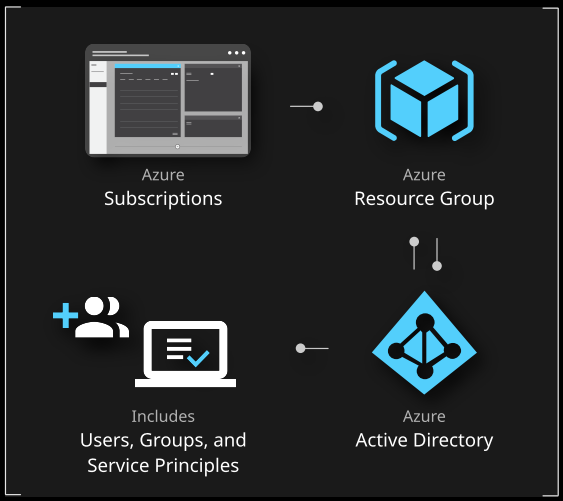
Azure Subscriptions

Course Navigation

An Azure subscription is a **logical unit of Azure services** that is linked to an Azure account. Subscriptions help you organize access to cloud services resources.

Azure Accounts

An Azure account is an identity in Azure Active Directory (AD) or a directory that is trusted by Azure AD, such as a work or school organization.



Next

Back to Main

Linux Academy

There are three main roles in an Azure classic subscription:

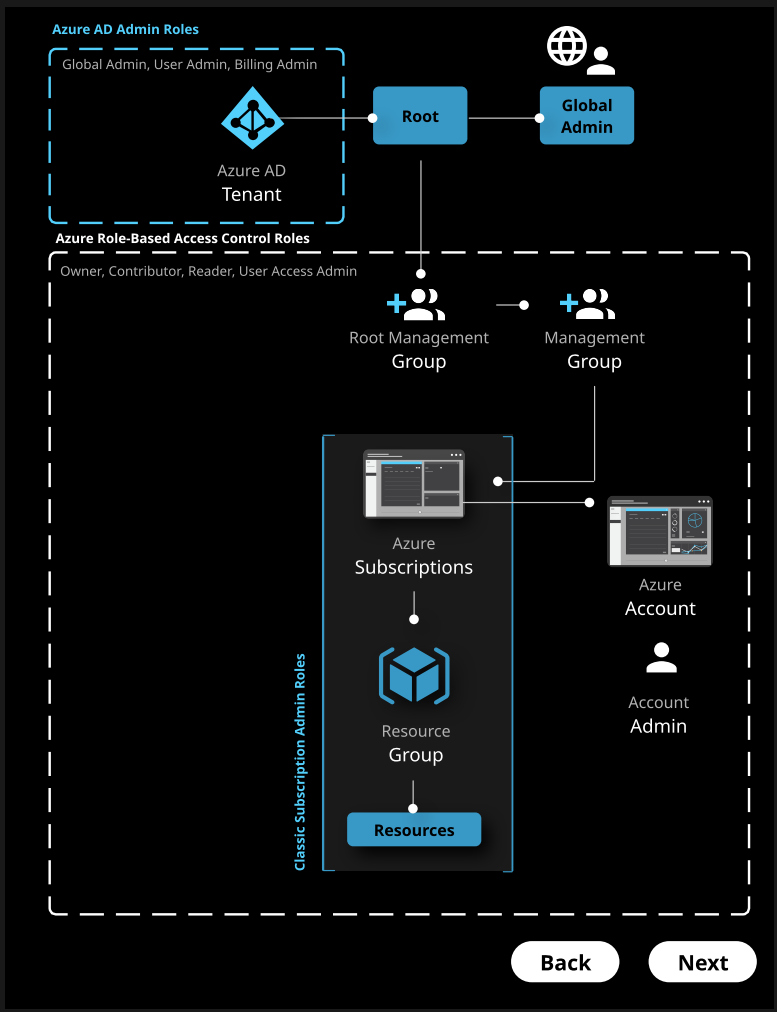
1. **Account Administrator** (one per Azure account): Authorized to **access** the account center.
2. **Service Administrator** (one per Azure subscription): Authorized to **access** the Azure portal for all subscriptions in an account. This role has **control** over all services in the subscription.
3. **Co-Administrator** (up to 200 per subscription): Same as the **Service Administrator**, but **can't change** the association of subscriptions to Azure directories.

See Diagram

Manage Azure Subscriptions and Resources

Azure Subscriptions

Course Navigation

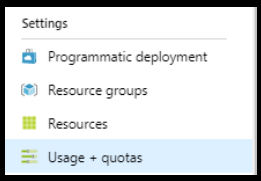


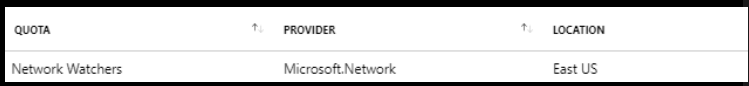
Back to Main

Linux Academy

Check Resource Limits and Quota

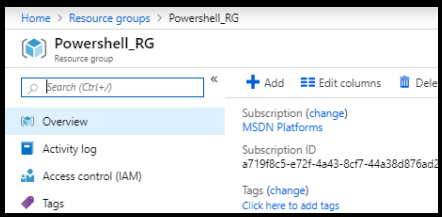
Azure provides the ability to see the **number** of each network resource type you've deployed in your subscriptions and what your **subscription limits** are. The ability to view resource usage against limits is helpful to track current usage and plan for future use.





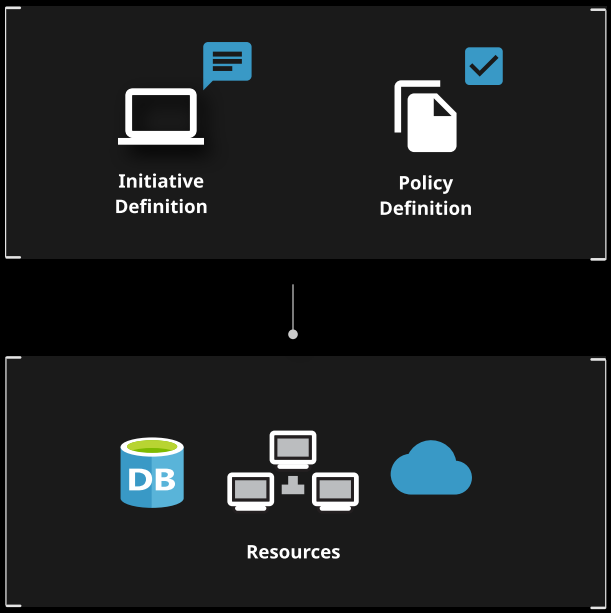
Resource Tags

You can **apply tags** to your Azure resources to logically organize them by categories. Each tag consists of a **name and value**.



Azure Policy

Azure Policy is a service for **creating, assigning, and managing policies**. These policies enforce different rules over your resources so those resources **stay compliant** with your corporate standards and service level agreements. Azure Policy does this by running **evaluations** of your resources and scanning for those not compliant with your policies.



Implementing Azure Policy

To implement Azure policies, follow the steps below:

1. A policy definition expresses what to evaluate and what actions to take.

Browse Policy Definitions

2. An initiative definition is a set of policy definitions to help track your compliance state for a larger goal.

Create Initiative Definition

3. You can limit the scope of the initiative definition to management groups, subscriptions, or resource groups.

Scope the Initiative Definition

4. Once the initiative definition is assigned, you can evaluate the state of compliance for all your resources.

Review Evaluation and Manage Exclusions

Back Next

Course Navigation

Manage Azure Subscriptions and Resources
- Azure Subscriptions
- Resource Utilization and Consumption
- Resource Groups
- Role-Based Access Control

Implement and Manage Storage

Deploy and Manage Virtual Machines

Configure and Manage Virtual Networks

Manage Identities

Action Groups enable you to configure a list of actions to take when the alert is triggered. An action group ensures the same actions are taken each time an alert is triggered. The action types you can select when defining the group are listed below.

1. **Select Email/SMS/PUSH/Voice** — Provides the ability to send email, SMS, push notifications, or a voice call
2. **Logic App, Webhook, IT Service Management, Function** — Run a Logic App; Deploy a Webhook; Integrate with an IT management service; Run a function app
3. **Automation Runbook** — Run an Azure Automation runbook

Next

Back to Main

Manage Azure Subscriptions and Resources

Analyze Resource Utilization and Consumption

Course Navigation

Azure Monitor enables **core monitoring** for Azure Services by **collecting metrics, activity logs, and diagnostic logs.** This is very useful for running diagnostics and troubleshooting multiple resources within Azure. The key capabilities of Azure Monitor include:

1

Monitor and Visualize Metrics

Metrics are numerical values available from Azure resources that help you understand the health and performance of your system.

2

Query and Analyze Logs

Activity logs, diagnostic logs, and telemetry are monitoring solutions which can provide useful information through analytic queries.

3

Setup Alerts and Actions

Alerts notify you of critical conditions and can take automated corrective actions. Triggers for alerts can be based on metrics or logs.

Back Next

Back to Main

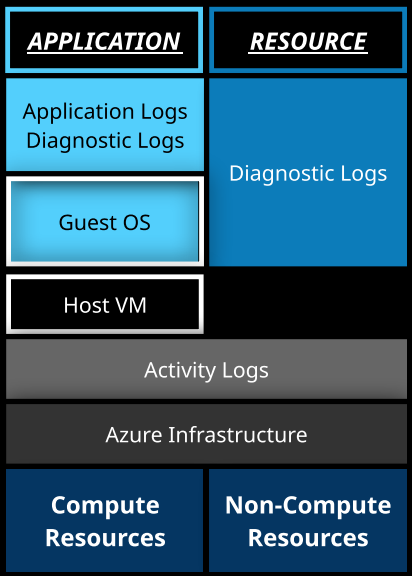
Linux Academy

Azure Monitor Diagnostic Logs

Azure Monitor diagnostic logs are logs provided by the Azure service that give useful data about the operation of Azure resources and services. The logs are constantly updating in real time to provide an accurate assessment of what is going on in the infrastructure.

Azure Monitor provides two types of diagnostic logs:

1. **Tenant logs** contain activity that occurs at the tenant level but is outside of the Azure subscription.
2. **Resource logs** contain information produced from Azure services which deploy resources in Azure.



Azure Monitor Alerts

Azure Monitor alerts can be configured to **notify you** or your team when your resources are **performing** at a **predetermined level** or if a detrimental event has occurred.

There are many **benefits** of Azure Monitor Alerts.

Better notification
All new alerts use action groups.

Unified experience
All alert metrics and logs are in one place.

View alerts in Portal
You can see alerts in your subscription.

Separation of Fired Alerts and Rules
Alert rules and fired alerts are differentiated. This keeps the operational and configuration views separate.

Better Workflow
An improved experience that guides you through the process of creating an alert.

Monitoring

Alerts

Course Navigation

Manage Azure Subscriptions and Resources
- Azure Subscriptions
- Resource Utilization and Consumption
- Resource Groups
- Role-Based Access Control

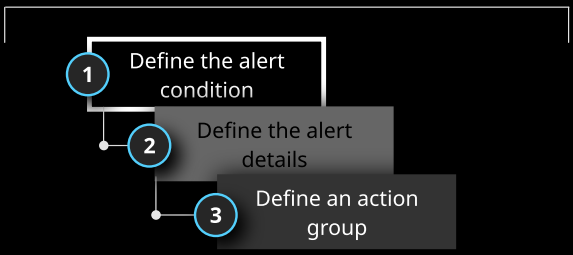
Implement and Manage Storage

Deploy and Manage Virtual Machines

Configure and Manage Virtual Networks

Manage Identities

Create Alert Rules



Creating an alert is a three-step process as shown above.

A **Define the alert condition including the following elements:**
- **Target selection:** i.e. storage account
- **Alert criteria**: i.e. used capacity
- **Alert logic:** i.e. an action is triggered once the disk space capacity has exceeded a specified size.

B **Define the alert details including the following elements:**
- **Alert rules name**
- **Description**
- **Severity**: A level ranging from severity 0 to severity 4.

C **Defining the action group**

Create an action group to either notify you or your team or take automated actions. Notifications can be delivered by email, text message, or both. Automated actions are performed using webhooks and runbooks.

Back
Next

Back to Main

Azure Metrics

Metrics are **numerical values** that provide details about your Azure Resource at a **specific time** or over a **specified range**. This information is collected on a regular basis and has **built-in alert options** which can be initiated programmatically.

This list provides some **examples** of how to use Azure Metrics.

Analyze
Metrics Explorer is used to gather information from different resources for analysis.

Visualize
With Metrics Explorer you can generate a chart for easier analysis and create a workbook to combine data.

Alert
Alerts can be configured to notify teams or individuals for specific events or triggers.

Automate
You can use auto scaling to adjust resources based on preset metric values.

Retrieve
Metrics data can be obtained using PowerShell cmdlets, Rest API, and CLI.

Export
Data can be sent to logs for analysis, to Azure Event Hubs, or routed to an external system outside of Azure.

Archive
Metrics can be kept for 93 days. Diagnostic logs can be routed to Log Analytics and configured to have a minimum retention of 30 days. Activity log entries are stored for 90 days.

Improvements for Metrics

Improved Latency	New metric alerts can run as frequently as every minute.
Support for Multi-Dimensional Metrics	You can set an alert on dimensional metrics to monitor an interesting segment of the metric.
More Control Over Metric Conditions	Youcandefinericheralertrulesthat supportmonitoringwithimproved capabilities.
Combined Monitoring of Multiple Metrics	You can monitor multiple metrics with a single rule.
Metrics From Logs (Preview)	You can extract some data going into Log Analytics and convert it into Azure Metrics. This can be used for alerts like other metrics.

Signals are emitted by the target resource and can be of several types. Supported signal types include Azure Metrics, Activity Log, and Application Insights.

Creating a Baseline for Resources

A performance baseline is your current average and is used to compare to future performance levels. Once a baseline has been determined you can use **Metric Alerts with Dynamic thresholds** to monitor your resource performance.

Metric Alerts with Dynamic Thresholds uses **machine learning** to analyze historic data in order to give suggestions regarding possible service issues.

Monitoring for Unused Resources with Azure Advisor

Azure Advisor is a personalized cloud consultant that helps you follow best practices to optimize your Azure deployments. It is also useful tracking under-utilized resources. It reviews your resource configuration and usage telemetry to provide advice on how to improve expenses, efficiency, performance, availability, and security of your Azure resources.

The **Advisor Cost Recommendations** page can assist you in optimizing and shrinking your total Azure spending by identifying idle and underutilized resources.

The recommendations are divided into the following 4 categories.

1. **Availability**
   To ensure and improve the continuity of your business critical applications.
2. **Security**
   To detect threats and vulnerabilities that might lead to security breaches.
3. **Costs**
   To optimize and reduce your overall spending.
4. **Performance**
   To improve the speed of your applications.

Manage Azure Subscriptions and Resources

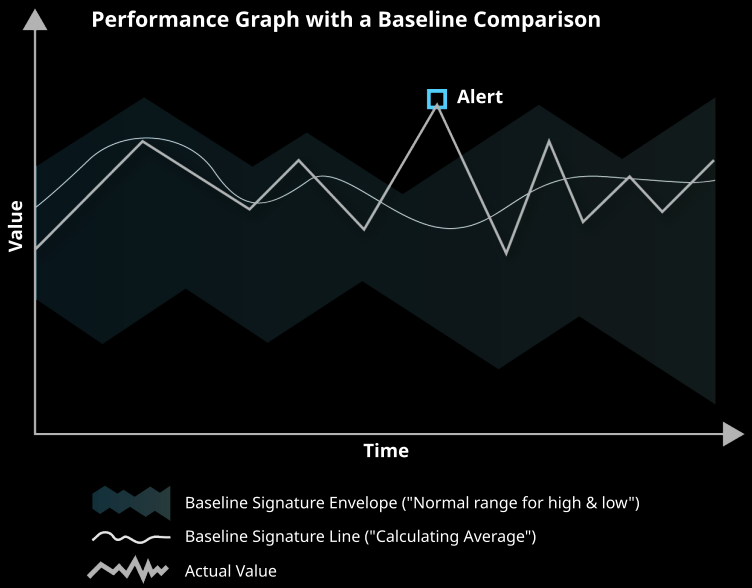
Analyze Resource Utilization and Consumption

Course Navigation

Creating a Baseline for Resources

A performance baseline is your current average and is used to compare to future performance levels. Once a baseline has been determined you can use **Metric Alerts with Dynamic thresholds** to monitor your resource performance.

Metric Alerts with Dynamic Thresholds uses **machine learning** to analyze historic data in order to give suggestions regarding possible service issues.



Back Next

Back to Main

Linux Academy

Manage Azure Subscriptions and Resources

Analyze Resource Utilization and Consumption

Course Navigation

Creating a Baseline for Resources

A performance baseline is your current average and is used to compare to future performance levels. Once a baseline has been determined you can use **Metric Alerts with Dynamic thresholds** to monitor your resource performance.

Metric Alerts with Dynamic Thresholds uses **machine learning** to analyze historic data in order to give suggestions regarding possible service issues.

3 Reasons to Use Dynamic Type

Scalable Alerting
Very useful when handling multiple resources across multiple subscriptions. Hundreds of metric series can be created at a time.

Smart Metric Pattern Recognition
Patterns in the metrics can be identified, causing the alerting to adapt and deviate as necessary.

Intuitive Configuration
Setting up metrics is an easy and user-friendly experience. There's no need for in-depth knowledge.

See Chart Here

Monitoring Spending and Reporting on Spending

In the move from on-premise computing to cloud-hosted services, tracking and estimating service usage and related costs are significant concerns. **It is important** to get an accurate estimate on what new resources will cost to run monthly and project how the billing will look for a given month based on the current spending. **Azure provides a set of Billing Rest API's** that give access to resource consumption and metadata information for Azure subscriptions.

Pricing calculator Provides estimates in all areas of Azure including compute, networking, storage, web, and databases	**Billing Alert Service** Provides ability to create alerts to send email when you approach spending limits.	**Cost Analysis** Supports different kinds of Azure account types and useful for exploring and analyze your organization's costs.
Customize cost views There are 4 built-in views: accumulated costs, daily costs, cost by service, and cost by resource.	**Download Reports** You can download information from cost analysis to generate a CSV file.	**Cost Analysis Prerequisites** Read access to billing account, department, enrollment account, management group, subscription, or resource group.

Manage Azure Subscriptions and Resources

Analyze Resource Utilization and Consumption

Course Navigation

Utilizing Log Search Query Functions

Log Analytics helps you collect, correlate, search, and act on log and performance data generated by operating systems and applications. It gives you real-time operational insights using integrated search and custom dashboards to readily analyze millions of records across all your workloads.

A

Log Analytics

Most of your interaction with **Log Analytics** will be through the **OMS portal**. The portal supports constructing queries to analyze collected data, customizing dashboards with graphical views, and provides additional functionality and analysis tools.

B

Connected Sources & Data Sources

Connected sources are the computers and other resources that generate data collected by **Log Analytics**.

Data sources are the different kinds of data collected from each **connected source.**

C

Query

Log Analytics provides a **query** syntax to quickly retrieve and consolidate data in the repository. You can create and save log searches to directly analyze data in the **OMS portal** or have log searches run automatically to create an alert if the results of the query indicate a predefined important condition.

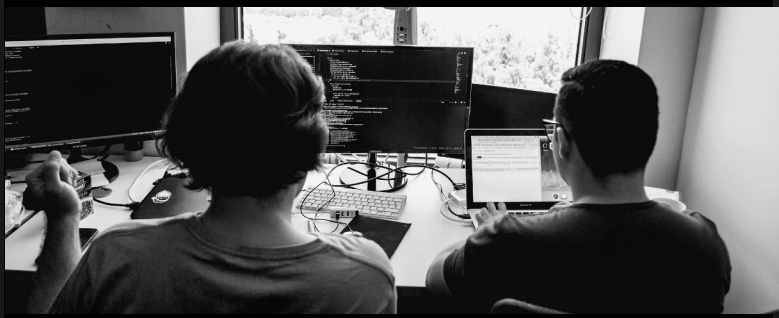
Workspace
To get started with Log Analytics you need to add a workspace.

Back Next

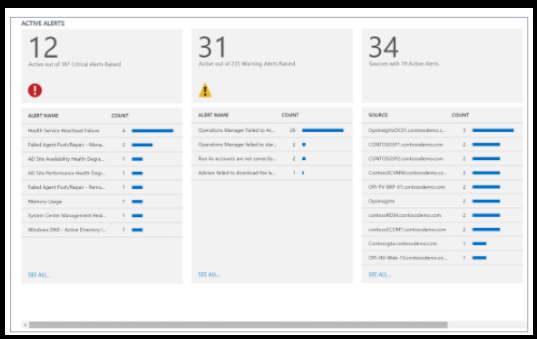
Back to Main

Linux Academy

Viewing Alerts in Log Analytics (Alert Management)



Alert Management helps you view your operations manager and Log Analytics log alerts across your entire environment. This is an effective tool which helps you centralize alerts and puts you in a better position to identify the cause of issues within your system.



