Microsoft Azure

Azure Study Group
AZ-300 - Microsoft Azure Architect Technologies

Jeff Wagner

Partner Technology Strategist

Microsoft Azure

Implement Workloads and Security (20-25%)

Agenda

1. Series Agenda
2. Speaker Introduction
3. Feedback Loop
4. Objective Review
5. Open Mic

Series Agenda

1	Deploy and Configure Infrastructure (25-30%)
2	Implement Workloads and Security (20-25%)
3	Create and Deploy Apps (5-10%)
4	Implement Authentication and Secure Data (5-10%)
5	Develop for the Cloud (20-25%)

https://aka.ms/azurecsg

Series Agenda

1	Deploy and Configure Infrastructure (25-30%)
2	Implement Workloads and Security (20-25%)
3	Create and Deploy Apps (5-10%)
4	Implement Authentication and Secure Data (5-10%)
5	Develop for the Cloud (20-25%)

https://aka.ms/azurecsg

Speaker Introduction - Jeff Wagner

- Partner Technology Strategist based in Atlanta
- 21+ years with Microsoft, more in the industry
- Been working with Microsoft Azure when we weren't sure if the was called Windows A*zure* or Windows *Azure*
- Constant learner - *Ancora Imparo*

Microsoft Azure

Feedback Loop

Objectives

Migrate servers to Azure

May include but not limited to: Migrate by using Azure Site Recovery (ASR); migrate using P2V; configure storage; create a backup vault; prepare source and target environments; backup and restore data; deploy Azure Site Recovery (ASR) agent; prepare virtual network

Configure serverless computing

May include but not limited to: Create and manage objects; manage a Logic App resource; manage Azure Function app settings; manage Event Grid; manage Service Bus

Implement application load balancing

May include but not limited to: Configure application gateway and load balancing rules; implement front end IP configurations; manage application load balancing

Objectives (cont.)

Integrate on premises network with Azure virtual network

May include but not limited to: Create and configure Azure VPN Gateway; create and configure site to site VPN; configure Express Route; verify on premises connectivity; manage on-premise connectivity with Azure

Manage role-based access control (RBAC)

May include but not limited to: Create a custom role; configure access to Azure resources by assigning roles; configure management access to Azure; troubleshoot RBAC; implement RBAC policies; assign RBAC roles

Implement Multi-Factor Authentication (MFA)

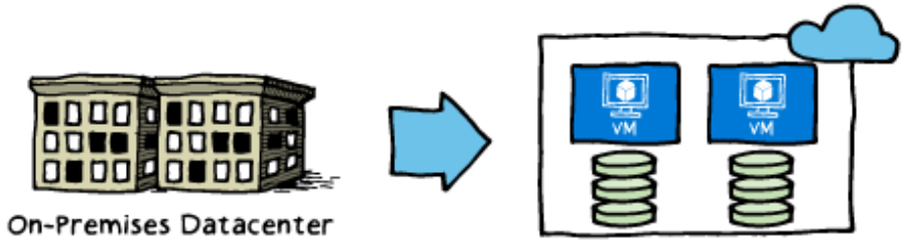
May include but not limited to: Enable MFA for an Azure tenant; configure user accounts for MFA; configure fraud alerts; configure bypass options; configure trusted IPs; configure verification methods; manage role-based access control (RBAC); implement RBAC policies; assign RBAC Roles; create a custom role; configure access to Azure resources by assigning roles; configure management access to Azure

Microsoft Azure

Migrate servers to Azure

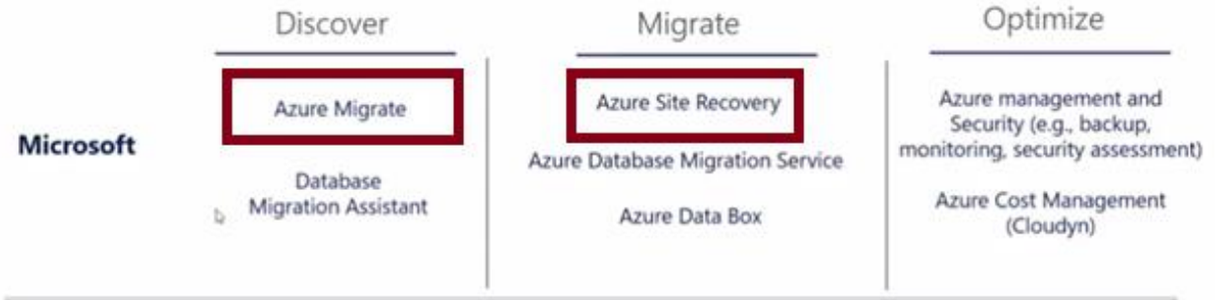
Migration Goals

- **Technology-focused and business-focused, including:**
  - **Addressing the hardware obsolescence cycle**
  - **Moving away from the ‘pre-purchase capacity’ model**
  - **Lack of IT agility**
  - **Desire to re-focus on core competencies**
  - **Expense of maintaining a global presence**
  - **Enable disaster-recovery scenarios**



Migration Phases

- **When planning for migration to Azure, consider the following phases:**
  - **Discover: gain better visibility of on-premises workloads and assess the optimal resource level to run them in Microsoft Azure.**
    - **Azure Migrate is the primary tool for this, and includes:**
      - **Automated server, app, and database discovery.**
      - **Intelligent workload right-sizing and costing for maximum ROI.**
      - **Workload configuration analyses and recommendations.**
  - **Migrate: move selected workloads to Azure.**
    - **Azure Site Recovery is the primary tool for this and includes:**
      - **Lifting and shifting of servers, apps, databases, and data.**
      - **Containerization of existing applications and infrastructure**
      - **Modernization options for apps and databases.**
  - **Optimize: fine tune your Azure-based workloads and maximize your ROI.**
    - **There are many Microsoft partners to help you with backup, monitoring, security assessments, and cost management.**



Azure Migrate

Azure Migration Service

Provides assessment of on-premises workloads for migration to Azure:

migration suitability of on-premises machines

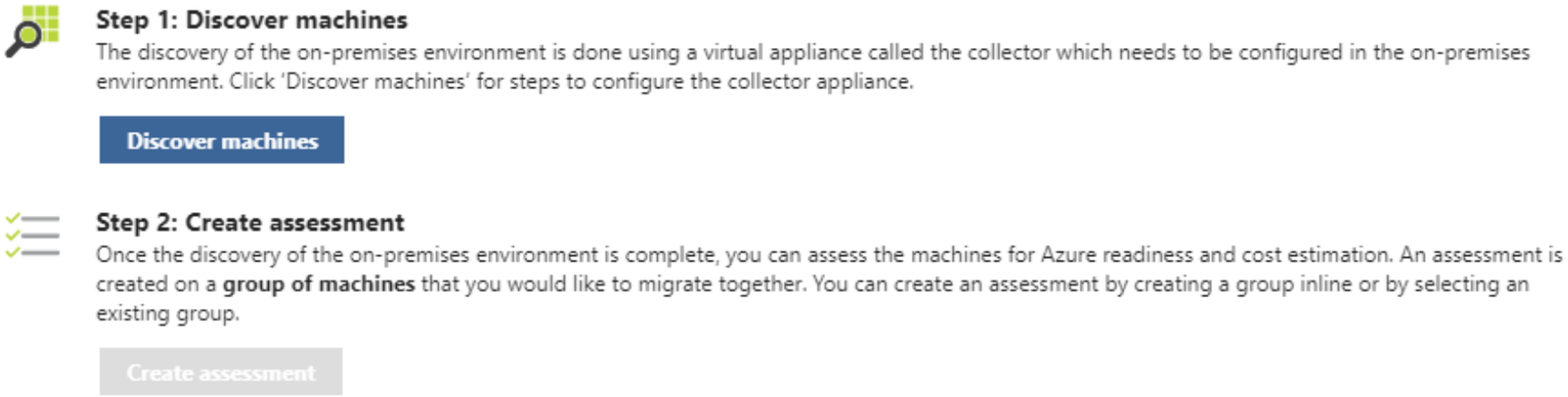
performance-based sizing

cost estimates for running your on-premises machines on Azure VMs

Follows two steps:

Discover machines

Create Assessments



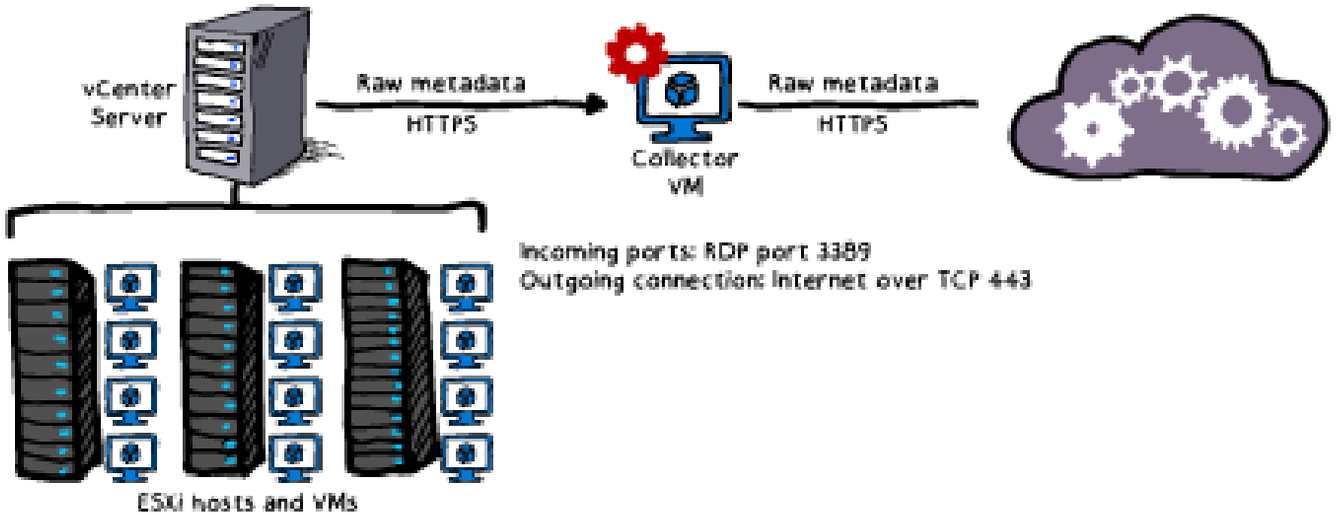
Using Azure Migrate: A Look ahead

Architecture:

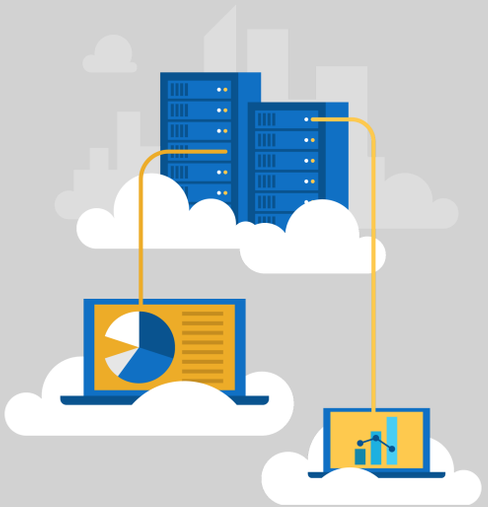
Closely integrated with the components of an VMware vCenter environment

Process

Create a project
Discover the machines
Collect the information
Assess the project



Azure Migrate Process



Creating a Project

The initial step of the Azure Migrate

A project contains metadata representing on-premises environment:

Name

Subscription

Resource Group

Location

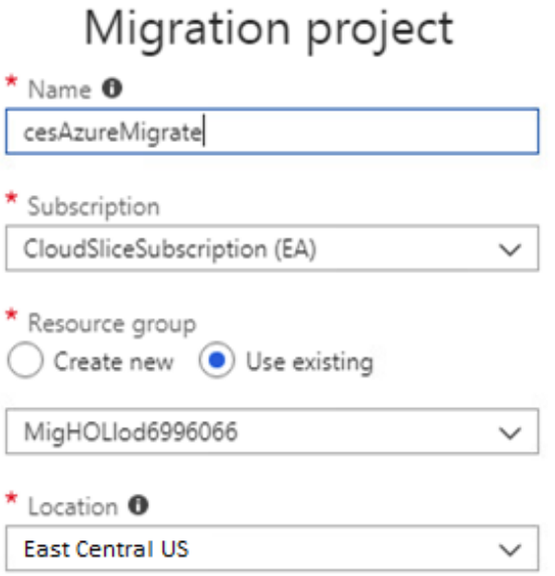
Discovery records of VMware VMs

Project limits:

Up to 1,500 discovered VMs per project

Up to 1,500 assessed VMs per project

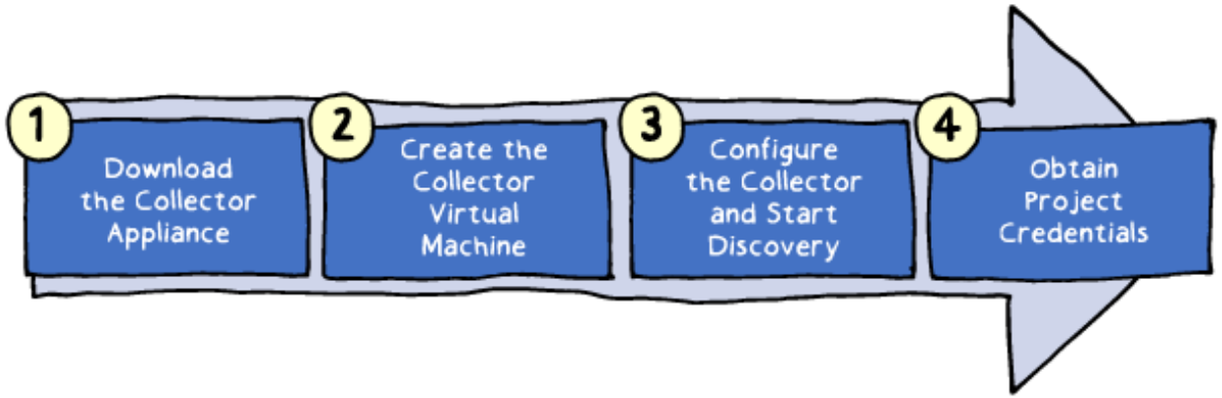
Up to 20 projects per subscription



Creating a Collector

A virtual appliance which handles discovery
Implementation consists of four main steps:

Downloading the Collector appliance (an Open Virtualization Appliance (.ova) file downloadable from the Azure Migrate project in the Azure portal)
Creating the Collector virtual machine by importing the .ova file on the vCenter server
Configuring the Collector to initiate discovery
Assigning to the Collector project credentials including Azure Migrate project ID and key



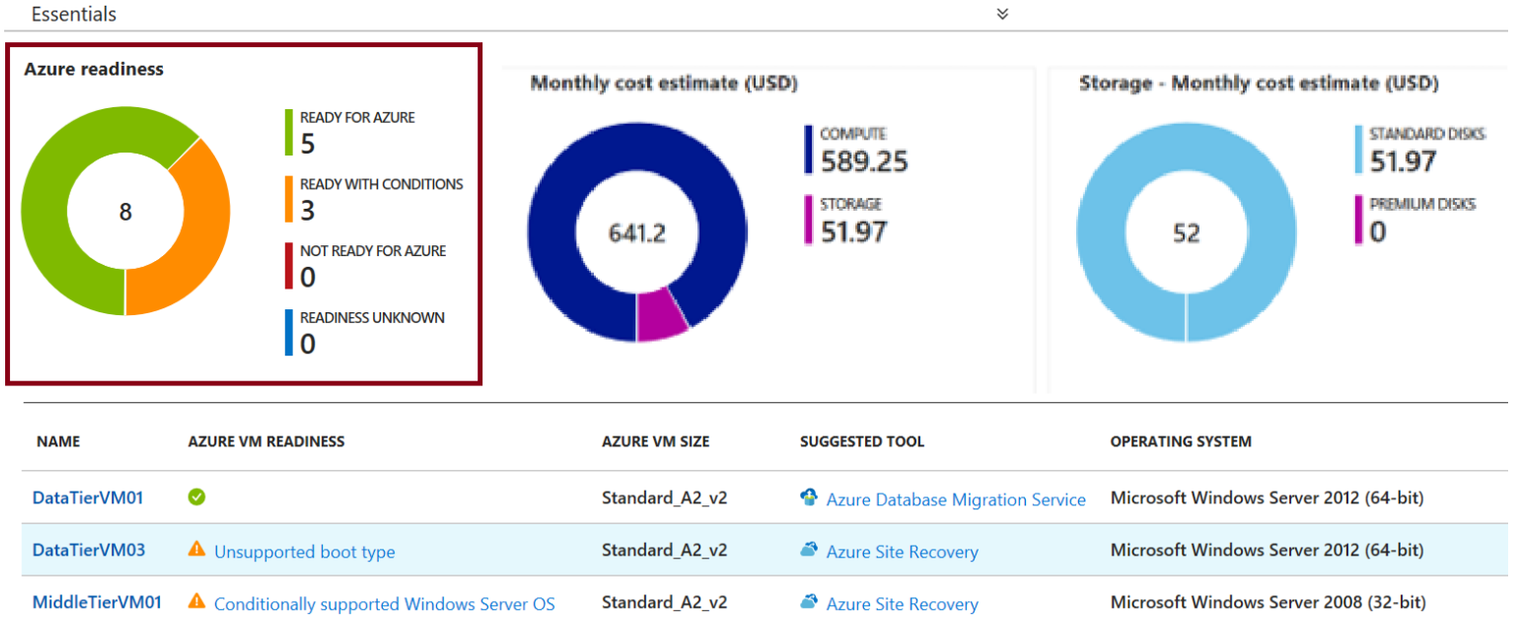
Assessing Readiness

Assessment is based on the readiness status of discovered VMs:

Ready for Azure (green): along with the recommended Azure VM size

Ready with conditions (Orange) and Not ready for Azure (Red): including readiness issues and remediation steps.

Readiness unknown (Blue)



Assessing VM Sizing

Azure Migrate VM assessment offers two types of sizing :

Performance-based sizing (default) which takes into account:

Storage: maps the size and performance of VM disks to Azure VM disks

Network: identifies Azure VM sizes that offer matching number and performance of network adapters

Compute: determines CPU and memory requirements based on performance history of discovered VMs

On-premises sizing:

Matches the size of on-premises VM to an equivalent Azure VM

Does not take into account performance history of discovered VMs

Estimating Cost

Provides the total costs of Azure VMs:

Compute: aggregated monthly cost, which takes into account:

OS type

Software Assurance

Reserved Instances

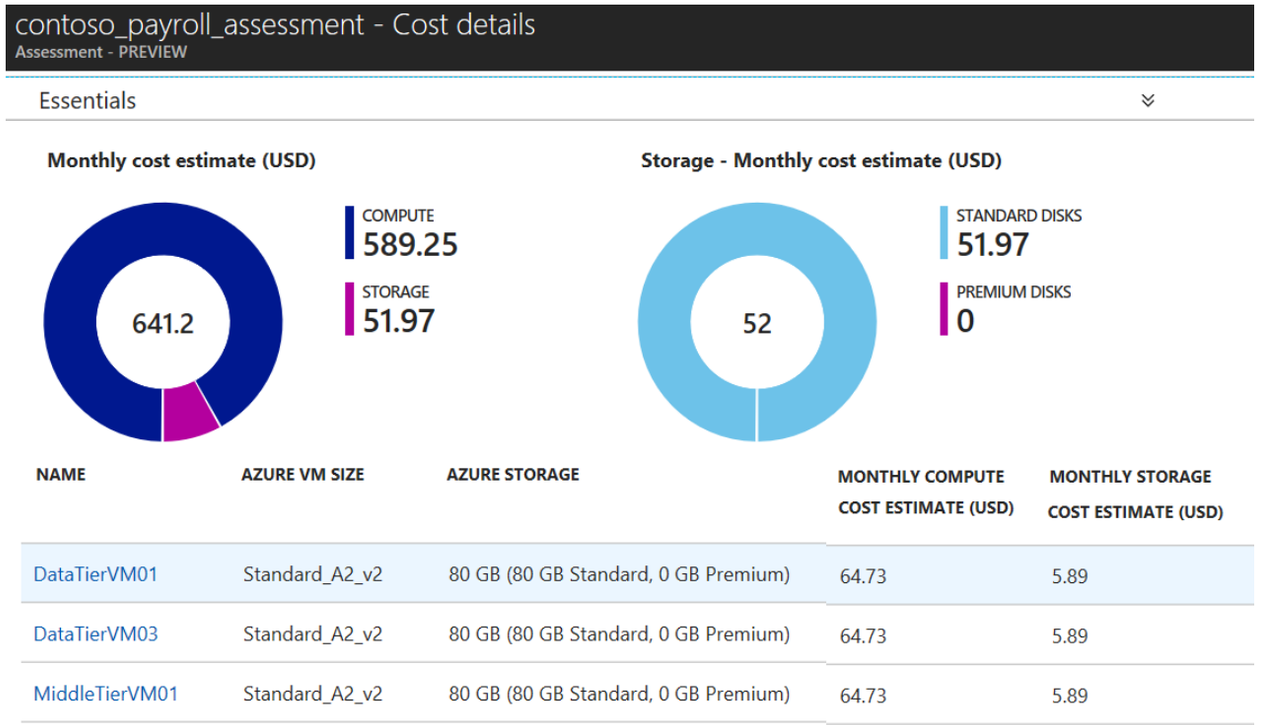
VM uptime

VM location

currency settings

Storage: aggregated monthly cost.

No offer specific settings



Customizing the Assessment

Customization settings include:

Performance history duration
Target location
Pricing tier
Storage type
Comfort factor
Currency
Discounts
VM uptime

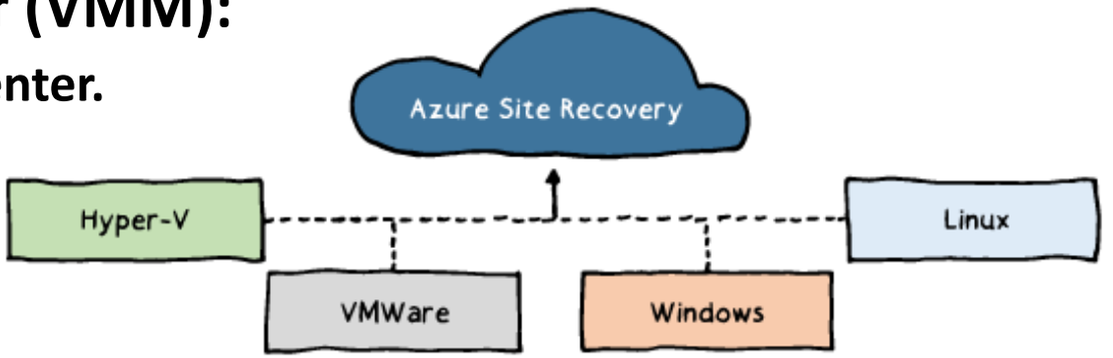
Setting	Details	Default
Target location	The Azure location to which you want to migrate. Azure Migrate currently supports 30 regions.	West US 2 is the default location.
Pricing tier	You can specify the pricing tier (Basic/Standard) for the target Azure VMs.	By default the Standard tier is used.
Storage type	You can specify the type of disks you want to allocate in Azure.	The default value is Premium managed disks.
Comfort factor	Azure Migrate considers a buffer (comfort factor) during assessment. This buffer is applied on top of machine utilization data for VMs.	Default setting is 1.3x.

Troubleshooting Azure Migrate

- **The most common issues include:**
  - **Migration project creation failed**
  - **No performance data**
  - **Collector is not able to connect to the internet**
  - **Date and time synchronization error**
  - **Error UnableToConnectToServer**

ASR Scenarios

- **Hyper-V VM replication:**
  - **Hyper-V with Virtual Machine Manager (VMM):**
    - **Replication to Azure or a secondary datacenter.**
  - **Hyper-V without VMM:**
    - **Replication to Azure only.**
- **VMware VM replication:**
  - **Replication to a secondary site running VMware or to Azure.**
- **Physical Windows and Linux machines:**
  - **Replication to a secondary site running VMware or to Azure.**

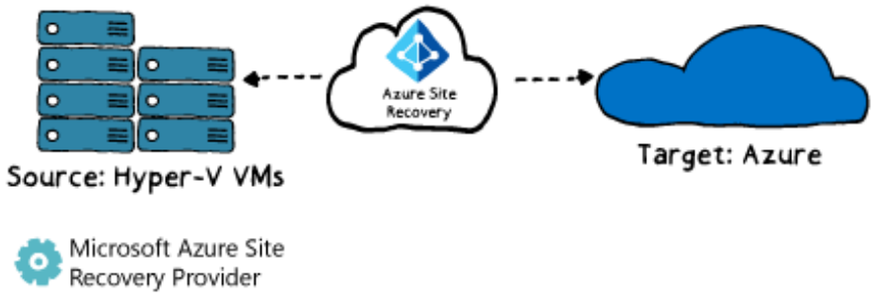


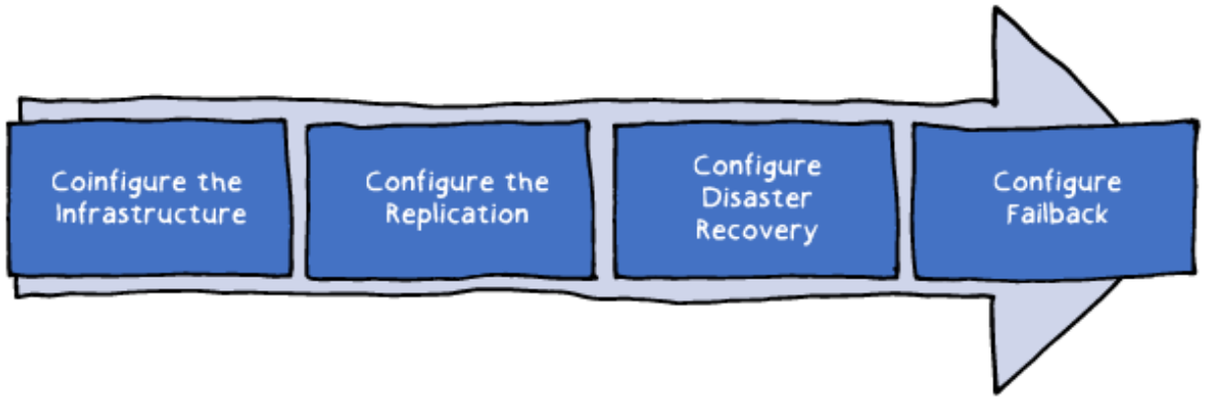
ASR Features

- **Eliminating the need for disaster recovery sites**
- **Reducing infrastructure costs**
- **Automatically replicating to Azure**
- **Safeguarding against outages of complex workloads**
- **Extending or boosting capacity**
- **Continuous health monitoring**