Using Azure Policy for Resource Groups

Resource groups at their **simplest definition** are a container for multiple resources. Resources need to be deployed to a new or existing resource group.

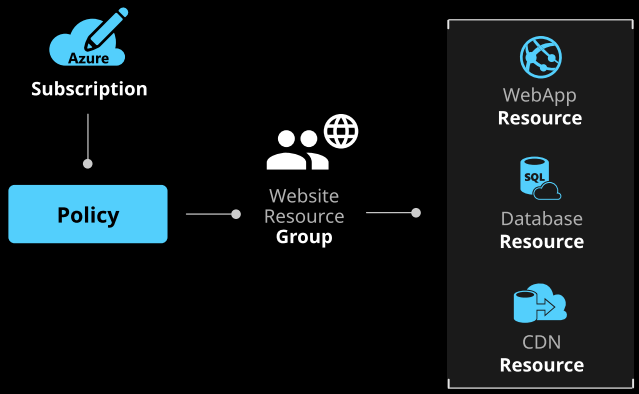
Rule 1:
Resources can only exist in one resource group.

Rule 2:
Resource groups cannot be renamed.

Resource Group Rules

Rule 3:
Resource groups can have resources of many different types called services.

Rule 4:
Resource groups can have resources from many different regions.



Next

Back to Main

Configuring Resource Locks

A common concern with resources provisioned in Azure is the ease with which they can be deleted. Resource Manager locks help with those concerns. There are two types of locks:

A

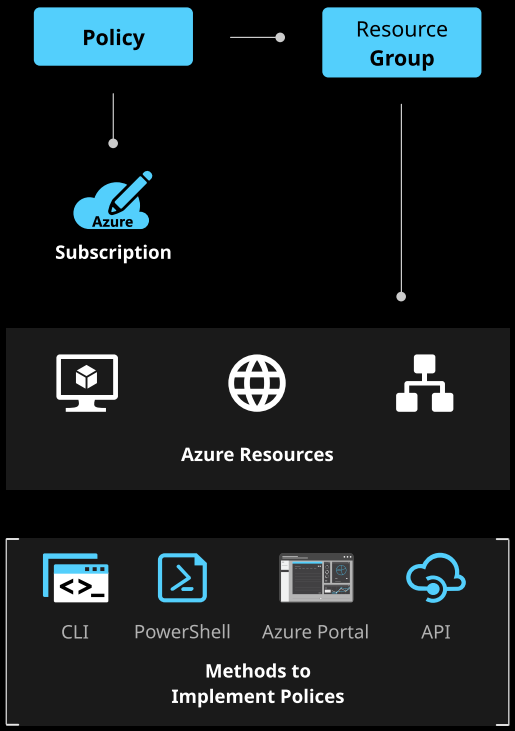
Read-only lock:
Prevents any change to the resource.

B

Delete lock:
Prevents the deletion of the resource.

Only Owner and User Access Administrator roles can create or delete management locks.

Configuring Resource Policies and Removing Resource Groups



Identifying Audit Requirements (Azure Security Center)

Azure Security Center is a built-in security management system that protects the data center within the Azure infrastructure as well as your on-premise environment, provided it's configured properly. **With the integration of Security Center in the Azure portal, it's easy to audit your environment**. You can do this to determine if your system is compliant with industry standards and your organizations specific security requirements. **Policies help with this effort.**

Azure Security Center addresses the three most urgent security challenges as follows:

Implementing and Setting Tags on Resource Groups

Tags are very useful in helping with the logical organization of Azure resources. **Each tag consists of a name-value pair**. You will need to be aware of the following limitations.

A. Not all resource types support **tags**.

B. VM and VM scale sets are **limited to 2048 characters** for all tag names and values.

C. Each resource or resource group can have a **max of 15** tag name-value pairs.

D. Tags applied to resource groups are **not inherited by resources** within the resource group.

E. **Tag names** are limited to 512 characters. **Tag values** are limited to 256 characters.

F. Tags can't be applied to **classic resources.**

G. Tag names can't contain **these characters**:
< > % \?/

Identifying Audit Requirements (Azure Security Center)

Azure Security Center is a built-in security management system that protects the data center within the Azure infrastructure as well as your on-premise environment, provided it's configured properly. **With the integration of Security Center in the Azure portal, it's easy to audit your environment**. You can do this to determine if your system is compliant with industry standards and your organizations specific security requirements. **Policies help with this effort.**

Azure Security Center addresses the three most urgent security challenges as follows:

3.

Get Secure Faster

Security Center operates faster then traditional threat assessment and detection systems. It utilizes machine learning with live data to help provide secure automated provisioning and built-in protection with Azure services.

Identifying Audit Requirements (Azure Security Center)

Azure Security Center is a built-in security management system that protects the data center within the Azure infrastructure as well as your on-premise environment, provided it's configured properly. **With the integration of Security Center in the Azure portal, it's easy to audit your environment**. You can do this to determine if your system is compliant with industry standards and your organizations specific security requirements. **Policies help with this effort.**

Azure Security Center addresses the three most urgent security challenges as follows:

2.

Protect Against Threats

Security Center reviews your workload and provides useful preventative advice on how to secure your workload. It also provides threat detection notifications.

Implementing and Setting Tags on Resource Groups

PowerShell Commands and Descriptions

1. 

   See the existing tags for a resource group.

```
(Get-AzResourceGroup -Name examplegroup).Tags
```

2. 

   Add tags to a resource group without changing existing tags.

```
Set-AzResourceGroup -Name examplegroup -Tag @{ Dept="IT";
Environment="Test" }
```

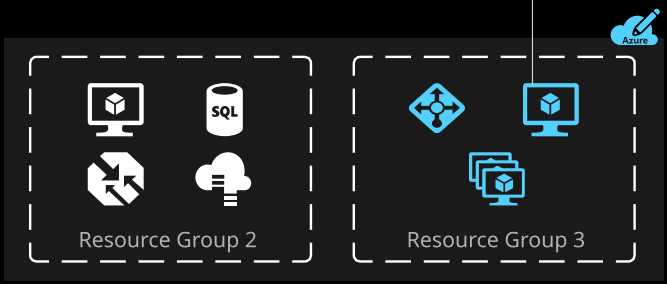
3. 

   Apply all tags from a resource group to its resources, replacing any existing tags on the resource.

```
$groups = Get-AzResourceGroup foreach ($g in $groups) {
Get-AzResource -ResourceGroupName $g.ResourceGroupName |
ForEach-Object {Set-AzResource -ResourceId $_.ResourceId
-Tag $g.Tags -Force } }
```

Moving Resources Across Resource Groups

Subscription-Finance



Subscription-IT Dept

Identifying Audit Requirements (Azure Security Center)

Azure Security Center is a built-in security management system that protects the data center within the Azure infrastructure as well as your on-premise environment, provided it's configured properly. **With the integration of Security Center in the Azure portal, it's easy to audit your environment**. You can do this to determine if your system is compliant with industry standards and your organizations specific security requirements. **Policies help with this effort.**

Azure Security Center addresses the three most urgent security challenges as follows:

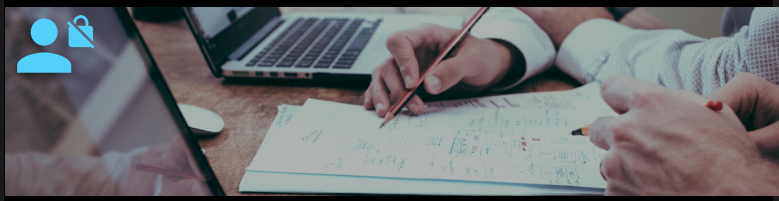
1. ### Strengthen Security Posture

   Security Center reviews your workload and helps you quickly and efficiently review the condition of your resources and determine if they are secure.

Creating a Custom Role

Role-based access control (**RBAC**) gives you the ability to grant appropriate **access to Azure AD users, groups, and services**.

A role defines what actions can be performed on Azure resources. A user or service can act on an Azure resource if they have been assigned a role containing that action. **There are existing pre-configured roles included**, but in some instances, the pre-configured roles won't fit your needs. In those situations, you need to **create a custom role, using the following steps:**

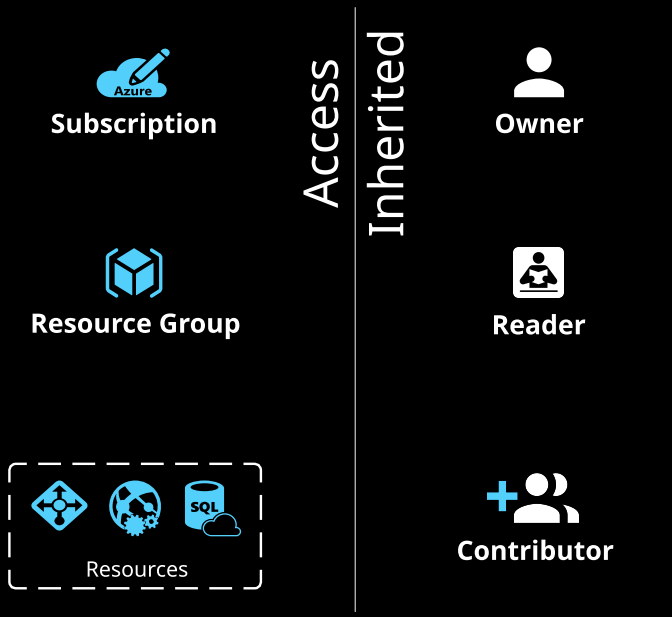


1. **Decide** how you will create the role (PowerShell, CLI, Rest API).
2. **Determine** the permissions you need.
3. **Create** the custom role.
4. **Test** the custom role.

Built-In Role Name	Description
Owner	Can manage everything including access.
Contributor	Can manage everything except access.
Reader	Can view everything but can't make changes.

Next

Custom Roles vs. Built-In Roles



Configuring Access to Azure Resources by Assigning Roles

Using Azure AD, you can designate separate administrators to serve different functions.

Administrator Permissions

Administrators can perform tasks such as adding or changing users, assigning administrative roles, resetting user passwords, managing user licenses, and managing domain names.

Global Administrator

Global administrators have access to all administrative features. By default, the person who signs up for an Azure subscription is assigned the Global Administrator role for the directory.

Viewing Role Membership

You can see and manage all members of administrator roles in the Azure Active Directory portal. When you view a role's members, you can see the complete list of permissions granted by the role assignment.

Configuring Access to Azure Resources by Assigning Roles

A **role assignment** associates a security principal to a role and is used to grant access to a resource scope. This decoupling allows you to specify that a specific role has access to a resource in your subscription and **easily add/remove security principals from that role**. Roles can be assigned to the following types of Azure AD security principals:

A

User

Roles can be assigned to **organizational users** that are in AD with which the Azure subscription is associated.

B

Groups

Roles can be assigned to **Azure AD security groups**. A user is automatically granted access to resource if the user becomes a member if a group that has access.

C

Service Principals

Service identities are represented as **service principals** in the directory. They authenticate with Azure AD and securely communicate with one another.

Resource Scope

Access does not need to be granted to the entire subscription. Roles can be assigned for both resource groups and individual resources. In Azure RBAC, a resource inherits role assignments from the parent resource.

Configuring Access to Azure Resources by Assigning Roles

In addition to **Owner**, **Contributor**, and **Reader**, Azure provides many other built-in roles to handle most security scenarios.

Role Definition

Each role is a set of properties defined in a JSON file. This **role definition** includes the name, ID, and description of the role, as well as the **permissions**, **denied permissions**, and **scope**.

Action

Allowable permissions are defined as **Actions.**

Not Action

Denied permissions are defined as **Not Actions.**

The Action and Not Action properties can be tailored to grant or deny the exact permissions you need.

Back Next

Troubleshooting RBAC, Implementing RBAC Policies, and Assigning RBAC Roles

Problems with RBAC Role Assignments

A

"Add Role Assignment" Disabled or Returns a Permissions Error

This means the client ID does not have authorization to perform the action. The user needs the following action: **Microsoft.Authorization/roleAssignments/write**

B

Error Message: "No more role assignments can be created (code:RoleAssignmentLimit Exceeded)"

The **maximum limit** of role assignments has been reached. As a workaround, use a group to reduce the number of role assignments. Azure supports **up to 2000 role assignments** per subscription.

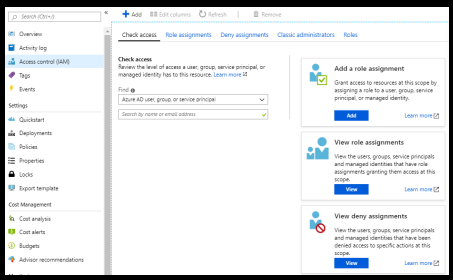
For FAQs and troubleshooting tips for role-based access control, review the Microsoft docs:

Troubleshooting RBAC Resources

Troubleshooting RBAC, Implementing RBAC Policies, and Assigning RBAC Roles

IAM (Identity and Access Management)

Access control (IAM) is used to manage access to Azure resources. Below are some of the main elements of IAM.



RBAC and the Azure Portal

A — **Resource Where IAM Is Opened**
Used to identify scope (e.g., resource group, resource, etc.)

B — **Add Button**
Used to add role assignment.

C — **Check Access Tab**
Used to view assignments for a user.

D — **Role Assignments Tab**
Used to view role assignments at active scope.

E — **Role Tab**
Used to view all roles and permissions.

Cloud IAM

Back

Configuring Network Access to Storage Accounts

Azure Storage is a service you can use to store files, messages, tables, and other types of data. You can use this storage to keep data for websites, mobile apps, desktop applications, and other types of data-driven solutions. Azure storage is also used by IaaS virtual machines and PaaS cloud services.

Azure Storage Can Be Divided into 3 Categories

Storage for VMs
This includes disks and files. Disks are persistent block storage for Azure IaaS VMs.

Unstructured Data
This includes Blobs and th Data Lake Store.

Structured Data
This includes tables, Cosmos DB, and Azure SQL Database.

Database Examples

PostgreSQL
Relational database service based on the open-source Postgres database engine.

Cosmos DB
This is a globally-distributed database service. It elastically scales throughput and storage.

Azure SQL Database
This is a fully managed database-as-a service built on Microsoft SQL.

Blobs and Data Lake Store:
Blobs are highly scalable, REST-based cloud data stores. **Data Lake Store** is a Hadoop Distributed File System (HDFS) as a service.

Configuring Network Access to Storage Accounts

Azure Storage has built-in security which allows you to **protect your storage accounts** by limiting access to a specified network. By default, this protection is not enabled.

The following steps limit access to the storage accounts to specific networks.

1. 

   Log in to the Azure Portal and go to the storage account you want to secure.

2. 

   Under the Settings menu, select Firewalls and virtual networks.

3. 

   To deny access, choose to allow access from selected networks. Click Save.

Configuring Network Access to Storage Accounts

You can manage default network access rules for storage accounts through Azure Portal, PowerShell, or CLIv2. Keep in mind the user must have proper permissions in order to apply a virtual network rule to a storage account. The user needs to have permissions to Join Service to a Subnet. Alternatively, if the user has the Storage Account Contributor built-in role, they have the required permissions.

Allowed Trusted Microsoft Services

Azure Backup (Microsoft.RecoveryServices)
Runs backups and restores of unmanaged disks in IaaS virtual machines.

Azure Data Box (Microsoft.Databox)
Enables import of data to Azure using Data Box.

Azure DevTest Lab (Microsoft.DevTest Lab)
Supports creating custom images and artifact installation.

Azure Event Grid (MicrosoftGrid)
Lets blob storage event publish & lets Event Grid publish to storage queues.

Azure Event Hubs (Microsoft.Event Hub)
Archives data with Event Hubs Capture.

Azure HDInsight (Microsoft.HDInsight)
Sends the default file system contents to an HD insight Cluster.

Azure Monitor (Microsoft.Insights)
Allows writing of monitor data to a secured storage account.

Azure Networking (Microsoft.Network)
Stores and analyzes network traffic logs.

Azure Site Recovery (Microsoft.Site Recovery):
Sets up disaster recovery by enabling replication for Azure IaaS VMs.

Azure SQL Data Warehouse (Microsoft.SQL):
Uses massively parallel processing to run complex queries on PBytes of data.

Creating and Configuring a Storage Account

Azure Storage Account	An Azure storage account provides a unique namespace in the cloud to store and access your data objects in storage.

When you create a storage account you can choose from a General Purpose **v1 Storage**, General Purpose **v2 Storage**, and **Blob Storage**.

```
The General Purpose v2 Storage account is recommended over
the General Purpose v1 Storage because it includes new
archive access and has a lower price per gigabyte.
```

Creating and Configuring a Storage Account

Azure Storage Account	An Azure storage account provides a unique namespace in the cloud to store and access your data objects in storage.

When you create a storage account you can choose from a General Purpose **v1 Storage**, General Purpose **v2 Storage**, and **Blob Storage**.

Blob Storage Account:

A blob storage account specializes in storing your unstructured data as blobs (objects) in Azure Storage. The following tiers are available for blob storage accounts:

Hot: Objects stored are accessed more frequently.

Cold: Objects stored are accessed less frequently.

Archive: Only applies to blob level storage in the general purpose v2 account.

```
The General Purpose v2 Storage account is recommended over
the General Purpose v1 Storage because it includes new
archive access and has a lower price per gigabyte.
```

Creating and Configuring a Storage Account

Every object that you store in **Azure Storage** has a unique URL. The storage account name forms the subdomain of the address. The combination of subdomain and domain name forms the **endpoint** of your storage account.

Below are some examples of endpoint URLs.

Blob Service:
http://mystorageaccount.blob.core.windows.net

Table Service:
http://mystorageaccount.table.core.windows.net

Queue Service:
http://mystorageaccount.queue.windows.net

File Service:
http://mystorageaccount.file.core.windows.net

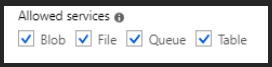
The **URL for accessing an object** in a storage account is built by appending the object's location in the storage account to the endpoint. So, if you want to access a blob, the **URL** would be the following: http://mystorageaccount.blob.core.windows.net/mycontainer/myblob

Generate Shared Access Signature

A **Shared Access Signature** (SAS) provides delegated access to resources in your storage account. An **SAS grants granular** control over the types of access you give a client who must access a storage account. The following examples are ways to set the client's acess.

1

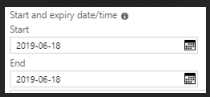
Account-Level SAS



Delegate access to multiple storage services (e.g. blob, file, queue, and table).

2

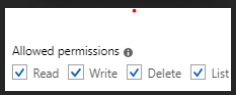
SAS Interval



Specify the start and end time a client has access.

3

SAS Permissions



Indicate which permissions are given to access the storage account (e.g. read, write, delete).

Creating and Configuring a Storage Account

Azure Storage Account	An Azure storage account provides a unique namespace in the cloud to store and access your data objects in storage.

When you create a storage account you can choose from a General Purpose **v1 Storage**, General Purpose **v2 Storage**, and **Blob Storage**.

General Purpose Account

A general purpose account grants access to Azure Storage Services like tables, queues, files, blobs, and Azure VM disks. It covers two performance tiers:

Standard: Allows you to store tables, queues, files, blobs, and Azure VM disks.

Premium: Supports only Azure VM disks

```
The General Purpose v2 Storage account is recommended over
the General Purpose v1 Storage because it includes new
archive access and has a lower price per gigabyte.
```

Generate Shared Access Signature

As you create SAS in the Azure Portal, **URI is created usingparameters and tokens.** The URI consis of your Storage Resource URI and SAS token. Below is an example of the URI as well as the components of the URI.

URI Link

Components of URI

Resource URI: Indicates a blob service endpoint with service properties parameters.
https://myaccount.blob.core.windows.net/?restype-service&comp=properties

Storage version: Indicates the version to use. **sv=2015-04-05**	**Services:** Indicates the SAS applies to the blob and file services. **ss=bf**	**Resources types:** Indicates the SAS applies to service-level operations. **srt-s**
Start Time: Specified in UTC time. **st=2015-04-29T22%3A18%3A26Z**	**Expiry time:** Specified in UTC time. **se=2015-04-30T02%3A23%3A26Z**	**Resource:** Indicates the resource is a blob. **sr=b**
Permissions: These permissions grant read and write access. **sp=rw**	**IP Range:** The IP address range accepting requests. **sip=168.1.5.60-...**	**Protocol:** Indicates requests must use https. **spr=https**

Signature: Used to authenticate access to the blob.
sig=F%6GRVA

Implement and Manage Storage

Manage Access Keys

When you create a storage account, Azure generates two 512-bit storage account **access keys**. You use these keys to authorize access to your storage account via **shared key.** You can rotate and **regenerate** the key without interrupting your applications.