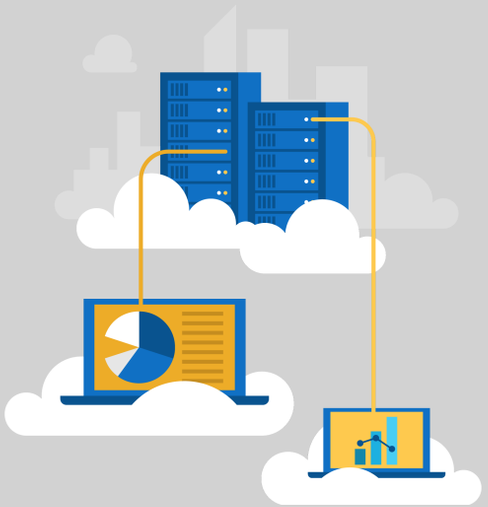
Using ASR: A Look Ahead

- Architecture:
  - **The primary and a secondary site**
  - **On-premises or in Azure**
- Process:
  - **Configure the Infrastructure**
  - **Configure the Replication**
  - **Configure Disaster Recovery**
  - **Configure Failback**





Preparing the Infrastructure



Configuring Azure

Delegate permissions within the target Azure subscription:

Virtual Machine Contributor built-in role

Site Recovery Contributor built-in role

Create an Azure Storage account:

General Purpose

Read-access geo-redundant

Secure transfer: Disabled

Create an Azure virtual network:

Ensure sufficient bandwidth for intersite replication

ASR Deployment Planner

A command line tool for profiling Hyper-V VMs to estimate:

Intersite network bandwidth requirements

Azure Storage requirements

Generates a report including:

On-premises summary

Recommendations

VM storage placement

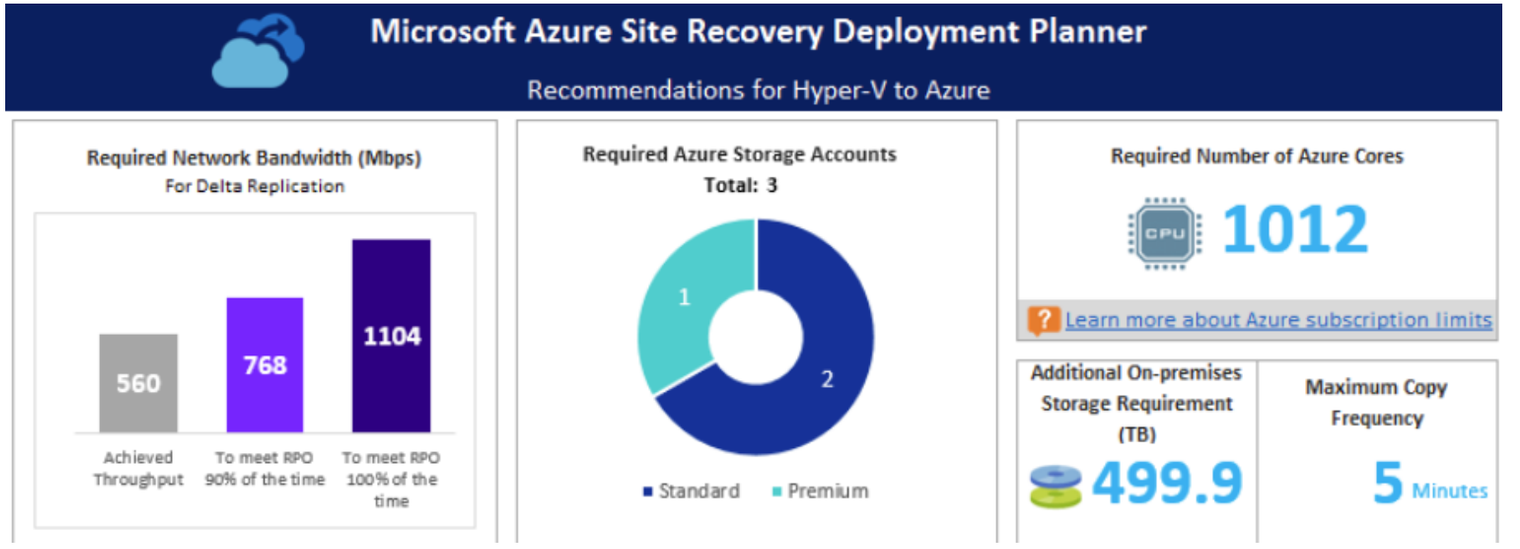
Compatible VMs

Incompatible VMs

Storage requirement

Initial Replication batching

Cost estimate



Configure the Infrastructure: Requirements

Verify Hyper-V operating system requirements:

Windows Server 2016 (including server core installation)

Windows Server 2012 R2 with latest updates.

Mixing hosts running Windows Server 2016 and 2012 R2 isn't supported.

Verify Hyper-V host and guest storage requirements:

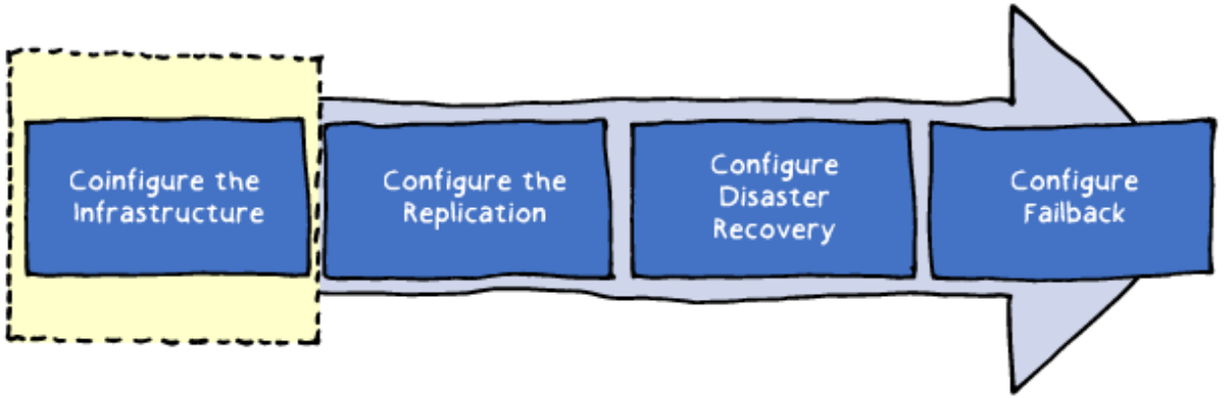
Hyper-V host storage can include SMB 3.0, SAN (iSCSI), and Multi-path (MPIO).

No support for shared cluster disks and encrypted disks.

Verify internet access:

Hyper-V hosts should have direct access to the internet without using a proxy.

Prepare Windows VMs for access following a failover (optional)



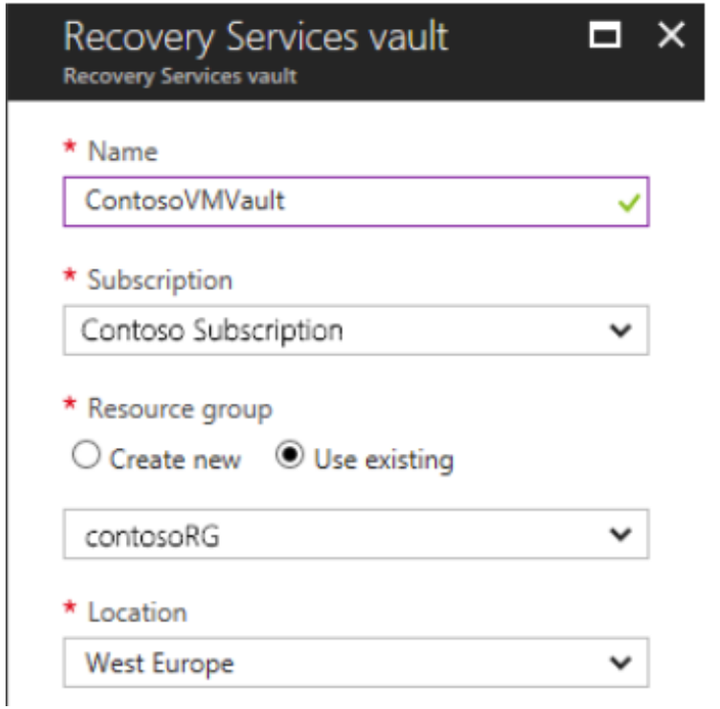
Configure the Infrastructure: Process

The Azure portal driven sequence:

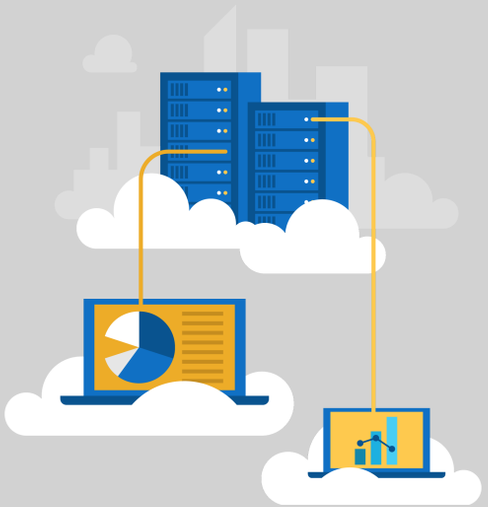
1. **Protection goal (e.g. replicating Hyper-V machines to Azure)**
2. **Source environment (e.g. Hyper-V site)**
3. **Target environment (e.g. a resource group in an Azure subscription)**
4. **Replication settings**
5. **Deployment planning**

Recovery Services Vault

Hosts ASR metadata and replicated VM disks
Can be created directly from the Azure portal
Serves as the primary configuration interface



Completing the Migration Process



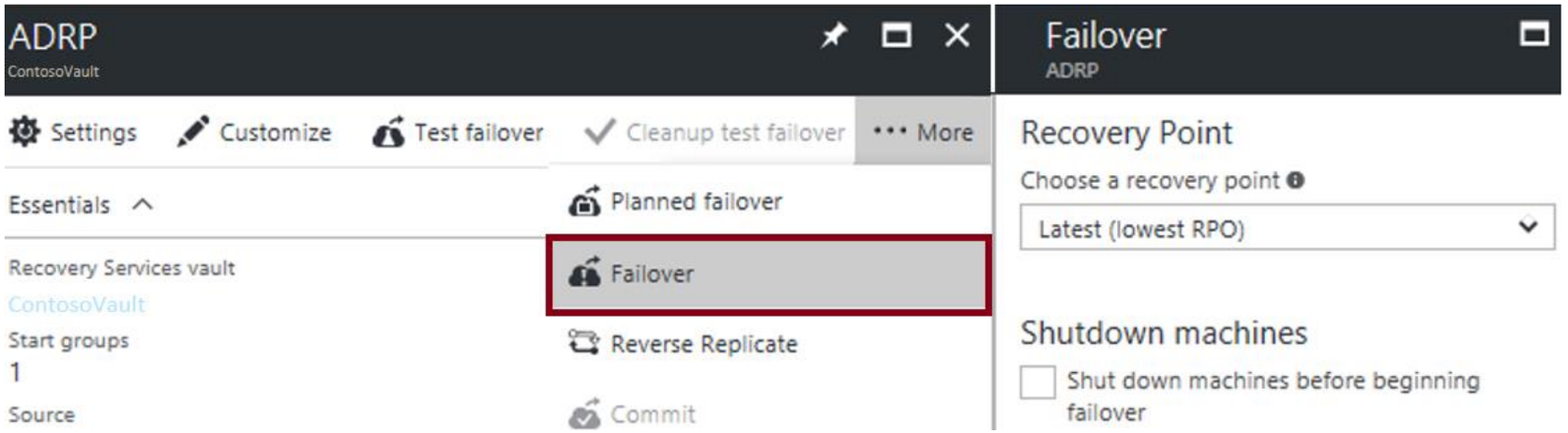
Customize the Recovery Plan

- Select a replication goal
- Set up the source and target environment
- Set up a replication policy
- Enable replication
- Run a test migration to make sure everything's working as expected
- Run a one-time failover to Azure

Planned and Unplanned Failovers

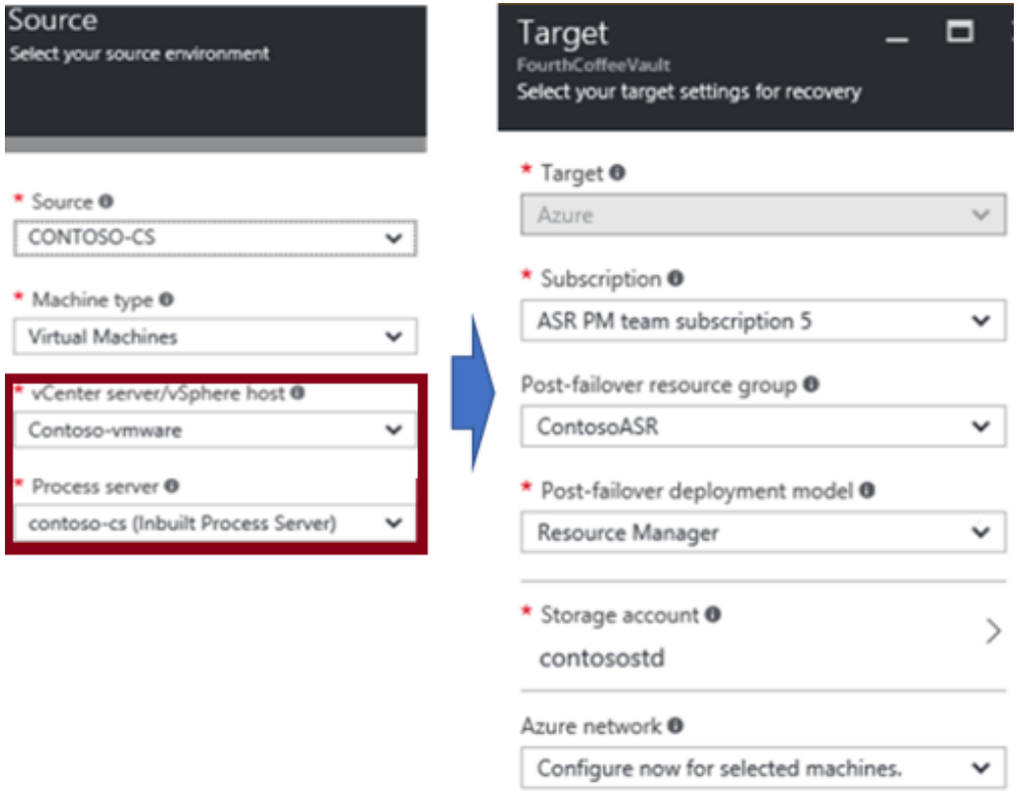
- **ASR facilitates two types of failover:**
  - **Planned: guaranteed no data loss**
  - **Unplanned: possible data loss**
- **ASR failovers:**
  - **are not automatic**
  - **can be initiated from the Azure pc**

Scenario	Requirement
Planned failover due to an upcoming datacenter downtime	Zero data loss for the application when a planned activity is performed
Failover due to an unplanned datacenter downtime (natural or IT disaster)	Minimal data loss for the application



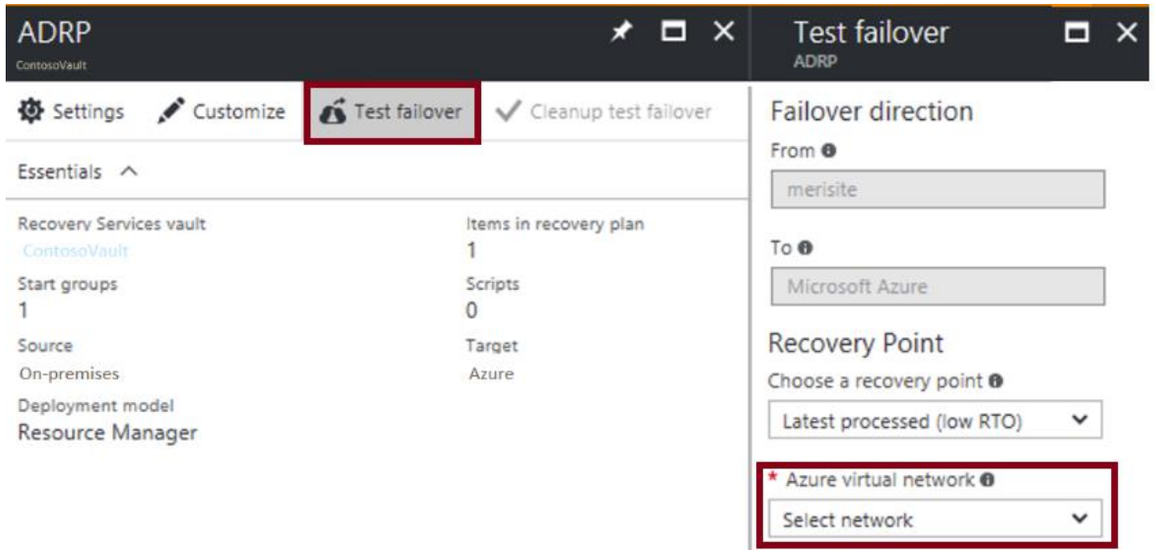
Enable Replication

- **Start by configuring:**
  - **Replication source**
  - **Replication target**



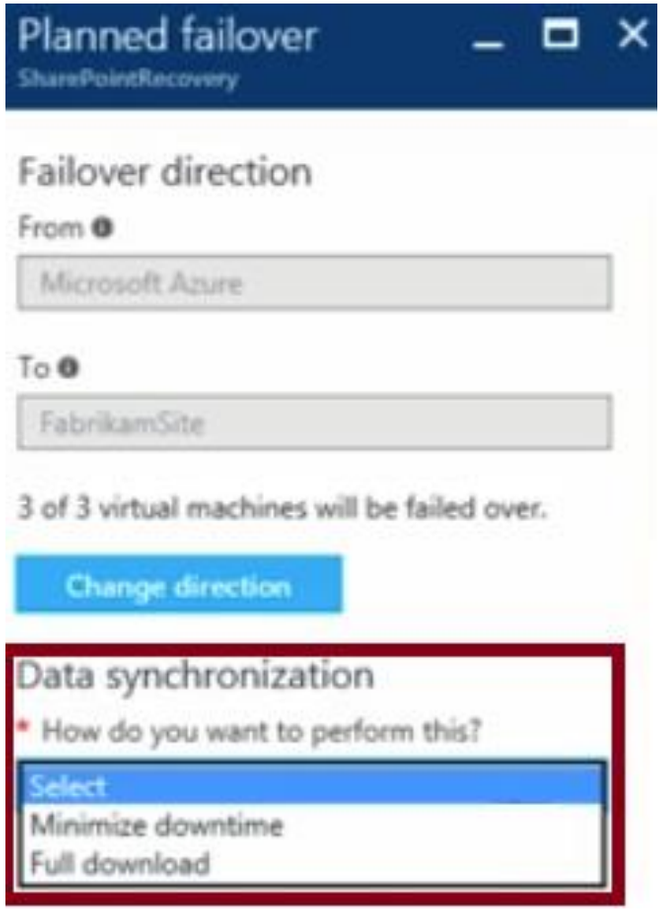
Test Failover

- **No-impact validation of replication and disaster recovery strategy:**
  - **Does not affect ongoing replication and the production environment**
  - **Supports single or multiple VMs**
  - **Does not require downtime**
  - **Does not introduce data loss**
- **Best practices:**
  - **Use failover groups**
  - **Failover to an isolated network**
  - **Integrate with Azure Automation**



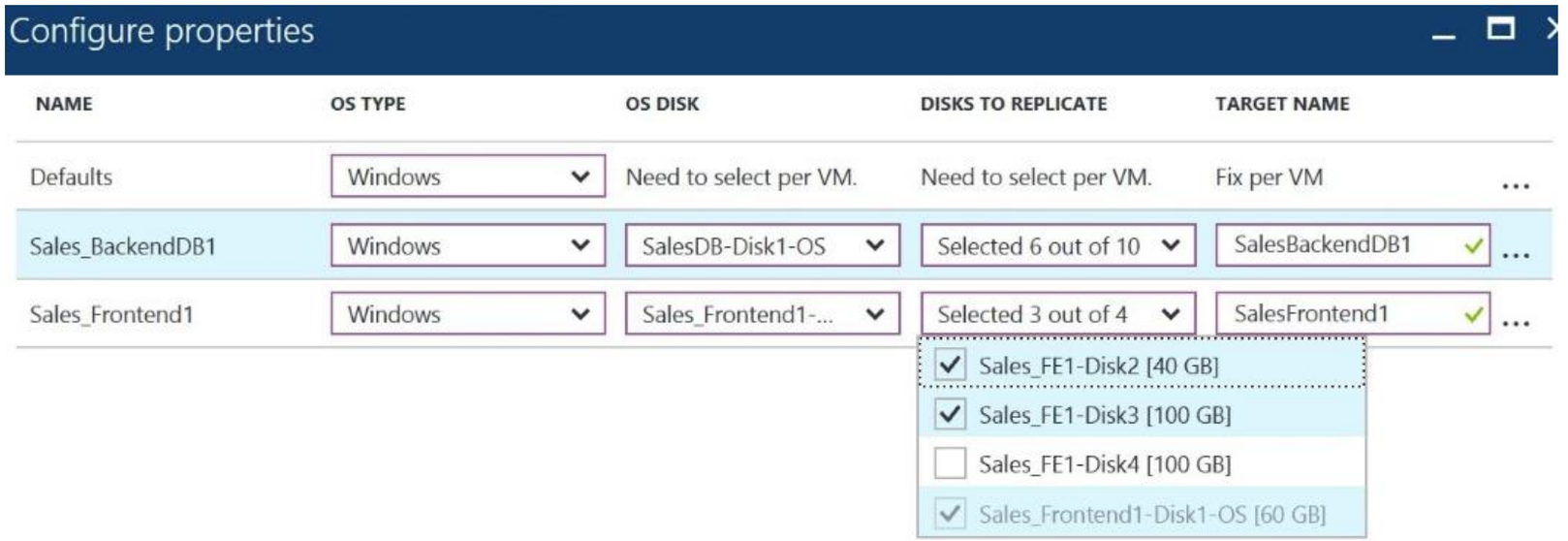
Failback

- **A planned failover back to the primary site**
- **Offers two data synchronization methods:**
  - **Full download:**
    - **Faster**
    - **Requires VMs to be shut down**
  - **Minimize downtime:**
    - **Slower**
    - **Eliminates the need to shut down VMs**



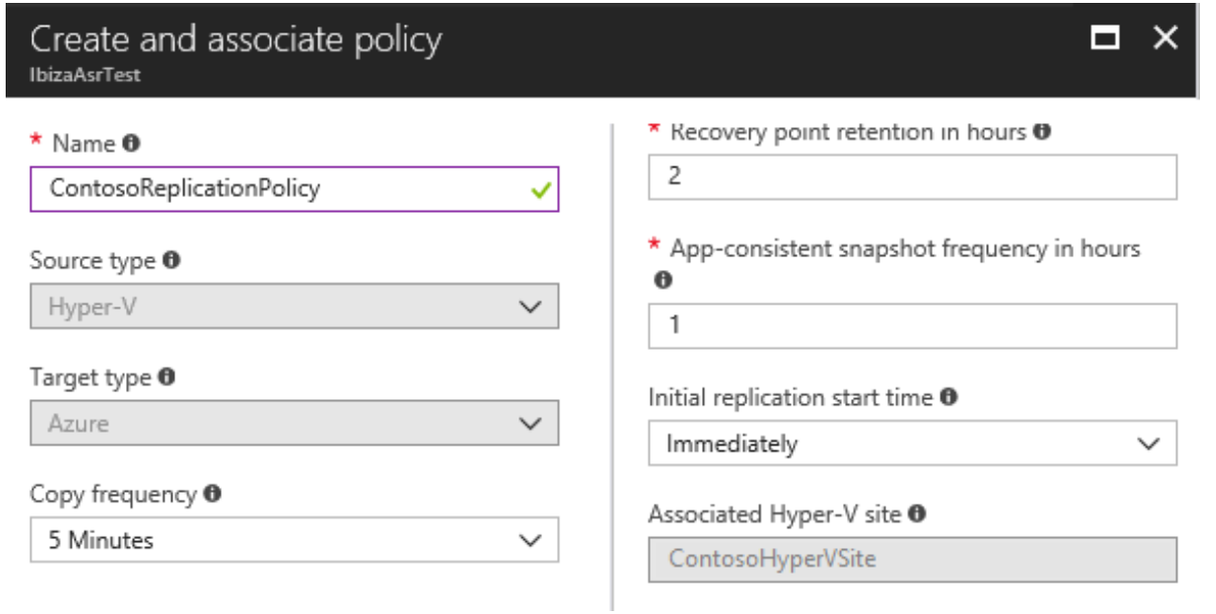
Virtual Machine Replication

- **To configure:**
  - **Select source computers**
  - **For each, select the OS type, OS disk, and data disks to replicate**
  - **Optionally, exclude disks (e.g. disks hosting a paging file)**
  - **For each, specify the name of a target VM**



Replication Policy

- To configure, specify the following parameters:
  - **Copy frequency:**
    - **30 seconds**
    - **5 minutes**
    - **15 minutes (to be retired)**
  - **Recovery point retention**
  - **App-consistent snapshot**
  - **Initial replication start time**