Steps to View and Copy Access Keys

1. Navigate to Azure Portal.

2. Locate your storage account.

3. In the settings section, select Access Keys.

4. Find the key value under *key1* and click Copy.

5. You can also copy the entire connection string.

Manage Access Keys

Microsoft recommends you **periodically regenerate your access keys** to help secure your storage account. Two access keys are assigned so you can rotate your keys, you can **ensure your application maintains access** to Azure Storage throughout the process.

Steps to Rotate View and Copy Access Keys

1. Update the connection strings in application code to use the second key.

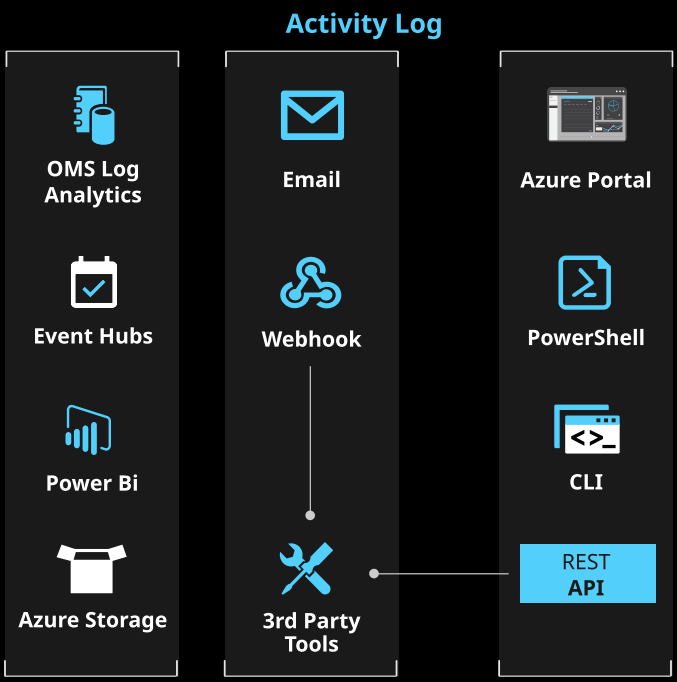
2. Regenerate the primary access key.

3. Update the connection strings in application code to use the new primary key.

4. Regenerate the secondary access key.

Monitoring the Activity Log by Using Log Analytics

The **Azure Activity Log** is a subscription log which provides insight into subscription-level events that occurred in Azure. This includes a range of data from **Azure ARM operational data to updates on Service Health Events.** The **Activity log** was previously known as Audit Logs or Operational Logs. Using the **Activity log,** can determine the who, what, and when for any write operations taken on the resources in your subscription.



Monitoring the Activity Log by Using Log Analytics

Log Analytics collects activity logs and stores the logs for **90 days free of charge**. If you store logs for longer then 90 days, you will incur data retention charges for that data. **When you are on the Free pricing tier**, activity logs do not apply to your daily data consumption.

Actions Available with Azure Activity Log Tile

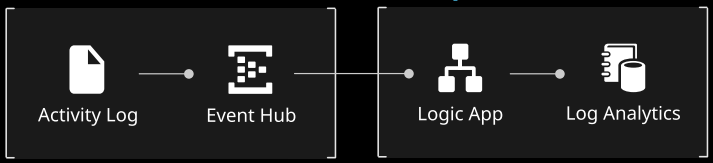
You can analyze the **activity logs** with predefined views.	You can **analyze and search** activity logs from multiple Azure subscriptions.	You can keep activity logs for longer than the default **90 days.**
You can **correlate activity logs** with other Azure platform and application data.	You can see **operational activities** aggregated by status.	You can view **trends of activities** happening on each of your Azure resources.
You can obtain a report showing **authorization changes** on all your Azure resources	You are able to identify outage or service health issues **impacting your resources**.	You can use **Log Search** to correlate user activities, auto-scale operations, and other events.

Monitoring the Activity Log by Using Log Analytics

In some situations, you may need to send logs across different subscriptions. For example, if you are a managed server provider, **you may want to collect activity logs from your customer's subscription** and store them in a Log Analytics workspace in your own subscription.

The basic strategy to do this is to have the **Azure Activity Log send events to an Event Hub** where an AzureLogic App sends them to your Log Analytics workspace.

Collect Across Subscriptions



Advantages of this approach

A. Low latency since the Azure Activity Log is streamed into the Event Hub. **This triggers the Logic App** which posts the data to Log Analytics.

B. **Minimal code is required**, and there's no server infrastructure to deploy.

Implementing Azure Storage Replication

Data in your **Azure storage account** is always replicated to ensure durability and high availability. **Azure storage application copies your data so it's protected against planned and unplanned events** ranging from hardware failures to natural disasters. You can choose to replicate data within the same data center, across zonal data centers in the same region, or even across regions.

1. **Locally Redundant Storage (LRS)**
   Replicates data to a storage unit in the same data center.

2. **Zone-Redundant Storage (ZRS)**
   Replicates data to three different storage clusters in one region.

3. **Geo-Redundant Storage (GRS) or Read-Access Geo-Redundant Storage (RA-GRS)**
   Replicates data across multiple different regions.

Implementing Azure Storage Replication

Locally redundant storage (LRS) is a low-cost option for protecting your data from local hardware failure. However, if a disaster occurs to the entire datacenter (e.g. fire or flooding), all replicas may be lost or unrecoverable. To mitigate this risk, Microsoft recommends either **zone redundant storage** or **geo-redundent-storage**. Here are some scenarios where LRS would be considered appropriate.

A. If your application stores data that can be **easily reconstructed** if data loss occurs.

B. Some applications are restricted to replicating data only within a country due to **data governance requirements**.

1. **Replication Type** — Locally Redundant Storage (LRS)

2. **Number of Copies** — Three copies of your data are maintained.

3. **Strategy** — Data is replicated three times within a single facility in a single region.

Implementing Azure Storage Replication

Geo-redundant storage (GRS) is the default and recommended replication option. Sometimes it's called cross-regional replication. GRS replicates your data to a secondary region hundreds of miles away from the primary region. **GRS costs more than LRS**, but GRS provides a higher level of durability for your data in the case of a regional outage.

If you enable **RA-GRS** and your primary endpoint for blob service is myaccount.bolb.core.windows.net then your secondary endpoint is myaccount-secondary.blob.core.windows.net. The access keys for your storage account are the same for both the **primary and secondary endpoints**.

1. **Replication Type**
   - Geo-Redundant Storage (GRS)
   - Read-Access Geo-Redundant Storage (RA-GRS)
2. **Number of Copies**
   - Six copies of your data are maintained.
3. **Strategy**
   - GRS Data is replicated three times within the primary region and three times in a secondary region.
   - RA-GRS data is replicated to a second geographic region and you have read access to your data in the second region.

Implementing Azure Storage Replication

Zone redundant storage (ZRS) synchronously replicates your data across three storage clusters in a single region. Each storage cluster is physically separated from the others and resides in its own availability zone. Each availability zone and the ZRS cluster within it are **autonomous**, with separate utilities and network capabilities.

Storing your data in a **ZRS** account ensures you'll be able to access and manage your account if a zone becomes unavailable. **ZRS provides excellent performance and extremely low latency**. Here are some additional details about ZRS:

A. ZRS is not currently available in all regions.

B. Changing from one replication to ZRS requires movement of physical data from a single storage stamp to multiple stamps within a region.

C. ZRS may not protect your data against a regional disaster where multiple zones are affected. ZRS is effective if zones are temporarily unavailable.

1. **Replication Type** — Zone-Redundant Storage (ZRS)
2. **Number of Copies** — Three copies of your data are maintained.
3. **Strategy** — Data is replicated three times across two or three facilities. These facilities are either within a single region or across two regions.

Back

Implement and Manage Storage

Creating an Export From an Azure Job

Transferring large volumes of data to or from the cloud can be challenging even with a high speed connection. In this scenario, **you should consider using the Azure Import/Export service**. The Azure Import/Export Service allows you to:

A. **Securely import large amounts of data** to Azure Blob storage (block and page blobs) and Azure Files by shipping disk drives to an Azure Data Center. In this case, you ship hard drive(s) containing your data.

B. Export data from Azure storage to hard disk drives and ship them to your on-premise sites. **You can currently export Block Blobs, Page Blobs, or Append Blobs** from Azure Storage using this service. Exporting Azure files is not currently supported. In this case you ship empty hard drives.

Scenarios When the Import/Export Service Is Useful

Migrate Data to the Cloud	**Move large amounts of data** quickly and in a cost effective manner.
Content Distribution	**Quickly** send data to your customers' sites.
Backup	Take backups of your **on-premises data** to store in Azure Blob storage.
Data Recovery	**Recover large amounts of data** stored in Blob storage and have it delivered to your on-premises location.

Next

Back to Main

Implementing Azure Storage Replication

Zone redundant storage (ZRS) synchronously replicates your data across three storage clusters in a single region. Each storage cluster is physically separated from the others and resides in its own availability zone. Each availability zone and the ZRS cluster within it are **autonomous**, with separate utilities and network capabilities.

Storing your data in a **ZRS** account ensures you'll be able to access and manage your account if a zone becomes unavailable. **ZRS provides excellent performance and extremely low latency**. Here are some additional details about ZRS:

A. ZRS is not currently available in all regions.

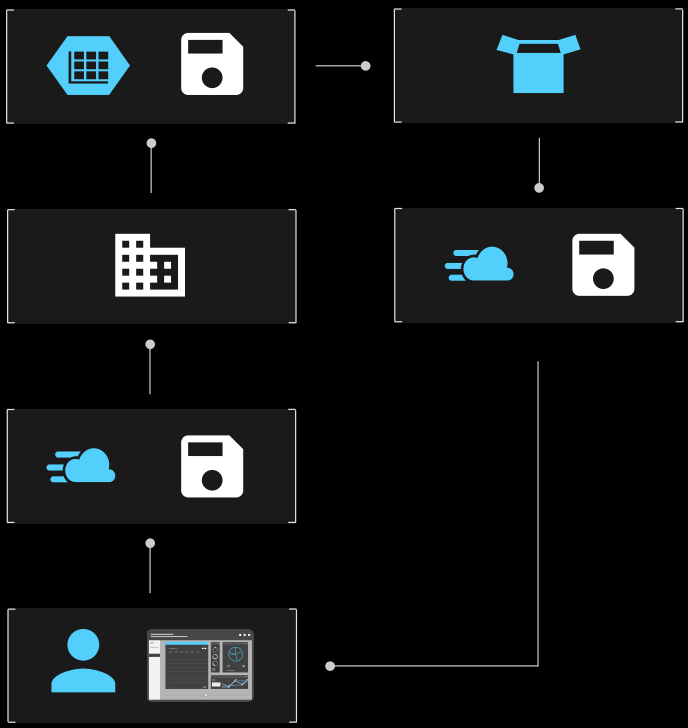
B. Changing from one replication to ZRS requires movement of physical data from a single storage stamp to multiple stamps within a region.

C. ZRS may not protect your data against a regional disaster where multiple zones are affected. ZRS is effective if zones are temporarily unavailable.

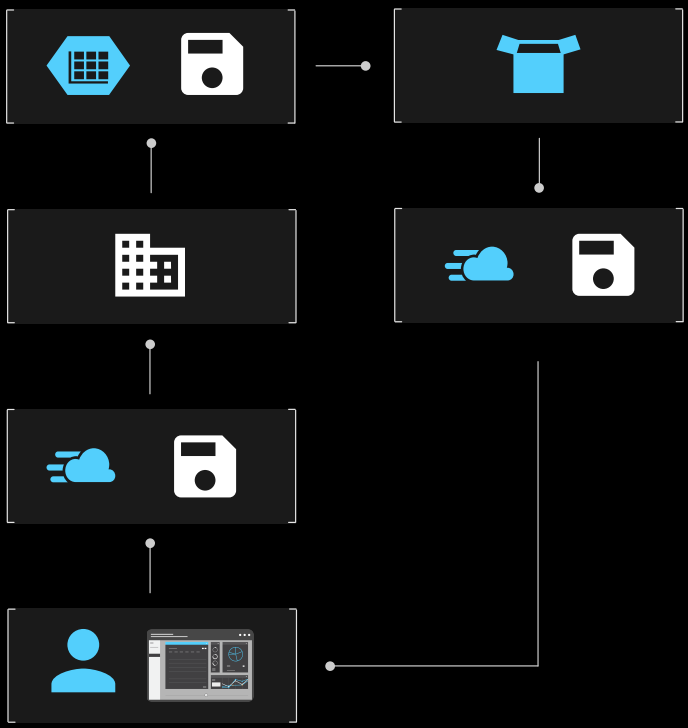
1. **Replication Type** — Zone-Redundant Storage (ZRS)
2. **Number of Copies** — Three copies of your data are maintained.
3. **Strategy** — Data is replicated three times across two or three facilities. These facilities are either within a single region or across two regions.

Back

Creating an Export From an Azure Job

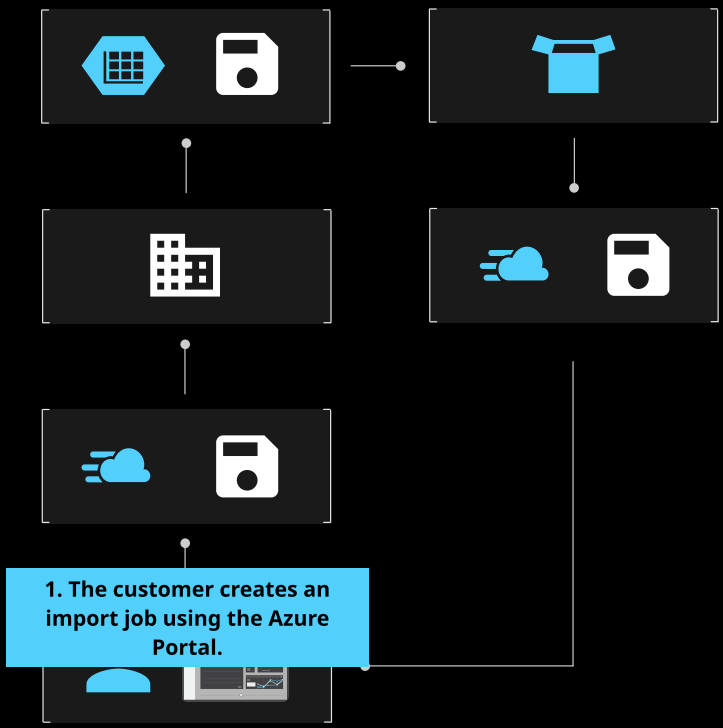


Creating an Import From an Azure Job



Back Next

Creating an Export From an Azure Job



Back Next

Using the Azure Data Box

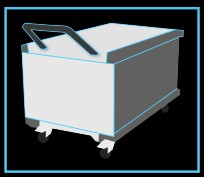
Azure Data Box products provide both offline and online solutions for moving your data to the cloud. **Offline solutions transfer large amounts of data to Azure** when there is limited or no network bandwidth.

Data Box Heavy:
Same service as Data Box, but targeted at PB-sized datasets.

Capacity: 1 PB

Weight: 500+ lb

Secure, ruggedized appliance.

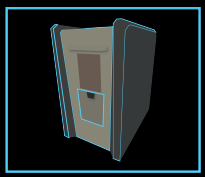


Data Box:
Bulk migration to Azure when a network isn't an option.

Capacity: 100 TB

Weight: 50 lb

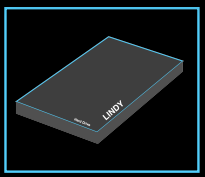
Secure, ruggedized appliance.



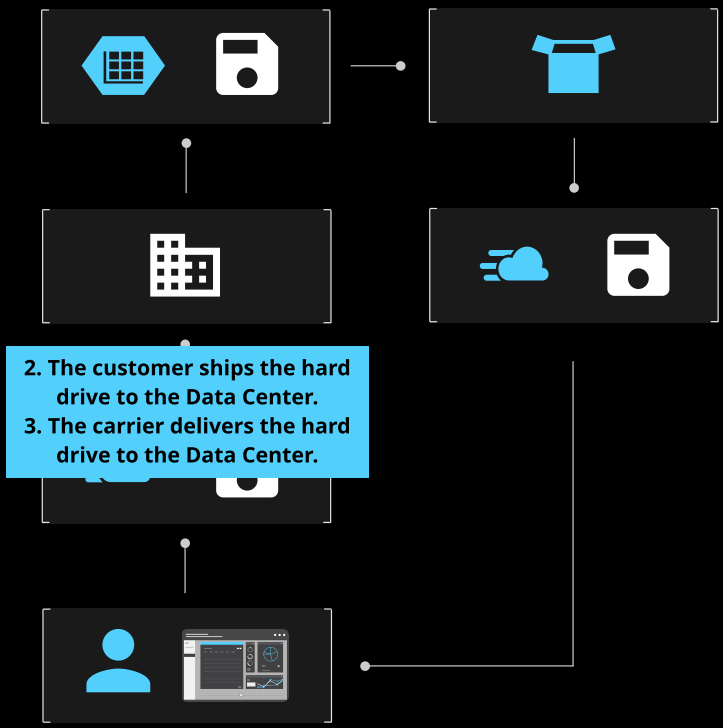
Data Box Disk:
Best for projects that require a smaller form factor.

Capacity: 8 TB each

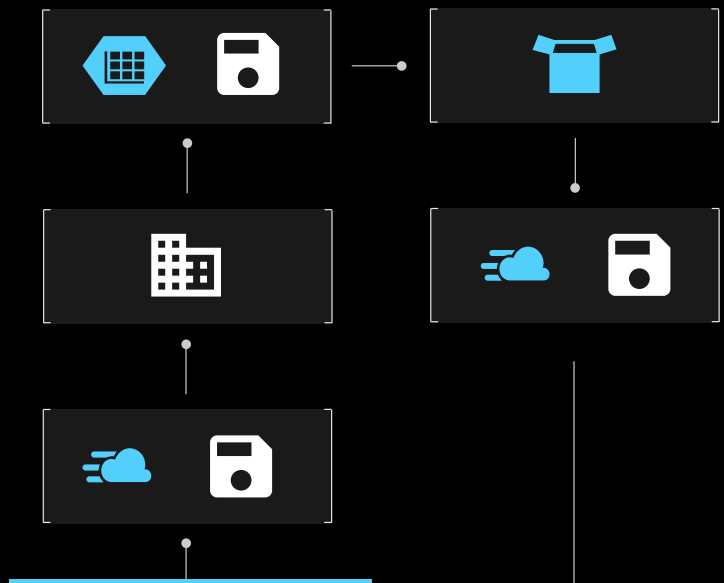
Secure, ruggedized USB drives. You can order these in packs of 5 (up to 40 TB).



Creating an Export From an Azure Job



Creating an Import From an Azure Job

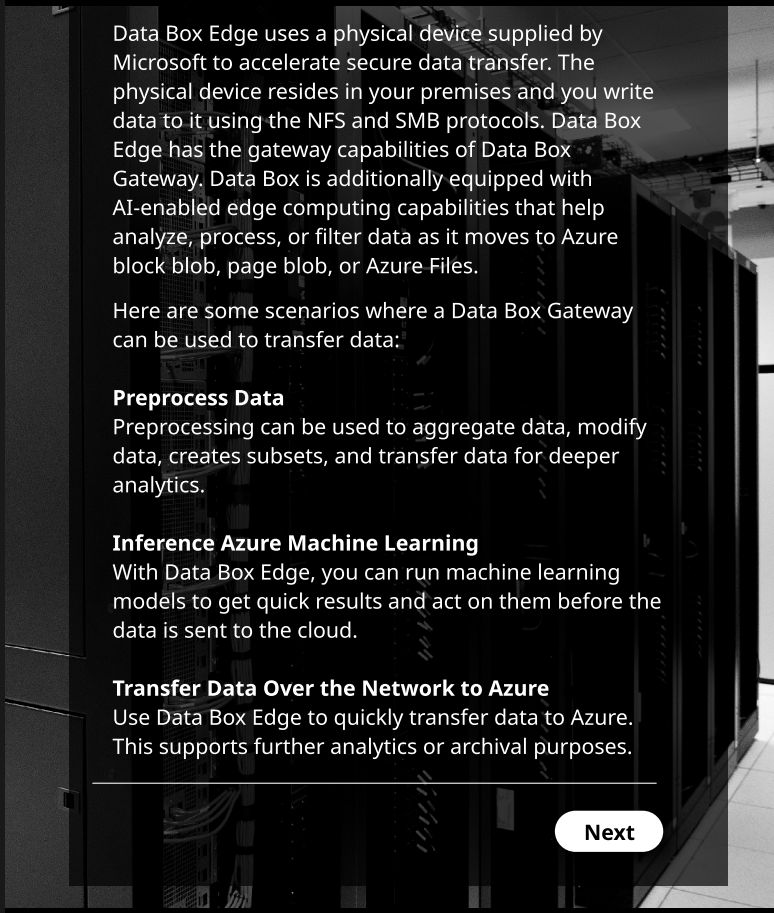


1. The customer prepares the hard drive using the Azure Import/Export client and encrypts the drive with BitLocker.
2. The customer creates an import job using the Azure Portal.

Back Next

Back to Main

Using the Azure Data Box



Data Box Edge uses a physical device supplied by Microsoft to accelerate secure data transfer. The physical device resides in your premises and you write data to it using the NFS and SMB protocols. Data Box Edge has the gateway capabilities of Data Box Gateway. Data Box is additionally equipped with AI-enabled edge computing capabilities that help analyze, process, or filter data as it moves to Azure block blob, page blob, or Azure Files.

Here are some scenarios where a Data Box Gateway can be used to transfer data:

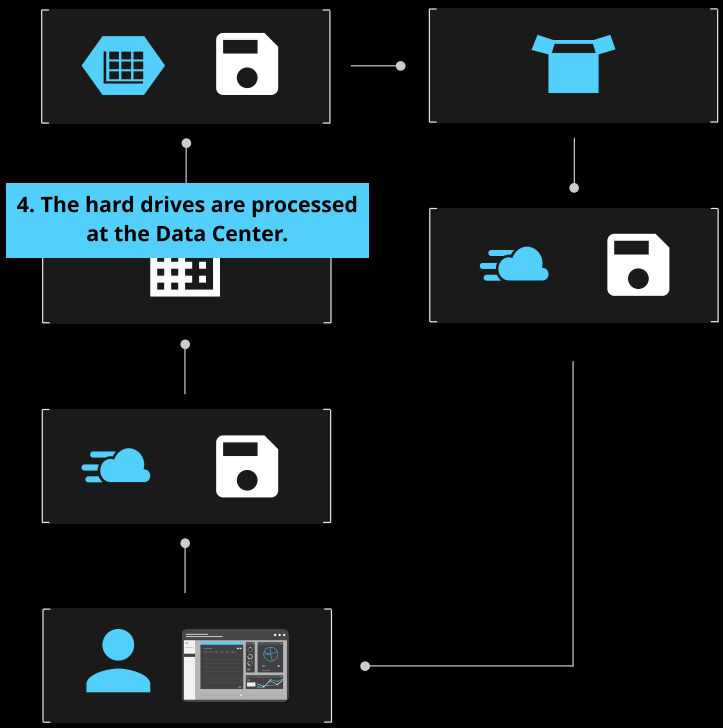
Preprocess Data
Preprocessing can be used to aggregate data, modify data, creates subsets, and transfer data for deeper analytics.

Inference Azure Machine Learning
With Data Box Edge, you can run machine learning models to get quick results and act on them before the data is sent to the cloud.

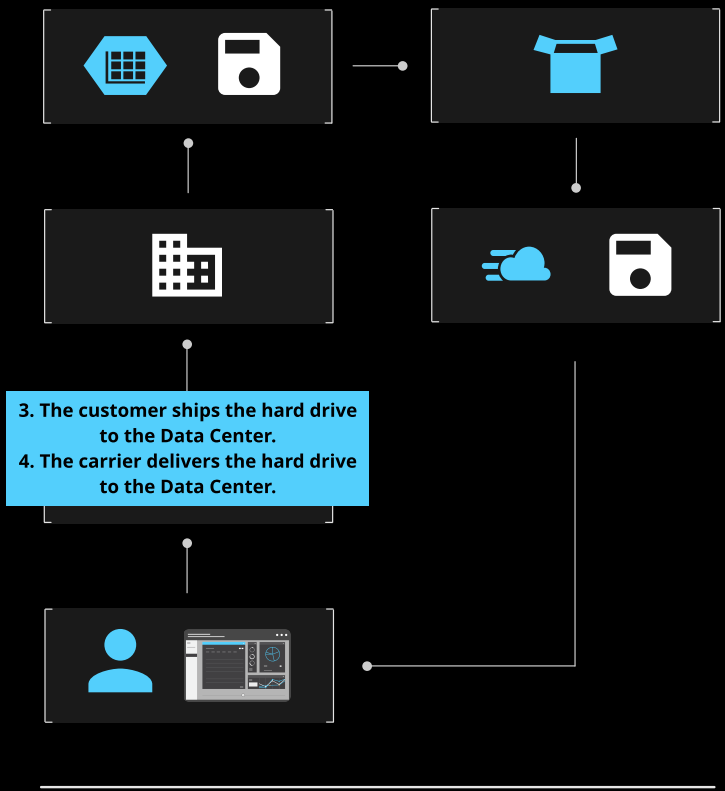
Transfer Data Over the Network to Azure
Use Data Box Edge to quickly transfer data to Azure. This supports further analytics or archival purposes.

Next

Creating an Export From an Azure Job



Creating an Import From an Azure Job



Course Navigation

Manage Azure Subscriptions and Resources

Implement and Manage Storage

Create and Configure Storage Accounts

Import and Export Data

Configure Azure Files

Implement Azure Backup

Deploy and Manage Virtual Machines

Configure and Manage Virtual Networks

Manage Identities

Using the Azure Data Box

Data Box is designed to move **large amounts of data to Azure with no impact to the network**. When selecting one of these offline products, the comparisons focus on speed and security.

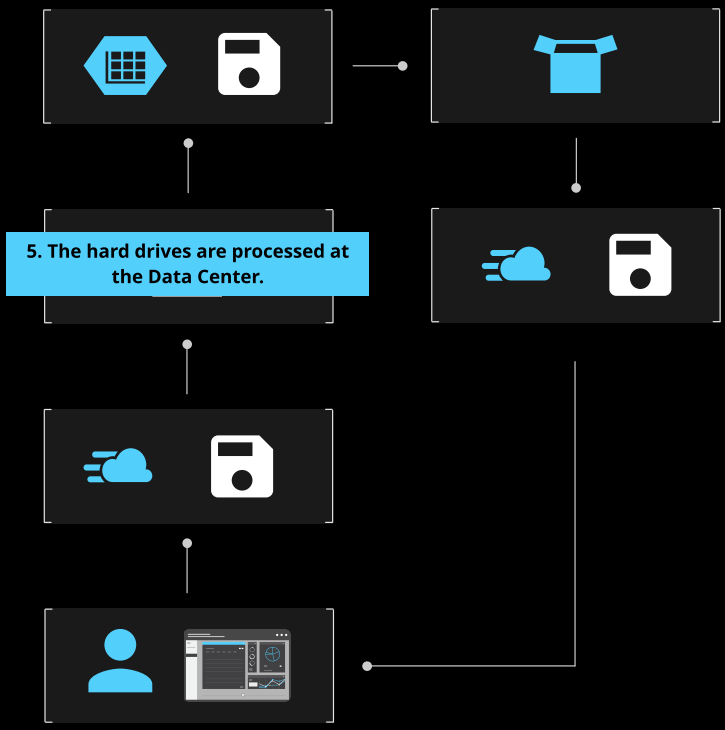
Use the estimated speed to **determine which Data Box transfers data in the time frame you need**. For example, for data less than 40 TB, use Data Box Disk, and for data greater than 500 TB, use Data Box Heavy.

Product Speed	Encryption	Physical Security
Data Box Disk (USB 3.0 connection, up to 430 MB/s)	The data is secured with **AES 128-Bit** encryption.	The disks are **tamper-resistant** and support secure update capability.
Data Box (1 Gbps or 10 Gbps network interfaces)	The data is secured with **AES 256-Bit** encryption.	**Rugged device casing** secured by tamper-resistant screws and tamper-evident stickers.
Data Box Heavy (High performance 40 Gbps network interfaces)	The data is secured with **AES 256-Bit** encryption.	**Rugged device casing** secured by tamper-resistant screws and tamper-evident stickers.

Back Next

Back to Main

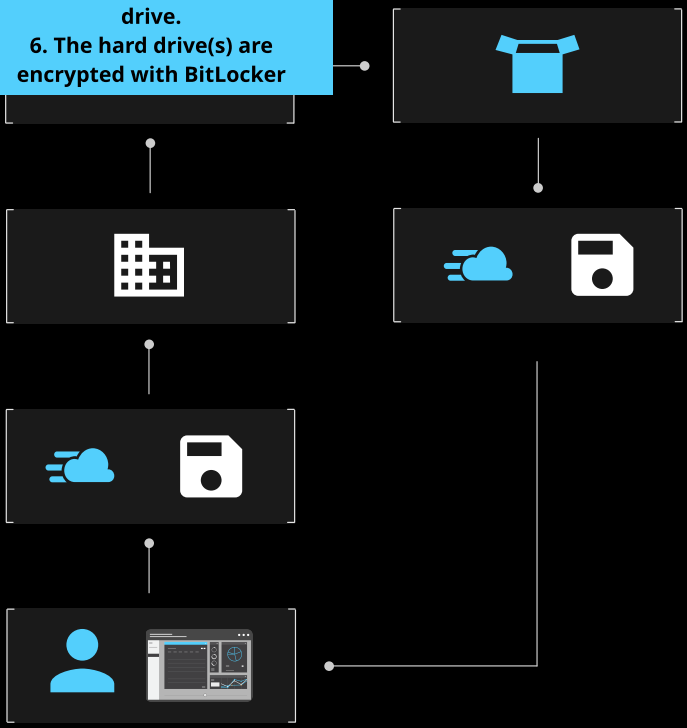
Creating an Import From an Azure Job



Back Next

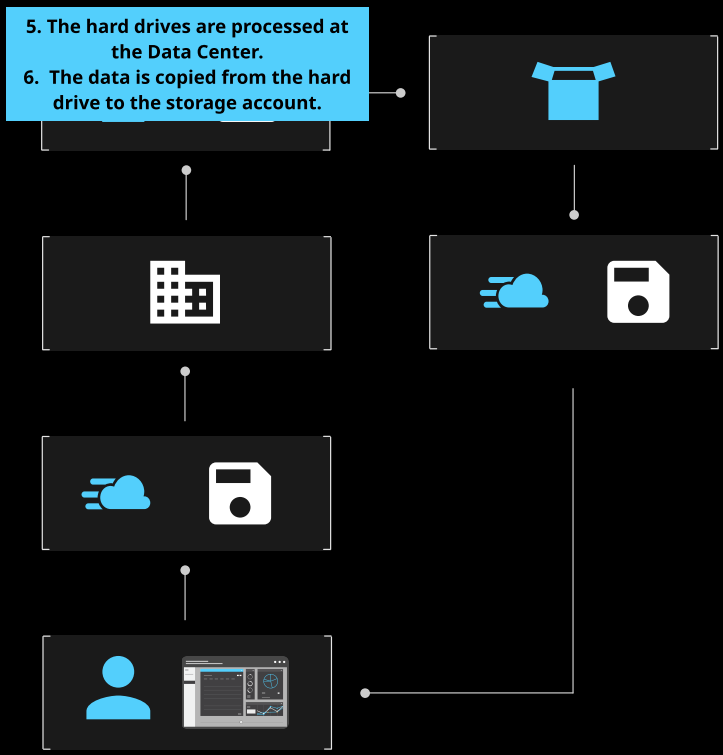
Creating an Export From an Azure Job

5. Azure copies the data from the storage account to the hard drive.
6. The hard drive(s) are encrypted with BitLocker



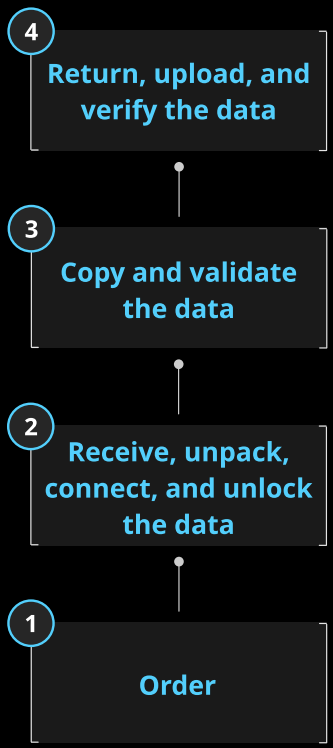
Back Next

Creating an Import From an Azure Job

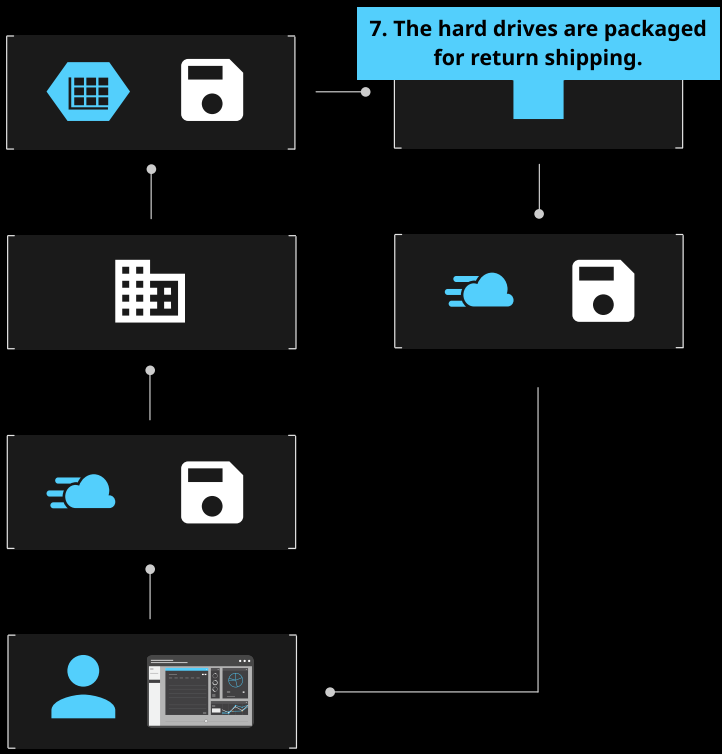


Using the Azure Data Box

Below is the offline implementation workflow which is the same for **Data Box**, **Data Box Disk**, and **Data Box Heavy**.

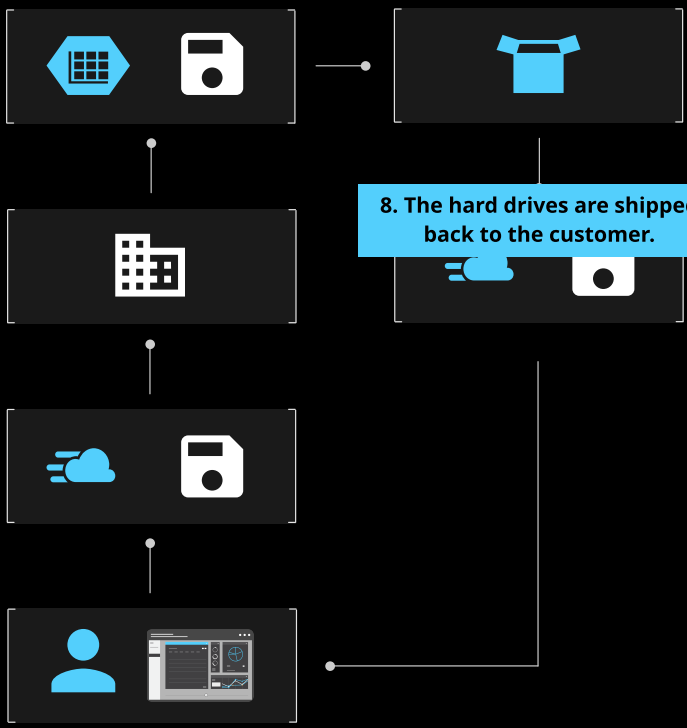


Creating an Export From an Azure Job



Back Next

Creating an Export From an Azure Job



Using the Azure Data Box

Online solution products act as network storage gateways to manage data between your site and Azure.

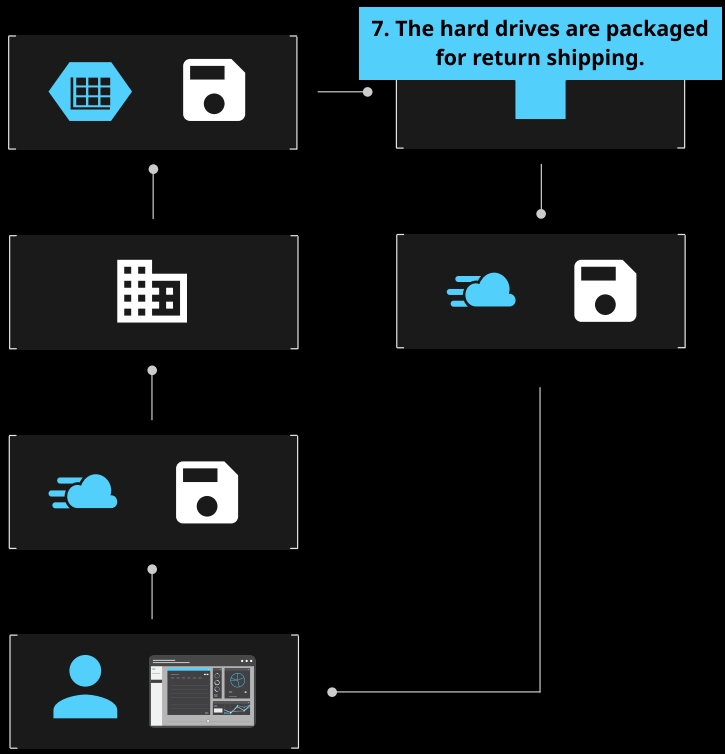
Data Box Gateway transfers data to and from Azure.

Data Box Edge is the on-premises physical network appliance used to transfer data to and from Azure.

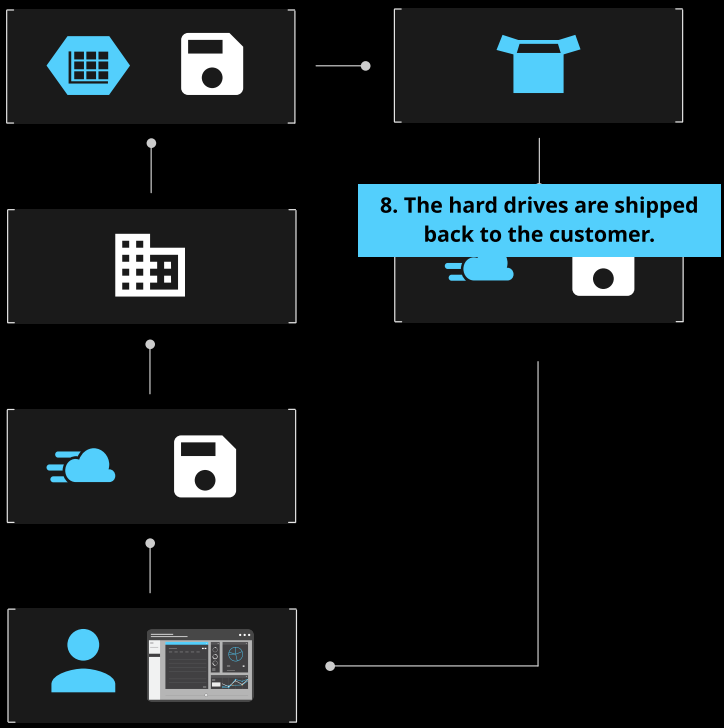
	
Data Box Gateway	**Data Box Edge**
Virtual device provisioned in your **hypervisor**. Supports storage gateway, SMB, NFS, Azure blobs, and files. **Virtual network transfer appliance** (VM) runs on your choice of hardware.	Local cache capacity is **12 TB**. Includes Data Box Gateway and IoT Edge. Data Box Edge **manages the upload to Azure** and can preprocess data prior to upload.

Implement and Manage Storage

Creating an Import From an Azure Job



Creating an Import From an Azure Job



Back Next

Course Navigation

Manage Azure Subscriptions and Resources

Implement and Manage Storage

- Create and Configure Storage Accounts
- Import and Export Data
- Configure Azure Files
- Implement Azure Backup

Deploy and Manage Virtual Machines

Configure and Manage Virtual Networks

Manage Identities

Using the Azure Data Box

Data Box Gateway is a virtual device based on a virtual machine provisioned in your virtual environment or hypervisor. The virtual device resides in your premises and you write data to it using the NFS and SMB protocols. The device then transfers your data to an Azure block blob, page blob, or Azure Files.