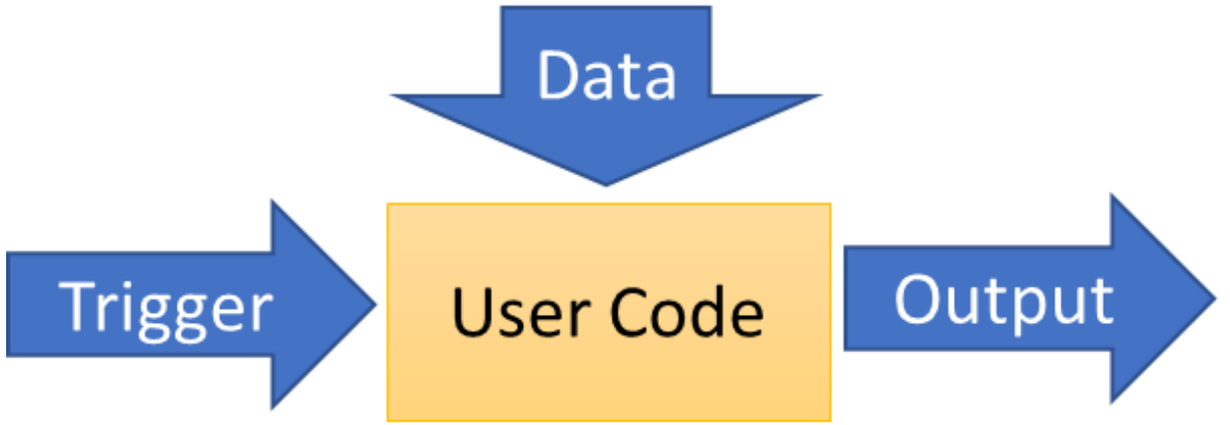


Microsoft Azure

Configure serverless computing

Serverless Computing

- **Abstracts servers, infrastructure, and operating systems**
- **Reacts to events and triggers in near-real time**
- **Offers a range of benefits:**
  - **Eliminates management overhead**
  - **Allows developers to focus on business logic**
  - **Implements flexible scaling**
  - **Reduces costs**



Infrastructure spun-up, scaled, and spun-down when no longer needed

Serverless Applications

- **Compute:**
  - **Azure Functions**
- **Cloud Messaging:**
  - **Event Grid**
  - **Service Bus**
- **Workflow Orchestration:**
  - **Logic Apps**

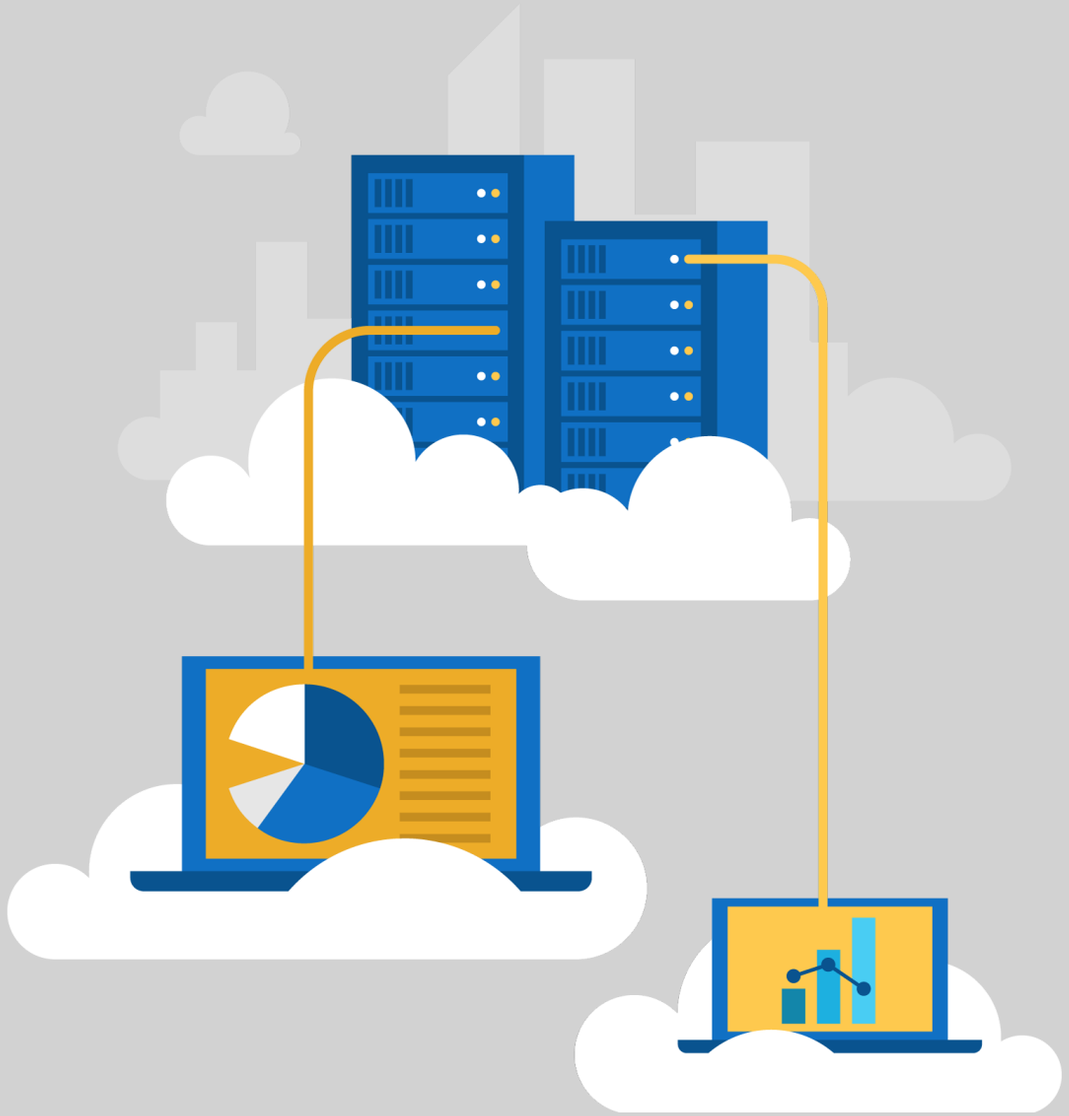
Comparing Serverless Options

	Durable Functions	Logic Apps
Development	Code-first (imperative)	Designer-first (declarative)
Connectivity	About a dozen built-in binding types, write code for custom bindings	Large collection of connectors, Enterprise Integration Pack for B2B scenarios, build custom connectors
Actions	Each activity is an Azure function; write code for activity functions	Large collection of ready-made actions
Monitoring	Azure Application Insights	Azure portal, Operations Management Suite, Log Analytics
Management	REST API, Visual Studio	Azure portal, REST API, PowerShell, Visual Studio
Execution context	Can run locally or in the cloud.	Runs only in the cloud.

	Functions	WebJobs with WebJobs SDK
Serverless app model with automatic scaling	Yes	No
Develop and test in browser	Yes	No
Pay-per-use pricing	Yes	No
Integration with Logic Apps	Yes	No