Scenarios Where You Can Use Data Box Gateway For Data Transfers

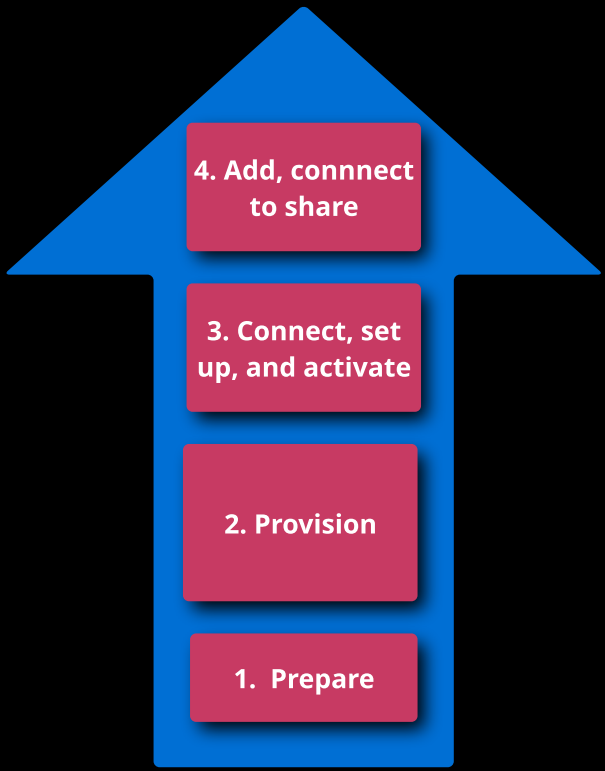
1. **Cloud Archiving** — Copy hundreds of TB of data to Azure storage using Data Box Gateway in a secure and efficient manner.
2. **Data Aggregation** — Aggregate data from multiple sources into a single location in Azure Storage for data processing.
3. **Integration with On-Premises Workloads** — Integrate with on-premises workloads such as backup and restore which use cloud storage and need local access for commonly-used files.

Back Next

Back to Main

Using the Azure Data Box

Below is the online implementation workflow which is the same for Data Box, Data Box Disk, and Data Box Heavy.



Configuring and Using Azure Blob Storage

Azure Blob storage is a service that stores unstructured data in the cloud as objects or blobs. Blob storage can store any type of text or binary data such as documents, media files, or application installers. Blob storage is also referred as object storage.

Common Uses of Blob Storage

A. **Serving images** or documents directly to a browser.

B. **Storing files** or distributed access, like installations.

C. Streaming **video and audio.**

D. **Storing data** for backup and restore, disaster recovery, and archiving.

E. Storing data for **analysis** by on-premises resources or through Azure hosted services.

Configuring and Using Azure Blob Storage

A **container** groups a set of **blobs**. All Blobs must be in a container. An account can contain an unlimited number of containers. A container can store an unlimited number of blobs.

Container names may only be in lowercase letters, numbers, or hyphens. They also must begin with a letter or a number.

Access Level for Container

Private

Use Private to ensure there is no **anonymous access** to the container and blobs.

Blob

Use **Blob** to allow anonymous public read access to blobs only.

Container

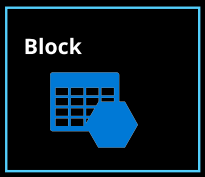
Use Container to allow anonymous **public read and list access** to the entire container, including blobs.

In addition to using Azure Portal to create containers, you can also create Blob containers using PowerShell with the `New-AzureStorageContainer` command.

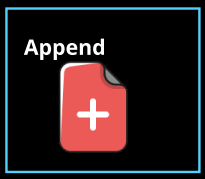
Configuring and Using Azure Blob Storage

A blob can be any file type or size. Azure storage offers three types of blobs: block blobs, page blobs, and append blobs. You specify the blob type when you create the blob. The default is a block blob.

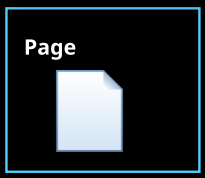
Types of Blobs



Block blobs are ideal for storing text or binary files such as documents and media files.



Append blobs are like block blobs in that they are made up of blocks, but are optimized for append operations like logging.



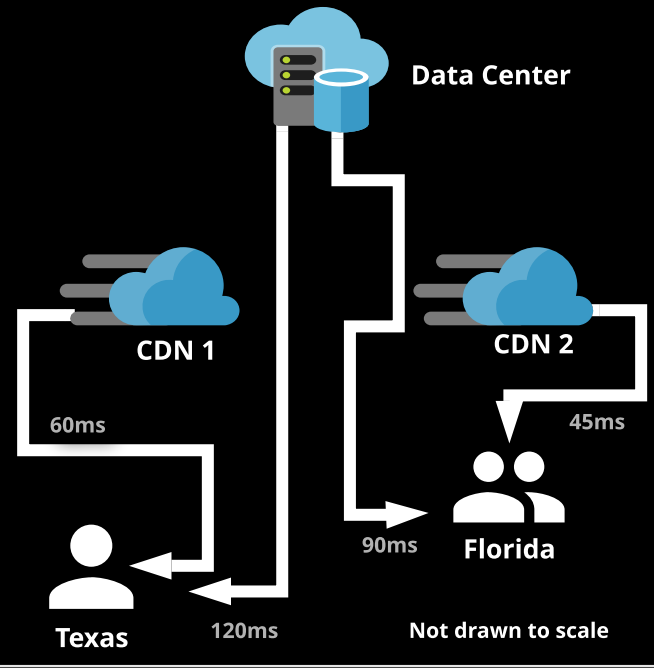
Page blobs can be up to 8 TB in size and are more efficient for frequent read/write operations (e.g. Azure Virtual Machines).

Once a blob has been created, its type cannot be changed. You can upload a local file to blob storage using PowerShell **`Set-AzureStorageBlobContent`** command.

Configuring Content Delivery Network Endpoints

A **content delivery network** (CDN) is a distributed network of servers that can efficiently deliver data to users. CDNs store cached content on edge servers close to end-users.

CDNs are typically used to deliver static content such as images, style sheets, documents, client-side scripts, and HTML files.



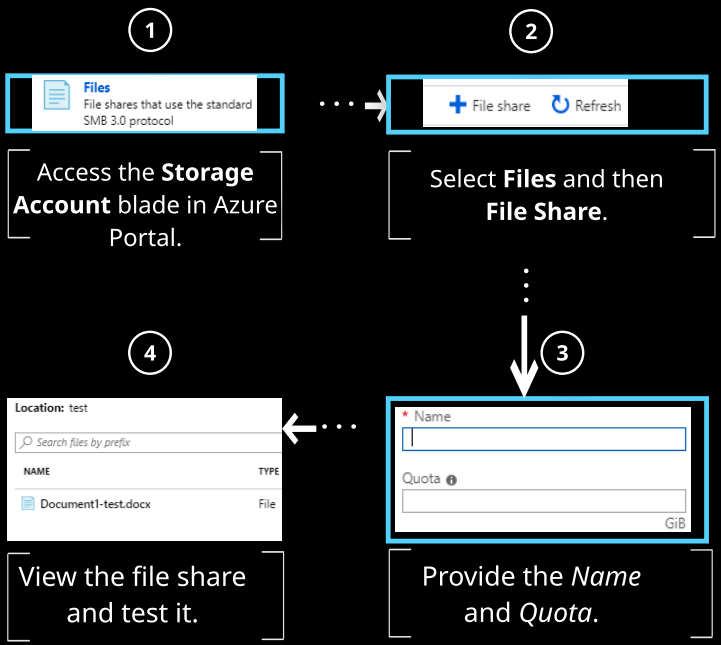
Creating an Azure File Share

Azure Files offers fully managed file shares in the cloud that are accessible via the industry-standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS. Additionally, Azure file shares can be cached on Windows servers with Azure File Sync for fast access near where the data is being used.

Different Azure Options

Feature	Description	When to Use
Azure Files	Provides an SMB interface, client libraries, and a REST interface for access to stored files from anywhere.	You want to "lift and shift" an application to the cloud which already uses the native file system APIs to share data between it and other running applications.
Azure Disks	Provides client libraries and a REST interface allowing users to store and access data from an attached VHD.	You want to "lift and shift" an application to the cloud which already uses the native file system APIs to read and write data to persistent disks.
Azure Blobs	Provides client libraries and a REST interface enabling accessing and storing unstructured data on a massive scale (block blob).	You want your application to support streaming and random access scenarios. You want to access the application data from anywhere.

Creating an Azure File Share

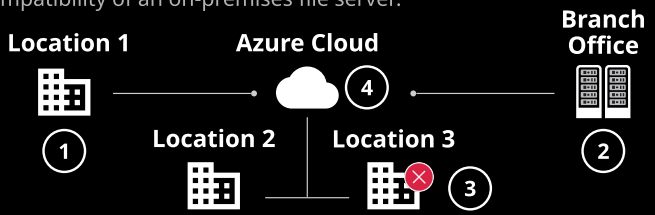


Back

Next

Create an Azure File Sync Service

Use Azure File Sync to centralize your organization's file shares in Azure Files, while keeping the flexibility, performance, and compatibility of an on-premises file server.



Uses and Advantages of File Sync

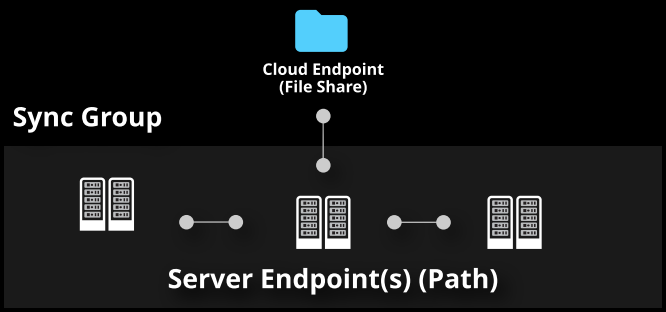
1 **Lift and Shift:**
The ability to move applications between Azure and on-premises systems. Provides write access to the same data across Windows Servers and Azure Files. This lets companies with multiple offices utilize a share with all offices.

2 **Branch Offices:**
Useful when branch offices need to backup files, or you need to set up a new server to connect to Azure storage.

3 **Backup and Disaster Recovery:**
Once File Sync is implemented, Azure Backup backups your on-premises data. Also, you can restore file metadata immediately and recall data as needed for rapid disaster recovery.

4 **File Archiving:**
Only recently-accessed data is located on local servers. Non-used data moves to Azure in what is called cloud tiering. Cloud tiering files have gray icons with an offline O file attribute to let users know the file is only in Azure.

Creating a Sync Group



Sync group defines the sync topology for a set of files. Endpoints within a sync group are kept in sync with each other. A sync group must contain at least one cloud endpoint created in your storage account and at least one server endpoint. The cloud endpoint represents an Azure file share. The server endpoint represents a path on a Windows Server.

Additional Components of the File Sync Service:

1. **Registered Server:** the name of the server or cluster where you want to create the server endpoint.
2. **Path:** The Windows Server path to be synced as part of the sync group.
3. **Cloud Tiering:** A switch to enable or disable cloud tiering.
4. **Volume Free Space:** The amount of free space to reserve on the volume where the server endpoint is located.

Troubleshooting Azure File Sync

Azure File Sync can be used in file share scenarios ranging from the very simple to the most complex. It's important to be able to troubleshoot and resolve issues you might encounter with your Azure File Sync deployment.

Troubleshooting Steps for Different Scenarios

A **Agent installation and server registration:**
During the registration, some PowerShell Commands may fail or are not recognized because they are not supported in the current version of the sync agent.

B **Sync group management:**
Within sync group management, failures with cloud server endpoint creation can occur. Remediation options are provided.

C **File synchronization:**
Issues with file synchronization range from automatically detecting exactly when a sync occurs, to sync errors on the server, to a lack of free space. Azure Files does not currently support notifications or journaling.

D **Cloud tiering (two paths for failures):**

* Files can fail to tier, which means Azure File Sync unsuccessfully attempts to tier a file to Azure Files.
* Files can fail to recall, which means the Azure File Sync file system filter (`StorageSyncs.sys`) fails to download data when a user attempts to access a tiered file.

Configuring and Reviewing Backup Reports

Azure Backup is the Azure-based service that can backup, protect, and restore your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or off-site backup solution with a reliable, secure, cost-competitive, cloud-based solution.

Steps to Configure Storage Report

1. Go into the **Azure Portal**, go to **All Services**, type **Recovery Services** in the list of resources, select **Recovery Services Vaults**, and select your vault.
2. Under **Monitoring and Reports**, select **Backup Reports**.
3. On the **Backup Reports** blade, select the **diagnostic settings** link.
4. Select **Turn on diagnostics** to open the user interface to configure a storage account.
5. In the *Name* box, enter a setting name. Select the **Archive to storage account** checkbox so reporting data can start flowing to the storage account.
6. Under the **Log** section, select the **AzureBackupReport** checkbox. Move the slider to select a retention period for this reporting data.
7. In the *Name* box, enter a setting name. Select the **Archive to storage account** checkbox so reporting data can start flowing to the storage account.
8. Review all changes and select **Save**.
9. The **Diagnostic Settings** table should now show the new setting enabled for the vault.

Implement and Manage Storage

Performing a Backup Operation

1. **Create a recovery services vault**
   To back up your files and folders, you need to create a Recovery Services vault in the region where you want to store data. You also need to determine how you want your storage replicated (e.g. geo-redundant).
2. **Download files**
   Download the backup agent for Windows Server or Windows Client and the vault credentials. The vault credentials will be used in the next step to register the backup agent.
3. **Install and register the backup agent**
   You can use the Microsoft Azure Recovery Services Agent on-premises to backup your files and folders.
4. **Backup your files and folders**
   Your initial backup includes two key tasks: scheduling the backup and backing up the files and folders for the first time.

Back Next

Manage Azure Subscriptions and Resources

Implement and Manage Storage

- Create and Configure Storage Accounts
- Import and Export Data
- Configure Azure Files
- Implement Azure Backup

Deploy and Manage Virtual Machines

Configure and Manage Virtual Networks

Manage Identities

Back to Main

Creating a Recovery Services Vault

A Recovery Services vault is a storage entity in Azure that houses data. Customers typically use this vault to copy data or configuration information for virtual machines (VMs), workloads, servers, or workstations. You can use Recovery Services vaults to hold backup data for various Azure Services such as IaaS VMs (Linux or Windows) and Azure SQL databases. Recovery Services vaults support System Center DPM, Windows Server, Azure Backup Server and more.

Within the Azure subscription, you can create up to 25 Recovery Service vaults per region.

Back

Next

Back to Main

Creating and Configuring a Backup Policy

VMs are backed up based off the schedule configured in a backup policy. Recovery points are created from your backups. Recovery points are accessible within the Recovery Services vault.

Steps to Create Backup When You Create an Azure VM:

1. In Azure Portal, click **create a resource**.
2. In the Azure Marketplace, click **Compute,** and then select a VM image.
3. Set up the VM based off of the OS (e.g. Windows or Linux).
4. On the **Management** tab, click **Enable backup**.
5. Azure Backup backs up to a Recovery Service vault. Click **Create New**.
6. Select the default vault name or create your own unique name.
7. Specify or create a resource group where the vault will be located.

8. Accept the default backup policy or customize it to meet the needs of your organization.

Performing a Restore Operation

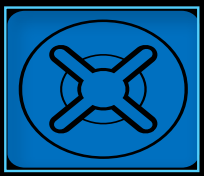
1. Sign in to the Azure portal and click **Virtual Machines** in the left pane.
2. In the virtual machines menu, click **Backup** to open the Backup dashboard.
3. In the Backup dashboard, click **File Recovery**. Select **recovery point** from the drop down menu.
4. To download the software used to copy files from a recovery point, click **Download executable** for Windows VM or **Download script** for Linux.
5. The downloaded file is password protected and requires a password. In the **File Recovery** menu, click the copy button.
6. Find the downloaded file and right-click the executable or script. Run it with admin credentials. When prompted, enter the necessary credentials.

Tasks Available When Performing Restore

Select recovery mode:
Identify the server where the backup was created.

Select volume and date:
You can restore from any point in time. First select the date and then the time.

Select items to recover:
Choose the files and folders you wish to restore.



Specify recovery option:
You can restore to the original location or another on the same machine.

Once you create your backup, you can use the Backup Agent to recover data.

Back

Deploy and Manage Virtual Machines

Create and Configure a VM for Windows and Linux

Creating a Virtual Machine

There are multiple ways to deploy virtual machines. But no matter which process we take to create virtual machines, these are the basic steps:

1. **Select an image or disk.**
2. **Provide required information.**
3. **Provide optional information.**
4. **Provision the machine.**

1. **Select an image or disk:** When deploying our virtual machine we typically select an image from the Marketplace. The newly created disk is in VHD format.
2. **Provide required information:** Information provided includes the host name, username, and password for the new virtual machine (VM).
3. **Provide optional information:** Optional information provided includes domain membership, virtual networks, storage account, cloud service, and availability set.
4. **Provision the machine:** We can provision through Azure Portal, Powershell, and CLI.

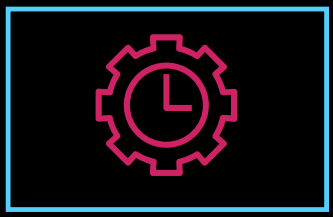
Deploy and Manage Virtual Machines

Create and Configure a VM for Windows and Linux

Configuring for High Availability

As an Azure administrator, you must be prepared for planned and unplanned failures. There are three scenarios that can lead to your VM in Azure being impacted: unplanned hardware maintenance, unexpected downtime, and planned maintenance.

Unplanned hardware maintenance:
This occurs when the Azure platform predicts that the hardware or any component related to the physical machine is about to fail.



Unexpected downtime:
This occurs when the hardware or physical infrastructure for the virtual machine fails unexpectedly.

Planned maintenance:
These are periodic updates made by Microsoft to the underlying Azure platform to improve overall reliability, performance, and security of the platform infrastructure.

Back Next

Back to Main

Linux Academy

Configuring for High Availability

An Availability Set will help with the availability of your virtual machine(s). The Availability Set is a logical grouping capability you can use in Azure to ensure the VM resources you place within it are isolated from each other when they are deployed within an Azure Data Center.

Principles When Creating an Availability Set

For redundancy, configure multiple virtual machines in an Availability Set.

Configure each **application tier** into separate Availability Sets.

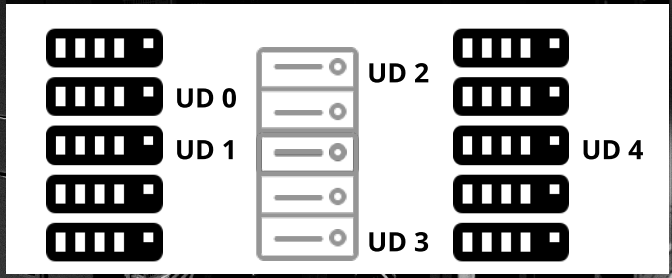
Combine a L**oad Balancer** with Availability Sets.

Use managed disks with the **virtual machines** (VMs).

Configuring for High Availability

An **update domain** allows Azure to perform incremental or rolling upgrades across a deployment. Each update domain contains a set of virtual machines and associated physical hardware that can be updated and rebooted at the same time. During the planned maintenance, only one update domain is rebooted at a time. By default, we have five update domains, but we can configure up twenty.

A **fault domain** defines a group of Virtual Machines sharing a common set of hardware or switches that share a single point of failure.

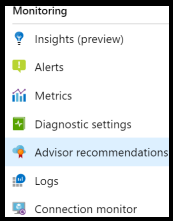


Back Next

Back to Main

Linux Academy

Configuring Monitoring, Networking, Storage, and Virtual Machine Size

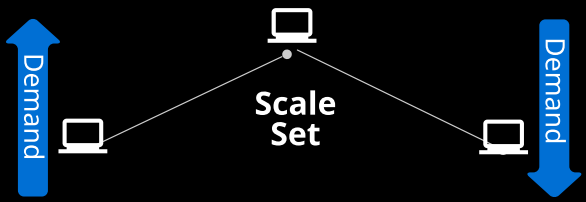


You can take advantage of many opportunities to monitor your VMs by collecting, viewing, and analyzing diagnostic and log data. To do simple monitoring in the Overview screen of Azure portal, you will see the CPU, Network, Disk bytes, and Disk Operations. The Monitoring section provides access to the metric, diagnostic settings, and Advisor recommendations.

Back
Next
Back to Main

Deploying and Configuring Scale Sets

Virtual machine scale sets are in Azure Compute Resource. You can use these to deploy and **manage a set of identical VMs**. When all VMs have identical configurations, VM scales are designed to support true autoscaling and no prior provisioning of VMs is required.



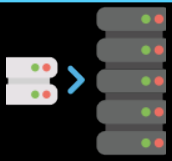
Benefits of Scaling

All VM instances are created from the same base OS image.

Scale sets support the use of either Azure Load Balancer (Layer 4) or Application Gateway (Layer 7).

Scale sets can be used to run multiple instances of your application.

Customer demand for your application may constantly change. With scaling you pay for what you use.

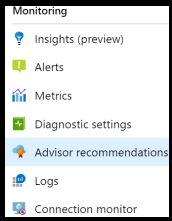


Scale set supports up to 1000 VM instances.

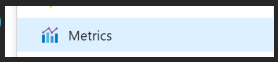
Deploy and Manage Virtual Machines

Create and Configure a VM for Windows and Linux

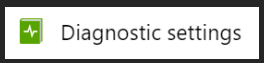
Configuring Monitoring, Networking, Storage, and Virtual Machine Size



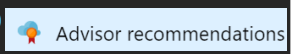
You can take advantage of many opportunities to monitor your VMs by collecting, viewing, and analyzing diagnostic and log data. To do simple monitoring in the Overview screen of Azure portal, you will see the CPU, Network, Disk bytes, and Disk Operations. The Monitoring section provides access to the metric, diagnostic settings, and Advisor recommendations.

1. 

Metrics:
This contains CPU details, data disk details, disk reads, OS disk details, network billing information, and much more information as it relates to that VM.

2. 

Diagnostic Settings:
The VM's diagnostic settings blade will be different for Windows VM compared to Linux VM. On Windows, it contains performance counters, logs, crash dumps, sinks, and Agent.

3. 

Advisor Recommendations:
Advisor is a personalized cloud consultant that helps you follow best practices to optimize your Azure deployments.

Back Next

Back to Main

Linux Academy

Modifying the Azure Resource Manager(ARM) Template

ARM templates are a way to declare objects for deployment. They define the set of resources needed for an application and are constructed in JSON format.

Template structure

```
$schema":"http://schema.management.azure.com/schemas/
2015-01-01/deploymentTemplate.json#", "contentVersion":
"","parameters": { }, "variables": { }, "functions": {
},"resources": [ ], "outputs": { } }
```

Element Name	Required	Description
`$schema`	✓	Location of **JSON schema** file that describes the version of the template language.
`contentVersion`	✓	The version of the **template** (such as 1.0.0.0).
`parameters`	–	Values provided when deployment is executed to customize **resource deployment.**

Modifying the Azure Resource Manager(ARM) Template

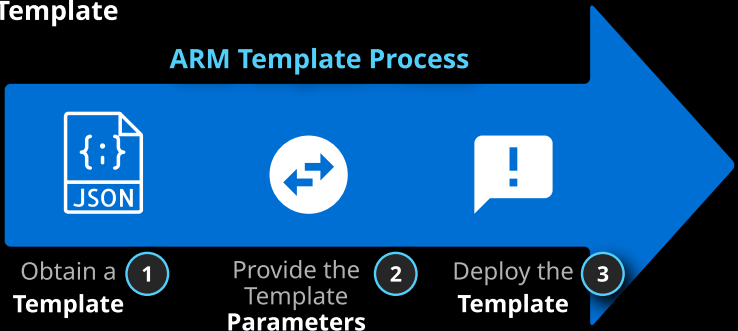
(continued)

Element Name	Required	Description
`variables`		Values used as JSON in the template to **simplify** the template language.
`functions`		User-defined **functions** available within the template.
`resources`		Resource types **deployed or updated** in a resource group.
`outputs`		Values returned **after** deployment.

Back Next

Modifying the Azure Resource Manager (ARM) Template

ARM Template Process



A Azure provides many quickstart templates. We use these when possible and modify them to address our business requirements.

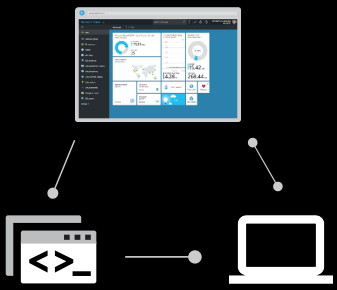
B Values for ARM templates are provided in a parameters file. This is also a JSON file type. By using this file, we can reuse a template without making changes to it.

C Once the template and parameter files are ready, we can use the portal, PowerShell, or the CLI to deploy the template.

Deploying Windows and Linux VMs

The Azure Portal allows access to **Bash/PowerShell.** Because of this, we can run **Powershell/CLI** through **Cloud Shell** as an alternative to running it locally.

Custom script extensions download and execute scripts on Azure VMs. This allows for post-deployment configuration, software installs, and multiple different types of configurations.



Adding a Data Disk

Just like any other computer, VMs use disks as **a place to store the operating system, applications, and data**. All Azure VMs have at least two disks: one for Windows and a second temporary disk. VMs can also have one or more data disks. All disks are stored as **VHDs**.

Virtual Machine Disk

Operating System Disk: 2048 GB Max
Registered as a SATA drive and labeled as the C drive by default.

Temporary Drive
Used for storing `pagefile.sys` and labeled as the D drive by default. On Linux, the disk is `/dev/sdb` and mounted to `/mnt`.

Data Disk: 4096 GB
This is used to store application data or other types of data we need to keep.

Types of Storage

Unmanaged Disks
In an unmanaged disk, we manage the storage accounts we use to store the VHD disks.

Managed Disks
Azure manages the storage accounts we use for our VM disks.

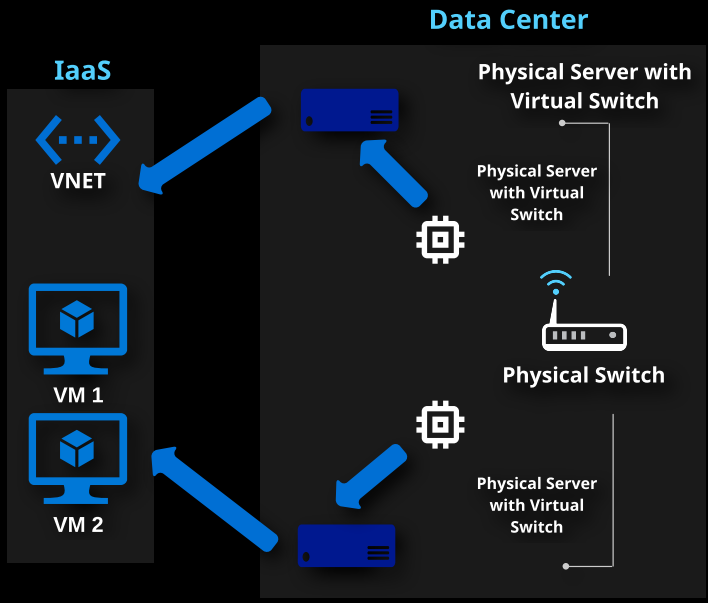
Premium Storage
Azure Premium Storage delivers high performance, low-latency disk support for VMS with I/O intensive workloads.

Adding a Network Interface

Virtual machines (VMs) on Azure can have multiple virtual network interface cards (NICs) attached to them.

We cannot add existing network interfaces to a new VM, nor create a VM with multiple network interfaces by using the Azure Portal. **We can do both by using the CLI or PowerShell.**

Different VM sizes support a varying number of NICs, so we will want to make sure our VM is sized accordingly.

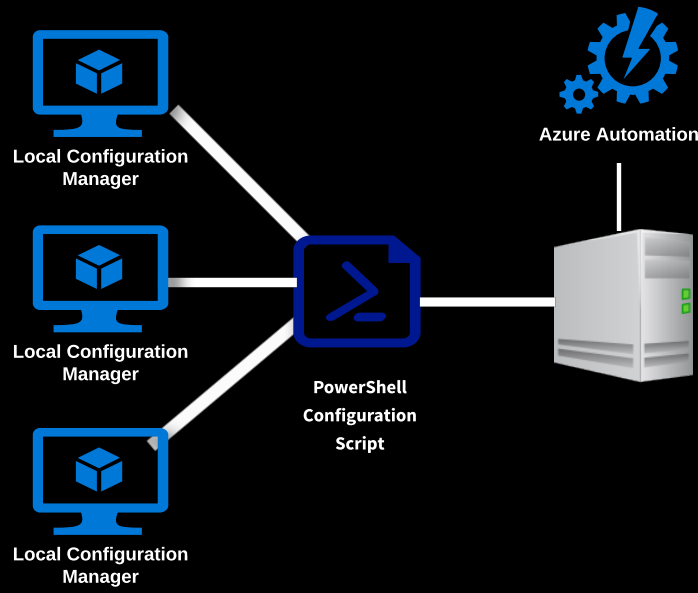


Automating Configuration Management Using PowerShell DSC and VM Agent Custom Script Extensions

Virtual machine extensions are small applications that provide post-deployment configuration and automate tasks on Azure VMs.

Custom Script Extension is a tool that can be used to automatically launch and execute VM customization tasks post-configuration.

DSC is a management platform in Windows PowerShell that enables deploying and managing configuration data for software services as well as managing the environments these services run.



Course Navigation

Manage Azure Subscriptions and Resources

Implement and Manage Storage

Deploy and Manage Virtual Machines

Create and Configure a VM for Windows and Linux

Automate Deployment of VMs

Manage an Azure VM

Manage VM Backups

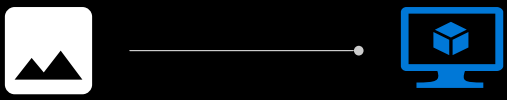
Configure and Manage Virtual Networks

Manage Indentities

Managing VM Sizes and Moving VMs From One Resource Group to Another

Changing the VM after deployment can be done per VM or in an Availability Set. The VM must be deallocated if we're changing it to a different hardware category. We can also create an image and then create the VM with a different size.

VM Size (SKU)	Type
B, Dsv3, DSv2, Av2, DC	General Purpose
Fsv2, Fs, F	Compute Optimized
Esv3, Ev3, M, GS, G, DSv2, Dv2	Memory Optimized
Ls	Storage Optimized
NV, NVV2, NC, NCv2, NCv3, ND	GPU (Graphics Optimized)
H	High Performance Compute



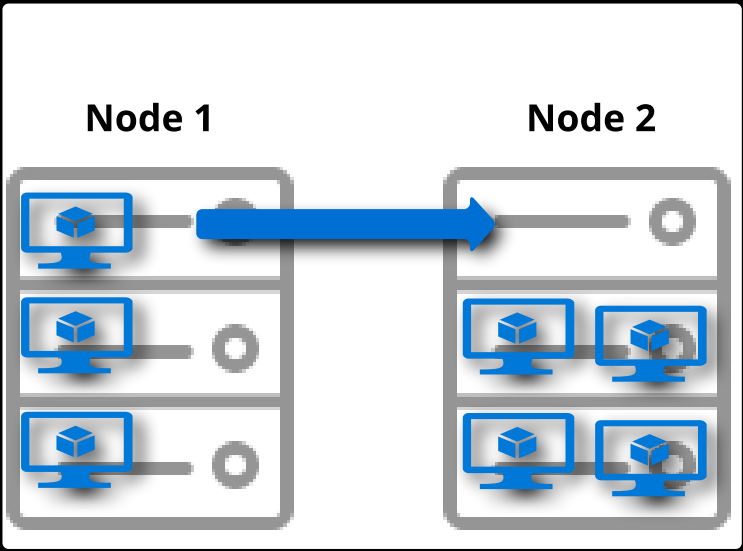
Capture Image and Resize

Back Next

Back to Main

Redeploying a VM

There may be situations where we're not able to connect to a VM, have intermittent problems, or **suffer temporary performance issues.** Redeploying the machine may help resolve the issue because this process moves the virtual machine to a different node such as a physical machine.



Configuring VM Backup

Protecting workloads, and documenting how those workloads are being protected, is a critical requirement for business continuity. Additionally, you will want to plan how often workloads are backed up, and what types of backups are performed.

Options for workload protection include:

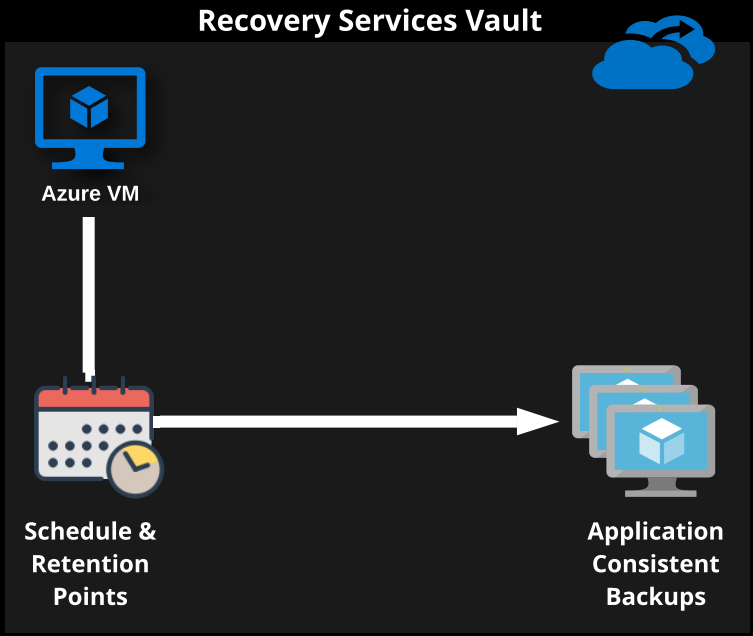
A. **Extending on-premises data protection solutions into Azure:** There are many backup solutions available within the Azure Marketplace.

B. **Using native features in Azure to enable data protection:** Something like Azure Backup, a native data protection service in Azure, allows for protection of on-premises and Azure workloads.

Backup Process Diagram

Defining and Implementing Backup Policies

Editing policies, adding new policies, and switching VMs between policies is all possible **within the Recovery Services Vault.**



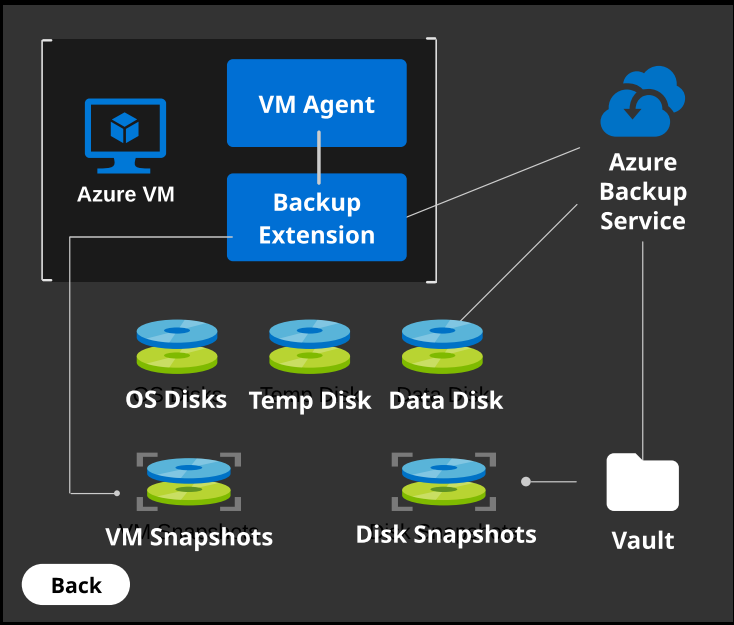
Configuring VM Backup

Protecting workloads, and documenting how those workloads are being protected, is a critical requirement for business continuity. Additionally, you will want to plan how often workloads are backed up, and what types of backups are performed.

Options for workload protection include:

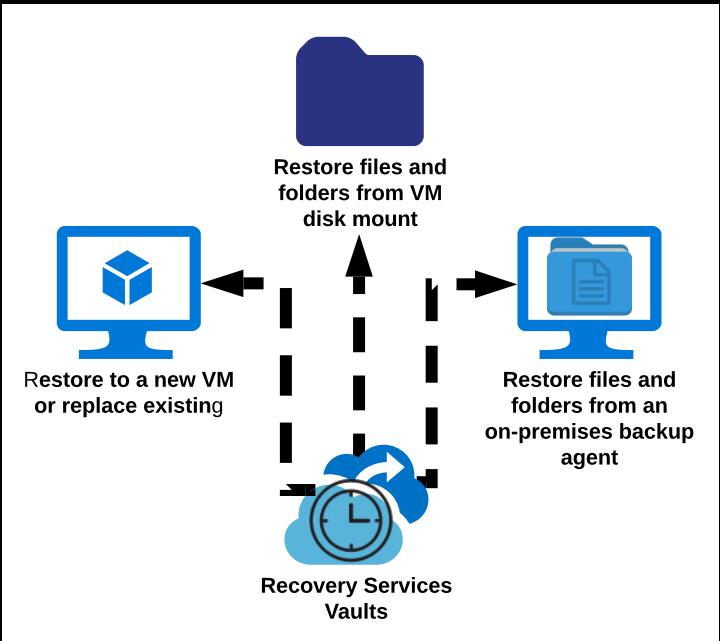
A **Extending on-premises data protection solutions into Azure:** There are many backup solutions available within the Azure Marketplace.

B **Using native features in Azure to enable data protection:** Something like Azure Backup, a native data protection service in Azure, allows for protection of on-premises and Azure workloads.



Performing VM Restores

From one of the **three consistent types of backups**, you may restore individual files or the entire snapshot.



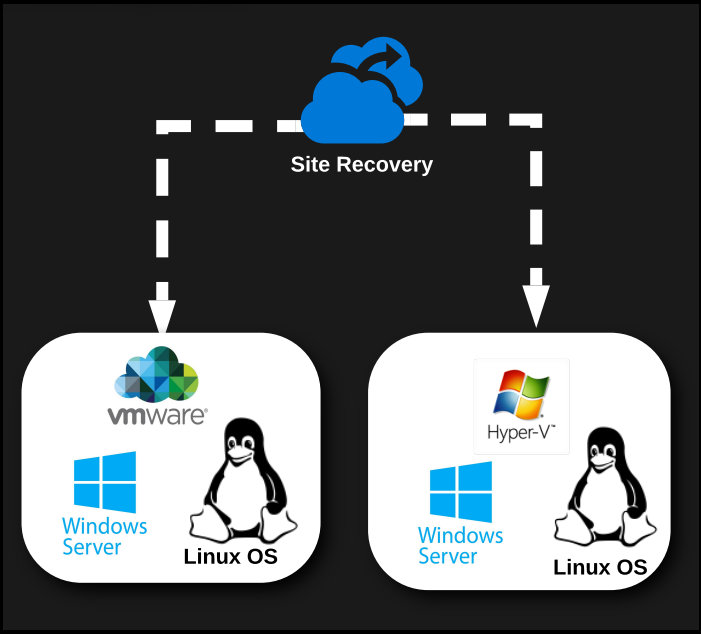
Back Next

Deploy and Manage Virtual Machines

Manage VM Backups

Azure Site Recovery

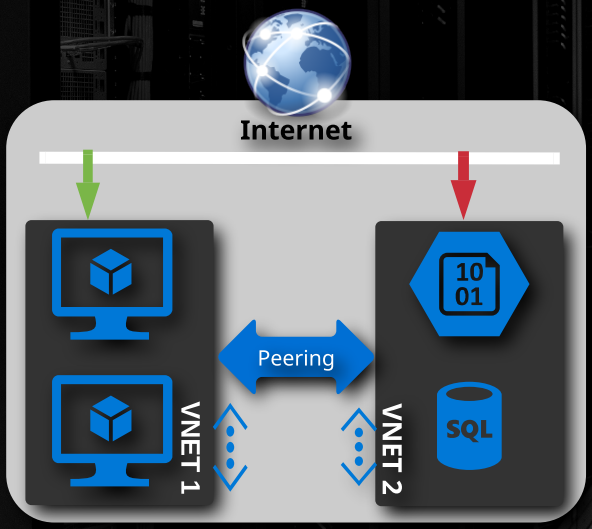
You will be using **Azure Site Recovery to replicate the on-premise environment** on both physical and virtual machines. One of the benefits of Azure Site Recovery is that it supports both **Hyper-V Machines and VMware machines.** Azure Site Recovery allows you to replicate data from the on-premises environment to the Azure cloud, or another physical site.



Back

Creating and Configuring VNet Peering

We can create peer networks in the same subscription or between two different subscriptions. Peered networks have a low-latency, high bandwidth connection. They have a private connection and act as if they existed in the same virtual network.



Next

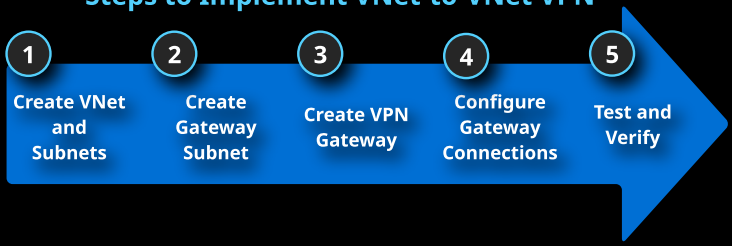
Configure and Manage Virtual Networks

Create Connectivity Between Virtual Networks

Creating and Configuring VNet to VNet

We can connect our VNets with a VNet-to-VNet VPN connection. Using this connection method, we create a VPN gateway in each virtual network. **We can also use the VPN gateway to provide a connection to an on-premises network**. This is called a Site-to-Site (S2S) connection. In both cases, a secure tunnel using IPsec/IKE provides the communication between both networks.

Steps to Implement VNet-to-VNet VPN



Information used to create VPN Gateway

Name and Gateway Type
Name your gateway and use the VPN gateway type.

VPN Type
Most VPN types are route-based. Route-based VPN does not reference VPN tunnel.

Virtual Network
In this section, we just need to associate the virtual network with the gateway.

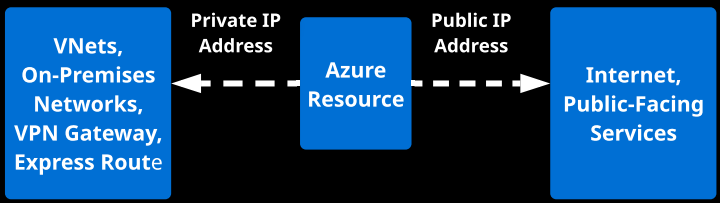
SKU
SKU choice affects how many tunnels we can have and the aggregate throughput benchmark.

Timing
It can take up to 45 minutes to provision the VPN gateway.

IP Address
A gateway needs a public IP address in its configuration to communicate with a remote network.

Configuring Private and Public IP Addresses, Network Routes, Network Interfaces, Subnets, and Virtual Networks

We can assign IP addresses to Azure resources to communicate with Azure resources, our on-premises network, and the Internet. There are two types of IP addresses we can use in Azure. Virtual networks can contain both public and private IP address spaces.



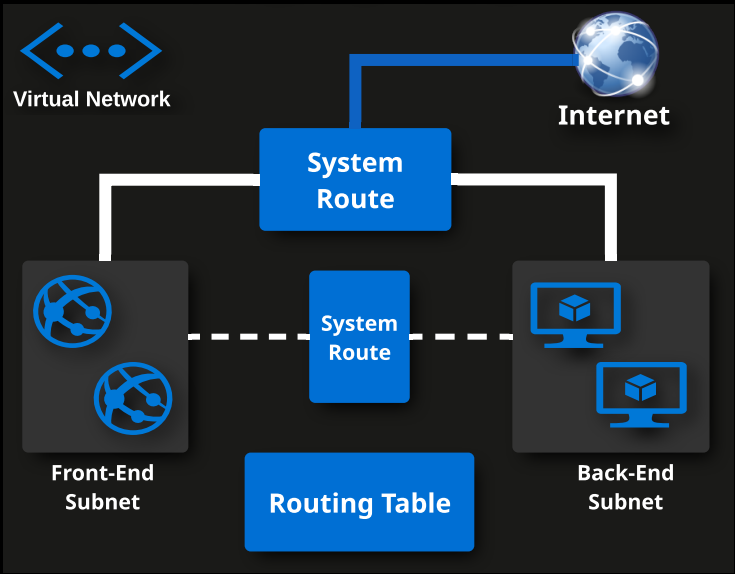
Private IP addresses: Used for communications within an Azure virtual networks (VNet) and our on-premises network when we use a **VPN gateway or ExpressRoute circui**t to extend our network to Azure.

Public IP addresses: Used for communication with the Internet, including public-facing Azure services.

Configuring Private and Public IP Addresses, Network Routes, Network Interfaces, Subnets, and Virtual Networks

Azure uses **system routes** to direct network traffic between VMs, on-premises networks, and the Internet.

The example below has a **Virtual Network with two subnets**. Communication between the subnets and from the front end to the Internet are all managed using the default system routes.



Configuring Private and Public IP Addresses, Network Routes, Network Interfaces, Subnets, and Virtual Networks

1. Create a Route Table
2. Create a Route
3. Associate the Route to the Subnet

1. Creating a route table is very straightforward, but be mindful of the the **Border Gateway Protocol** (BGP) route propagation setting. Most of the time, we want this enabled because routes will automatically be added to the route table of all subnets with **BGP propagation enabled**.

2. When creating a route there are several next hop types. Your choices are:

   A. **Using virtual appliance.**

   B. **Using Virtual Network Gateway.**

   C. **Internet.**

   D. **None.**

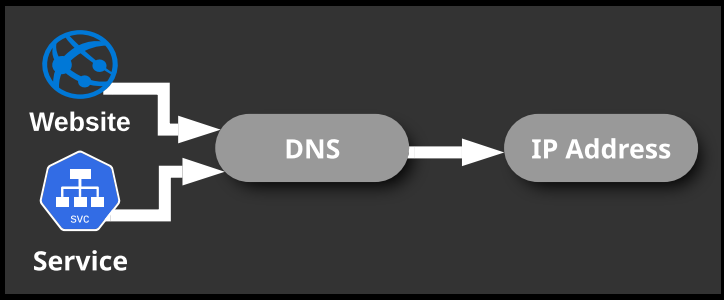
3. Each subnet can have **zero or one route tables** associated to it.

Configure and Manage Virtual Networks

Configure Name Resolution

Configuring Azure DNS

Azure Domain Name System (DNS) is a hosting service for DNS domains. DNS provides name resolution by resolving a website or service name to its IP address.



Elements of DNS

Root Domain: Top level of the DNS hierarchy.

Top-Level Domain: Under the root domain. Examples include `org`, `com`, `gov`, `mil`, etc.

Second-Level Domain: Private domains owned and locally managed (e.g. contoso, adatum).

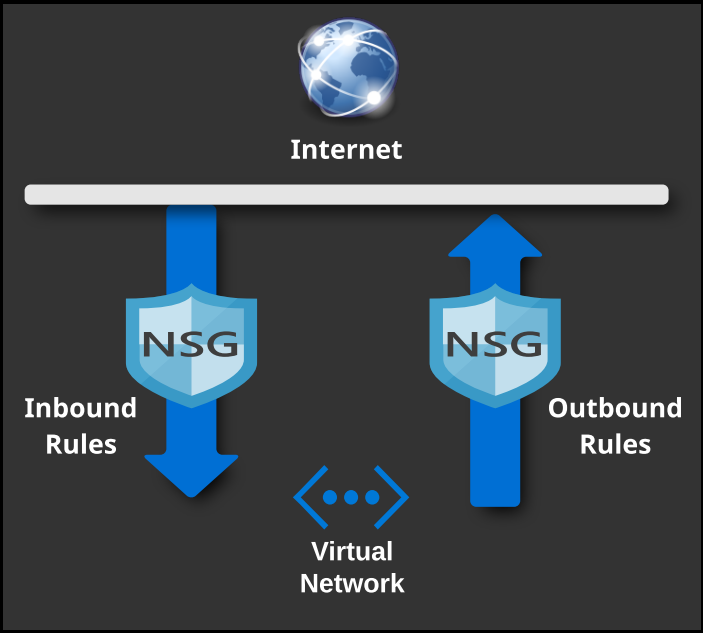
Sub-Domain: Under the second-Level domain (e.g. sales.contoso.com).

DNS
Azure DNS

Individual Machine: The computer where the DNS resolves to (e.g. mail.yahoo.com).

Creating Security Rules

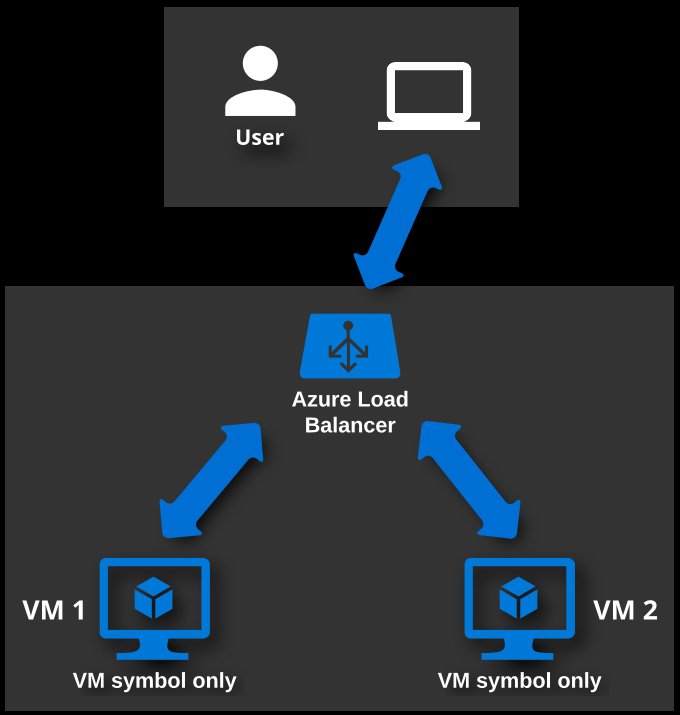
We can limit network traffic to resources in a virtual network using a **network security group (NSG).** A network security group contains a list of security rules allowing or denying inbound or outbound network traffic. An NSG can be associated with a subnet or network interface.



Configure and Manage Virtual Networks

Implement An Azure Load Balancer

Load balancing provides a higher level of availability and scale by spreading incoming requests across virtual machines. We can create the load balancer within the Azure Portal and **it will support inbound and outbound scenario**s. It can be used to distribute traffic internally within the Azure infrastructure.



Configure and Manage Virtual Networks

Monitoring and Troubleshooting VNets

Azure **Network Watcher** provides the tools to monitor, diagnose, view metrics , and enable or disable logs for resources in an Azure virtual network. After enabling the product, we will see four main blades: **Monitoring**, **Network Diagnostic Tools**, **Metrics**, and **Logs**.

Diagnostic Tools

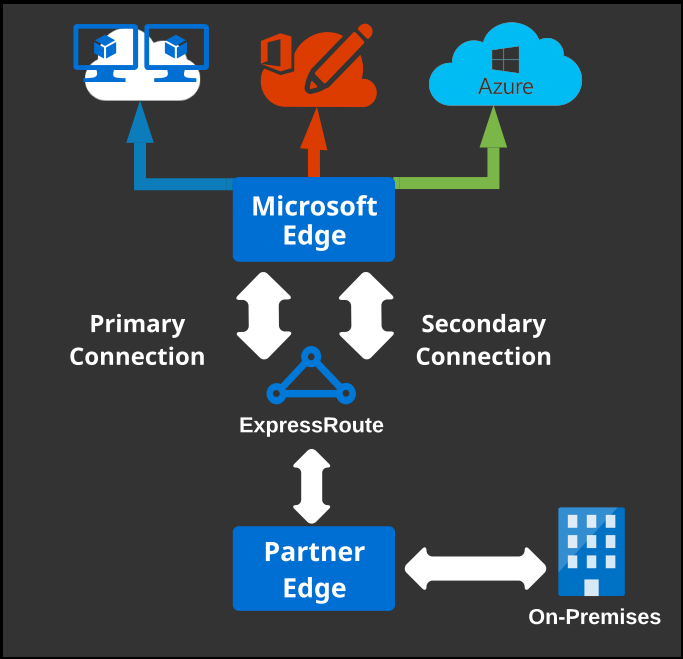
1. **IP Flow Verify**
   Purpose: Quickly diagnose connectivity issues from or to the Internet and the on-premises environment.
2. **Next Hop**
   Purpose: Determine if traffic is being directed to the intended destination by showing the next hop.
3. **VPN Diagnostic**
   Purpose: Troubleshoot gateways and connections.
4. **NSG Views and Flows**
   Purpose: Assess a VM for network vulnerabilities such as open ports.

Configure and Manage Virtual Networks

Integrate On-Premises Network with Azure VNet

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure

Microsoft Azure ExpressRoute lets us extend our on-premises networks into the Microsoft cloud over a dedicated private connection facilitated by a connectivity provider. With ExpressRoute we can **establish connections to Microsoft cloud services** like Microsoft Azure, Office 365, and CRM Online.



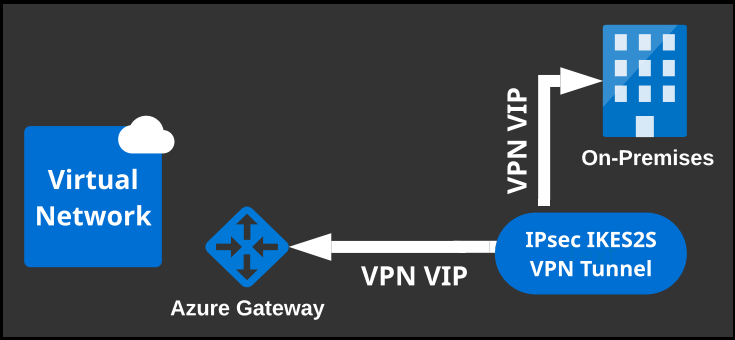
Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure

We can create a connection between our on-premises network and the Microsoft cloud in three different ways: **CloudExchange Co-location, Point-to-point Ethernet Connection, and Any-to-any (IPVPN) Connection**. Connectivity providers can offer one or more connectivity models. We can work with our connectivity provider to pick the model that works best for us.

CloudExchange Co-location	Point-to-point Ethernet Connection	Any-to-any (IPVPN) Connection
If we are co-located in a facility with a cloud exchange, we can order **virtual cross-connections to the Microsoft Cloud** through the co-location provider's Ethernet exchange.	We can connect our on-premises data centers and offices to the Microsoft cloud **through point-to-point Ethernet links**.	We can **integrate our WAN with Microsoft cloud**. IPVPN providers typically offer any-to-any connectivity between branch offices and data centers.

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure

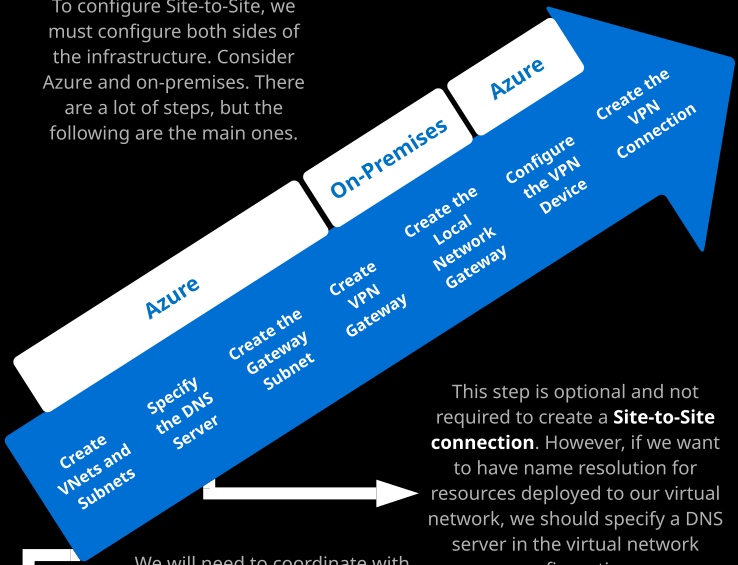
A Site-to-Site (S2S) connection is a connection over an IPsec IKE VPN tunnel. S2S connections can be used for cross-premises and hybrid configurations. This type of connection **requires a VPN device located on-premises** that has a public IP address assigned to it.



Multi-site connection is a variation of the **Site-to-Site connection**. We can create more than one VPN connection from our virtual network gateway, typically connecting to **multiple on-premises sites**.

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure

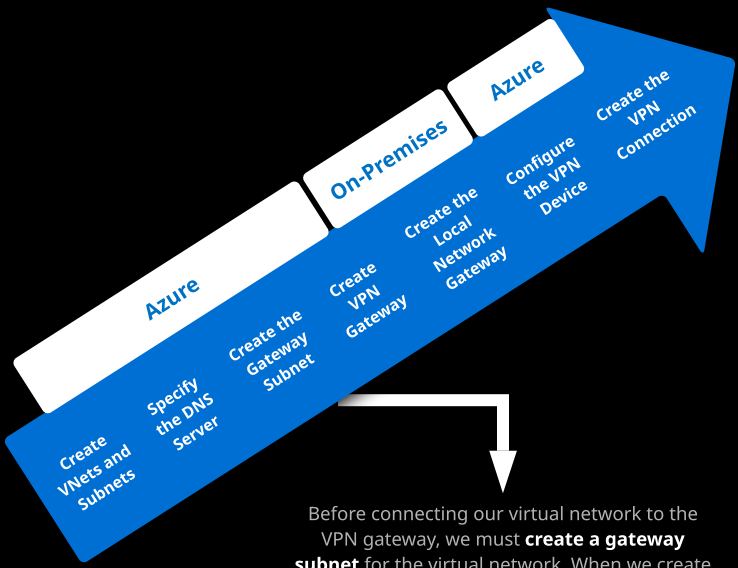
To configure Site-to-Site, we must configure both sides of the infrastructure. Consider Azure and on-premises. There are a lot of steps, but the following are the main ones.



This step is optional and not required to create a **Site-to-Site connection**. However, if we want to have name resolution for resources deployed to our virtual network, we should specify a DNS server in the virtual network configuration.

We will need to coordinate with our on-premises network administrators to **reserve an IP address range** we can use specifically for this virtual network.

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure

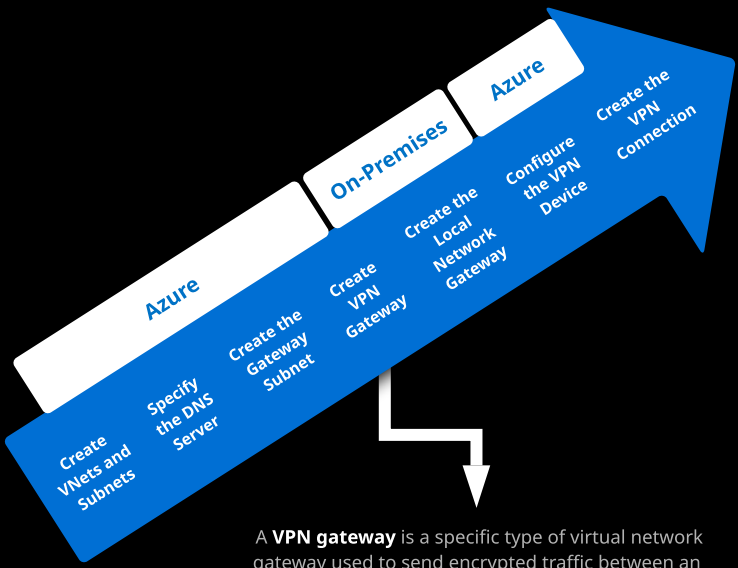


Before connecting our virtual network to the VPN gateway, we must **create a gateway subnet** for the virtual network. When we create a gateway subnet, VMs are deployed and configured with the required VPN gateway settings. We should not deploy any other resources in this subnet.

Back Next

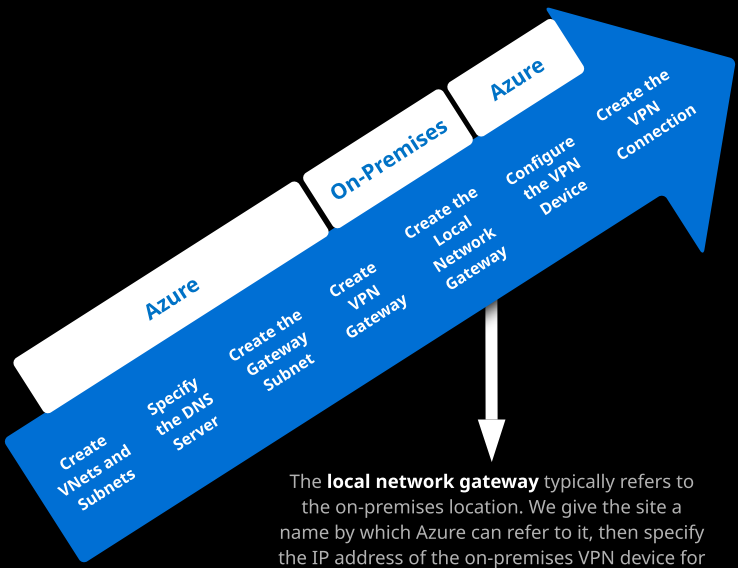
Integrate On-Premises Network with Azure VNet

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure



A **VPN gateway** is a specific type of virtual network gateway used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet.

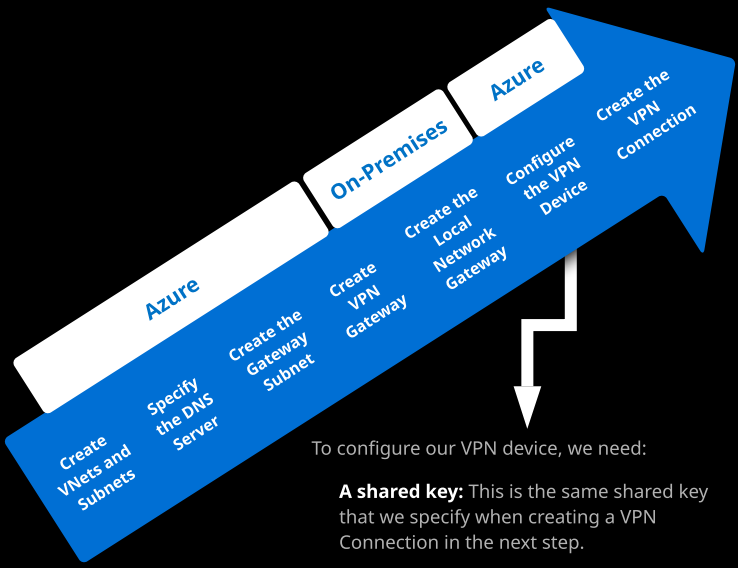
Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure



The **local network gateway** typically refers to the on-premises location. We give the site a name by which Azure can refer to it, then specify the IP address of the on-premises VPN device for the connection.

The **IP address i**n this section is the public IP address of the local gateway.

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure

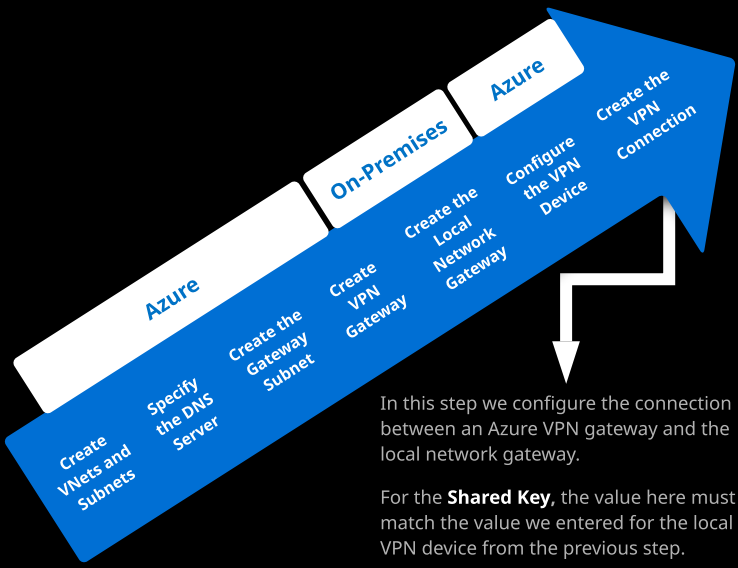


To configure our VPN device, we need:

A shared key: This is the same shared key that we specify when creating a VPN Connection in the next step.

The Public IP address of our VPN Gateway: When we create a VPN gateway we may configure a new public IP address or use an existing IP address.

Creating and Configuring Azure VPN Gateways, Site-to-Site VPN, ExpressRoute, and Troubleshooting On-Premises Connectivity with Azure



In this step we configure the connection between an Azure VPN gateway and the local network gateway.

For the **Shared Key**, the value here must match the value we entered for the local VPN device from the previous step.

This can take some time, but when the connection completes, we will see it appear in the connections blade for our gateway. Hopefully, the status shows **Succeeded** or **Connected**.

Back

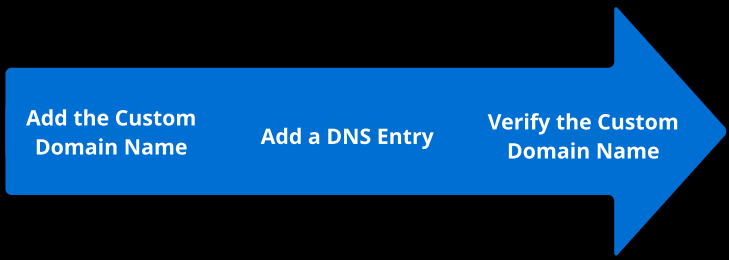
Adding Custom Domains

Initial Domain

By default, when we create an Azure subscription, an Azure AD domain is created for us. The instance of the domain has an initial domain name in the form domainname.onmicrosoft.com.

Custom Domain Name

Although the initial domain name for a directory can't be changed or deleted, we can add any routable custom domain we control.



When an administrator adds a custom domain name to an Azure AD, it is initially in an unverified state. **Azure AD will not allow any directory resources to use an unverified domain name**. This ensures only one directory can use a domain name, and the organization using the domain name owns that domain.

Azure AD verifies ownership of a domain name by looking for an entry in the domain name service (DNS) zone file for the domain. To verify ownership of a domain name, **an admin gets the DNS entry from Azure AD** that Azure AD will look for, and it adds that entry to the DNS zone file for the domain name.

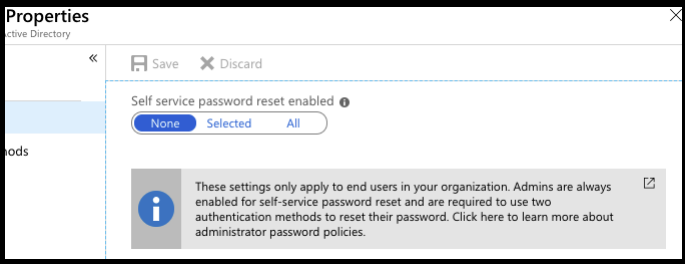
Next

Configuring Self-Service Password Reset

To configure a self-service password reset, we first determine who will be enabled to use self-service password reset. From our existing Azure AD tenant, access the Azure Portal and select **Password reset** under **Azure Active Directory**. In the password reset properties, there will be three options: **None**, **Selected**, and **All.**

The selected option is useful for creating specific groups who have self-service password reset enabled.

Azure Administrator accounts will always be able to reset their passwords regardless of the setting for this option.

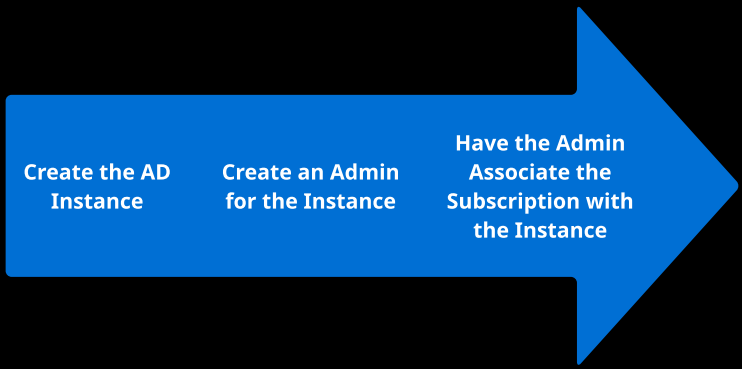


After **enabling password reset** for users and groups, we pick the number of authentication methods required to reset a password and the number of **authentication methods** available to users.

At least one authentication method is required to reset a password, but it is a **good idea to have additional methods available**. We can choose from email notification, a text or code sent to a user's mobile or office phone, or a set of security questions.

Manage Multiple Directories

A tenant represents an organization in Azure Active Directory. It is a dedicated instance of **Azure AD** that an organization receives and owns when it signs up for a Microsoft cloud service like Azure or Office 365. To use a tenant, **it must be associated with a subscription**. The basic steps are shown below.



In **Azure Active Directory** (Azure AD), each tenant is a fully independent resource. There is no parent-child relationship between tenants. **This independence between tenants includes resource independence**, administrative independence, and synchronization independence.

Creating Users and Groups

In Azure AD, **all users requiring access to resources must have a user accoun**t. A user account is an Azure AD user object that contains all the information required to authenticate and authorize the user during the sign-in-process and build the user's access token.

For the user accounts, there are multiple different sources depending on the types of identity.

Types of Identities

Cloud identities: Users that only exist in Azure AD.

Directory-synchronized identities (Windows Server AD): Users brought in to Azure through a sync activity using Azure AD Connect.

Guest users (Azure Active Directory): Users from outside Azure.

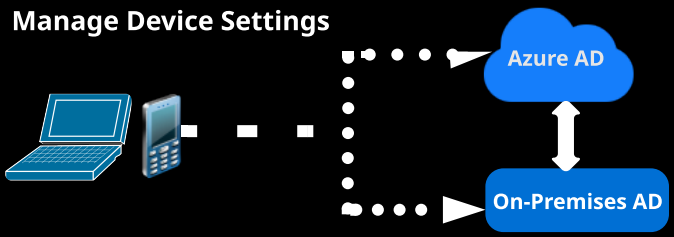
A group helps organize users to make it easier to manage permissions. Groups can be easily added through the portal. There are two types of groups: security groups and distribution groups.

Type of Groups

Security groups: Security-enabled and used to assign permissions and control access to various resources.

Distribution groups: Used mainly by email applications and not security enabled. We can easily add these groups in the Portal.

Manage Device Settings



If our environment has an **on-premises AD footprint** and we also want to benefit from the capabilities provided by Azure AD, we can implement hybrid Azure AD joined devices. These are **devices joined** to our on-premises Active Directory and our Azure AD.

Registered, AD Joined, and Hybrid Joined Devices

Registered Devices	**Device Type (Personal)** Registration is manual and the Windows OS is Windows 10.
AD Joined	**Device Type (Organization Owned)** Registration is manual and Windows OS is Windows 10.
Hybrid Joined Devices	**Device Type (Organization Owned)** Registration is automatic and Windows OS is Windows 7, 8, and 10.

Performing Bulk Updates

There are several ways you can use **PowerShell to import data into your directory**, but the most commonly used method is to use a CSV file.

Steps for Using the CSV File

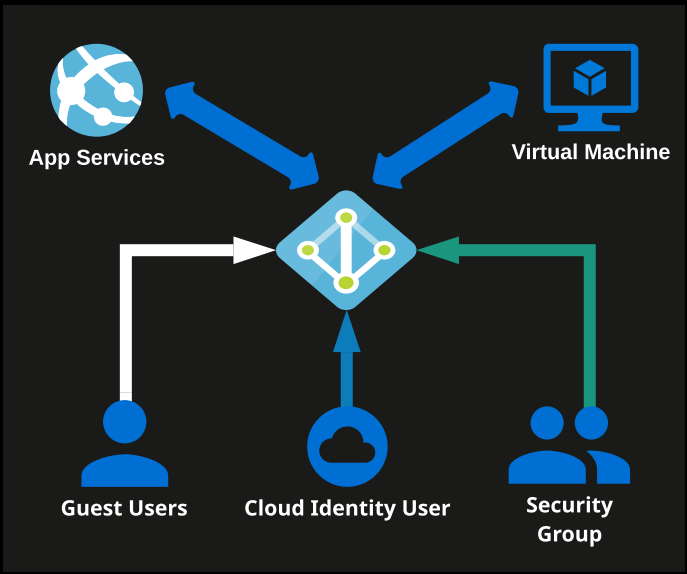
Back Next

Managing Guest Users

Guest users are external users with an email address outside of Azure AD.

Requirements for guest account:

1. Email address **needs to be valid** as the user will need to follow the instructions from the invitation email.
2. We will need to use a role that **allows us to create users** in the tenant directory (i.e. Global Administrator).



Performing Bulk Updates

There are several ways you can use **PowerShell to import data into your directory**, but the most commonly used method is to use a CSV file.

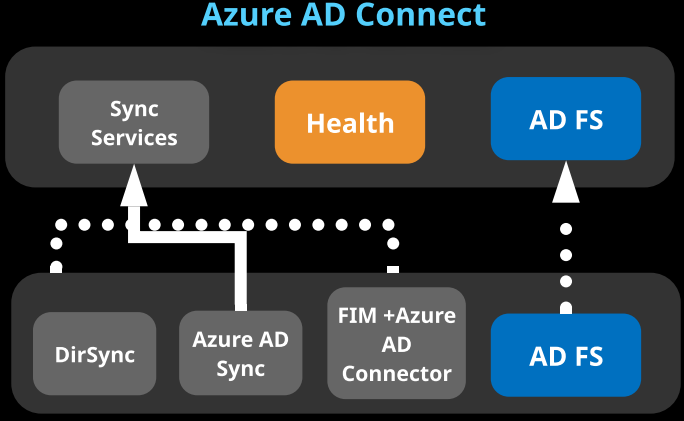
Steps for Using the CSV File

Step 2:

Create a new **Password Profile** for the new users.

Installing Azure AD Connect, Including Password Hash and Pass-Through Synchronization

Azure AD Connect integrates our on-premises directories with Azure Active Directory. This allows us to provide a common identity for our users for Office 365, Azure, and SaaS applications integrated with Azure AD.



Sync Services
This component is responsible for creating users, groups, and others.

Health Monitor
Azure AD Connect Health can provide monitoring and gives a central location to view this activity.

AD FS
Federation is optional in Azure AD Connect and can be used to configure hybrids using an on-premises AD FS infrastructure.

Back Next

Performing Bulk Updates

There are several ways you can use **PowerShell to import data into your directory**, but the most commonly used method is to use a CSV file.

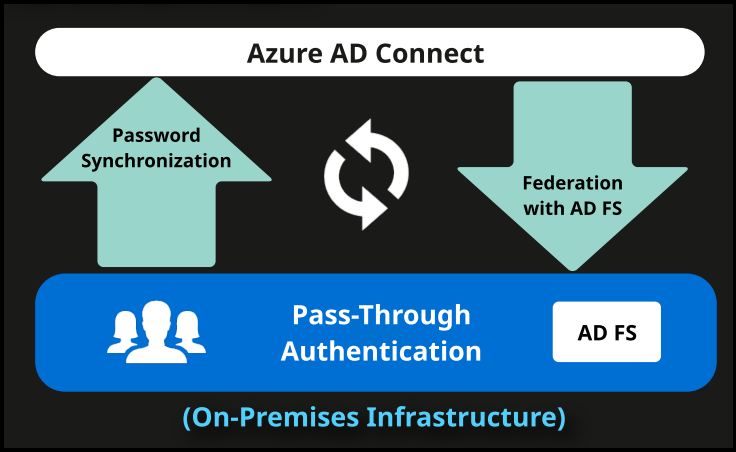
Steps for Using the CSV File

Step 3:

Use **Import-CSV** to import the CSV file.

Using Azure AD Connect to Configure Federation With On-Premises AD DS

AD Connect provides several sign-on methods: **Password Synchronization**, **Pass-Through Authentication**, and **Federation with AD FS**. These methods are used to synchronize user accounts, and optionally, passwords from an on-premises Active Directory instance to a cloud-based Azure AD instance.



Password Synchronization
This option can be used to synchronize an encrypted version of the password hash for users.

Pass-Through Authentication
With this option, the username and password are authenticated by on-premises DCs.

Federation with AD FS
Microsoft's implementation of an identity federation solution that uses claims-based authentication.

Performing Bulk Updates

There are several ways you can use **PowerShell to import data into your directory**, but the most commonly used method is to use a CSV file.

Steps for Using the CSV File

Step 1:

Use **Connect-AzureAD** to create a PowerShell connection to your directory.

Performing Bulk Updates

There are several ways you can use **PowerShell to import data into your directory**, but the most commonly used method is to use a CSV file.

Steps for Using the CSV File

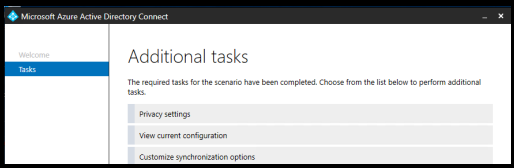
Step 4:

Loop through the users in the file **constructing the user parameters** required for each user.

Back Next

Managing Azure AD Connect

These are the **procedural tasks** available to configure when managing Azure AD Connect.



Privacy Settings
View what telemetry data is being shared with Microsoft.

View Current Configuration
View our current AD connection solution.

Customize Synchronization Options
Change the current configuration (e.g. adding Active Directory forests).

Configure Device Options
Set up the device options available for synchronization.

Refresh Directory Schema
Add new on-premises directory objects for synchronization.

Configure Staging Mode
Preview the synchronizations before they occur.

Change User Sign-in
Change the authentication method users are using to sign-in.

Manage Federation
Manage our AD FS infrastructure, renew certificates, and add AD FS servers.

Troubleshoot
Help with troubleshooting Azure AD Connect issues.

Performing Bulk Updates

There are several ways you can use **PowerShell to import data into your directory**, but the most commonly used method is to use a CSV file.

Steps for Using the CSV File

Step 5:

Use **New-ADUser** to create each user.

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

[Steps to Enable Password Writeback]

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

Steps to Enable Password Writeback

Step 1:

Sign into Azure AD Connect server and start the **Azure AD Connect** configuration wizard.

Configuring User Accounts for MFA, MFA Using Bulk Update, Fraud Alerts, Bypass Options, Trusted IPs, and Verification Methods

For organizations that need to be compliant with industry standards, such as **PCI DSS version 3.2**, MFA is a must-have capability to authenticate users. Beyond being compliant with industry standards, enforcing MFA to authenticate users can also help organizations **mitigate credential theft attacks**. MFA provides additional security by requiring a second form of authentication and delivers strong authentication through easy-to-use authentication methods.

Authentication methods:

1.  Something you know (e.g., a **password**).
2.  Something you have (e.g., a **trusted device** not easily duplicated, such as a phone).
3.  Something you are (e.g., **biometrics**).

Back Next

Configuring User Accounts for MFA, MFA Using Bulk Update, Fraud Alerts, Bypass Options, Trusted IPs, and Verification Methods

MFA Bypass Options:
Trusted IPs is a feature to allow federated users or IP address ranges to bypass two-step authentication.

The **one-time bypass** feature allows a user to authenticate a single time without performing two-step verification. The bypass is temporary and expires after a specified number of seconds.

Verification Methods:

1. 
   A code provided in an email or text message.
2. 
   A phone call or a notification code on their phone.
3. 
   Answers to security questions.

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

Steps to Enable Password Writeback

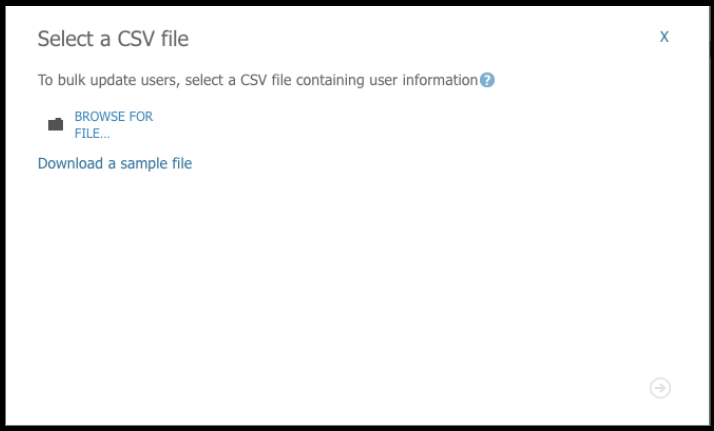
Step 2:

On the **Welcome** page, select **Configure**.

Configuring User Accounts for MFA, MFA Using Bulk Update, Fraud Alerts, Bypass Options, Trusted IPs, and Verification Methods

To enable MFA, go to **Users** under **Manage** in Azure Active Directory and then the **Multi-Factor Authentication** option. From there you can select the user to modify and select **Enable for MFA**. We can also bulk enable groups of users via PowerShell or with a CSV file in the Azure Portal.

```
Get-MsolUser -All | Foreach{ Set-MsolUser
-UserPrincipalName $_.UserPrincipalName
-StrongAuthenticationRequirements $auth}
```



Back Next

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

Steps to Enable Password Writeback

Step 3:

On the **Additional tasks** page, select **Customize synchronization options**, and then select next.

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

Steps to Enable Password Writeback

Step 4:

On the **Connect to Azure AD** page, enter a global administrator credential, and then select **Next**.

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

Steps to Enable Password Writeback

Step 5:

On the **Connect directories** and **Domain/OU** filtering pages, select **Next**.

Managing Password Sync and Password Writeback

With **password writeback**, we can configure Azure Active Directory (Azure AD) to write passwords back to our on-premises Active Directory. **Password writeback** removes the need to set up and manage a complicated on-premises self-service password reset (SSPR) solution, and it provides a convenient cloud-based way for users to reset their on-premises passwords wherever they are.

Steps to Enable Password Writeback

Step 6:

On the **Optional features** page, select the box next to **Password writeback** and select **Next**.

Course Navigation

Manage Azure Subscriptions and Resources

Implement and Manage Storage

Deploy and Manage Virtual Machines

Configure and Manage Virtual Networks

Manage Identities

The AZ-103: Microsoft Azure Administrator Exam

About the Exam

- Length: 180 Minutes
- Number of Questions: 50-65
- Cost: $165 (minus any applicable discounts)
- Format: Case Study, Drag and Drop, Exhibits, True or False, Labs
- URL to Schedule the Exam: https://www.microsoft.com/en-us/learning/exam-az-103.aspx

The exam can be taken at a local test center, at your home or office, or at a Certiport test center. If you choose to take the exam at home or in an office, please review the system requirements at: https://www.microsoft.com/en-us/learning/online-exams.aspx

How to Prepare for the Exam

- Watch and follow along with all video lessons.
- Complete all hands-on labs at least twice.
- Take and pass the practice exam at least twice.
- Understand concepts from the instructor flash card deck, and make your own. Take notes during the course video lessons.
- Participate in Linux Academy Community and a Linux Academy Study Group, or start your own.

Back to Main