Microsoft

Managing Logic App

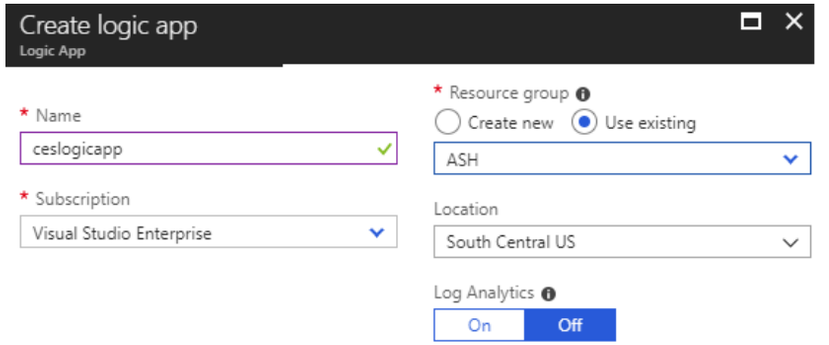


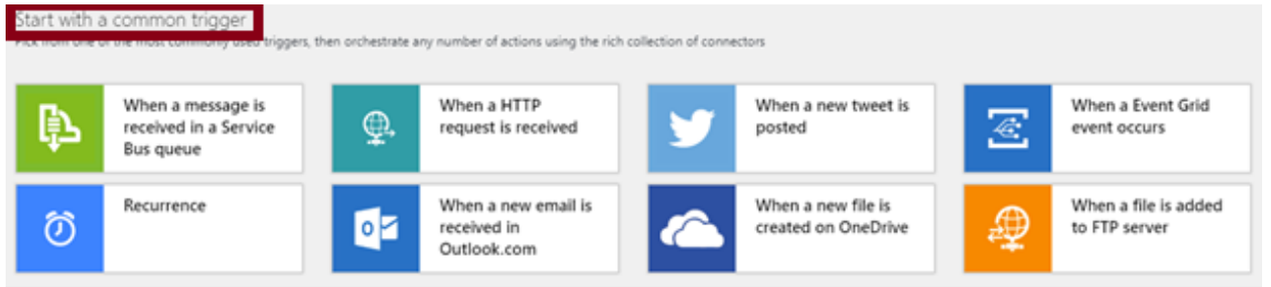
Logic Apps

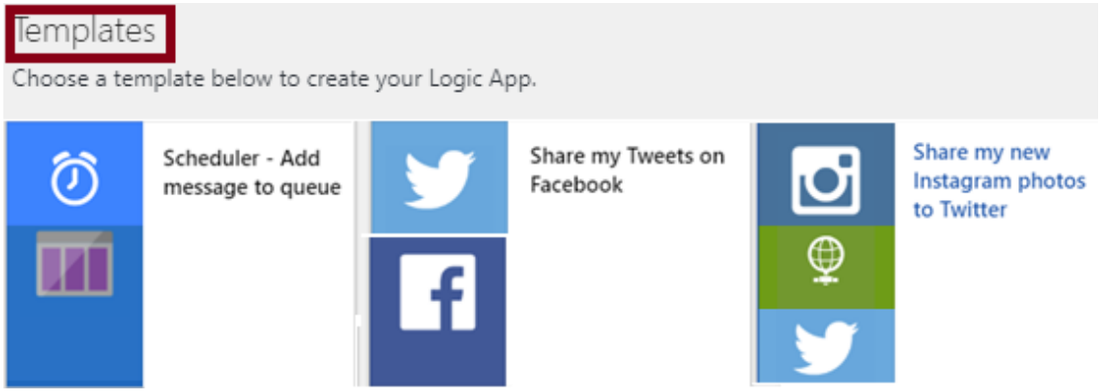
- **Implement scalable integrations and workflows**
- **Include a visual designer to model workflows**
- **Offer built-in connectors for cloud and on-premises services and apps**
- **Provide a range of benefits, including:**
  - **Getting started quickly from templates.**
  - **Saving time by designing complex processes using easy to understand design tools.**
  - **Implementing patterns and workflows seamlessly, that would otherwise be difficult to implement in code.**
  - **Customizing your logic app with your own custom APIs, code, and actions.**
  - **Connecting and synchronizing disparate systems across on-premises and the cloud.**

Implementing Logic Apps

- **From the Azure portal:**
  - **To create a logic app, provide:**
    - **Name**
    - **Subscription**
    - **Resource Group**
    - **Location**
  - **To configure a logic app:**
    - **Use the Logic Apps Designer to:**
      - **Add triggers**
      - **Add connectors**
      - **Leverage templates**

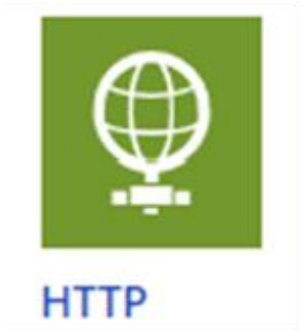


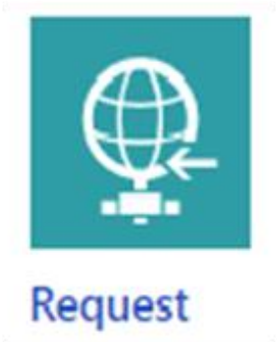


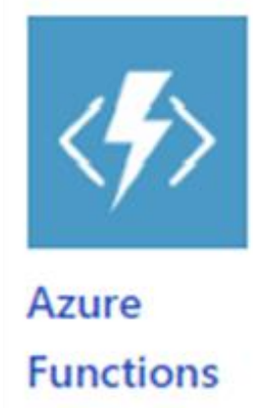


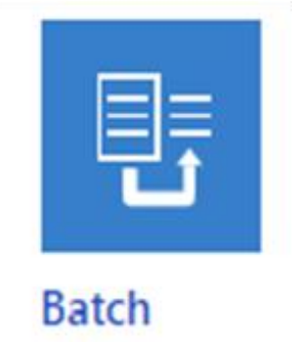
Built-in Triggers and Actions

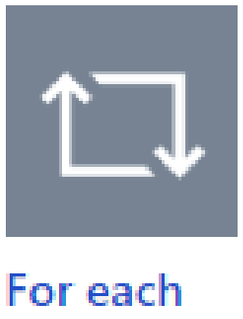
- **Triggers:**
  - **Recurrence**
  - **HTTP**
  - **Request**
  - **Azure Functions**
  - **Batch**
  - **Logic Apps**
- **Actions:**
  - **Condition**
  - **For each**
  - **Scope**
  - **Switch**
  - **Terminate**
  - **Until**

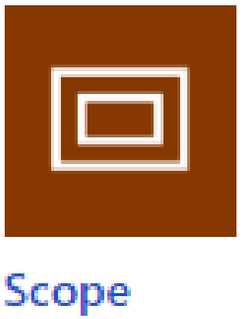


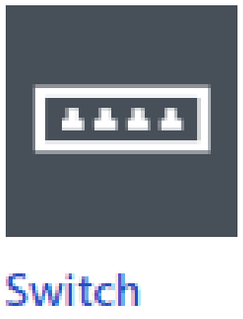


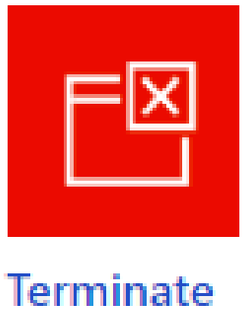


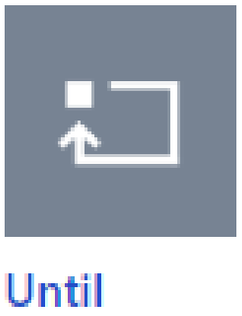










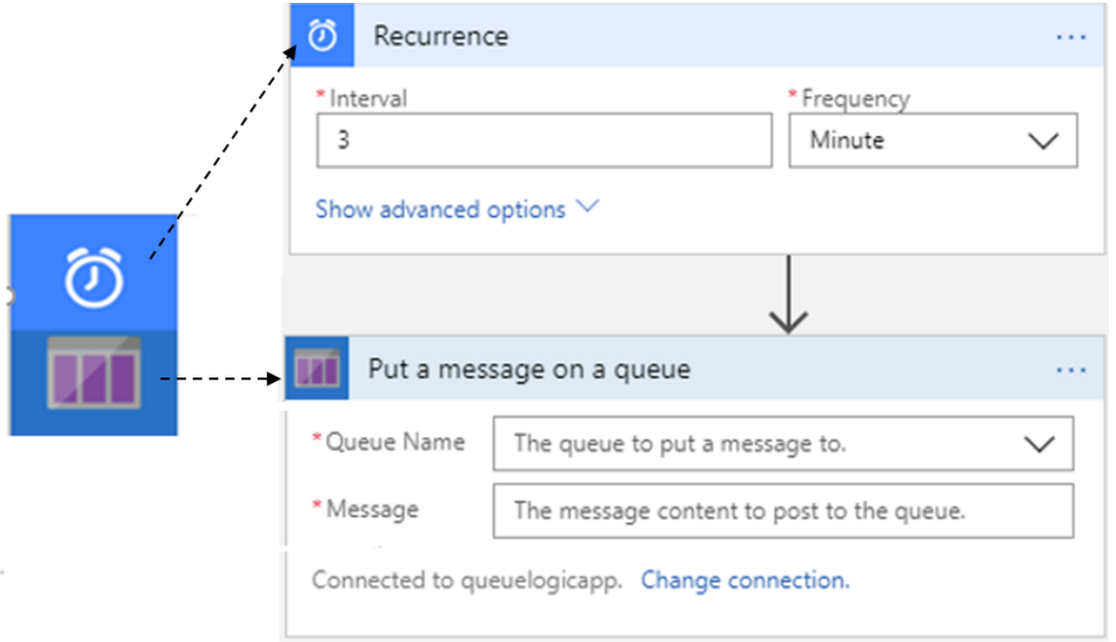


Managed Connectors

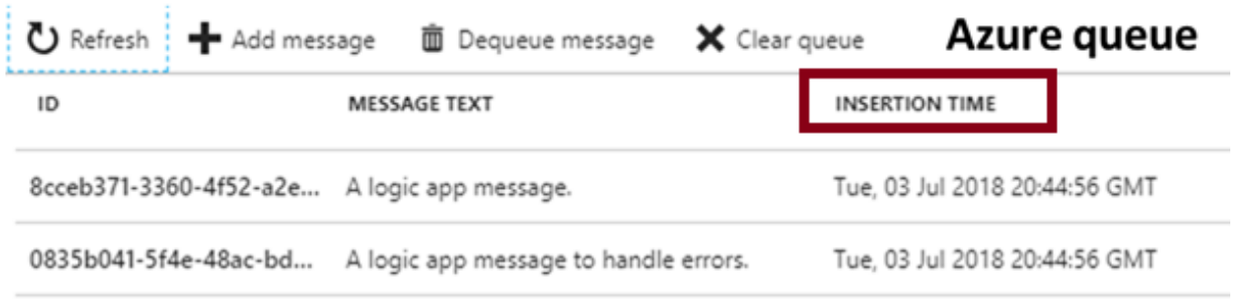
- **200+ built-in connectors, including:**
  - **Managed API connectors:**
    - **Azure Blob Storage, Office 365, Dynamics, Power BI, OneDrive, Salesforce, and SharePoint Online.**
  - **On-premises connectors:**
    - **SQL Server, SharePoint Server, Oracle DB, Twitter, Salesforce, Facebook, and file shares.**
  - **Integration account connectors:**
    - **Require a paid-for integration account**
    - **Transform and validate XML**
    - **Encode and decode flat files**
    - **Process B2B messages**
  - **Enterprise connectors:**
    - **Incur extra cost**
    - **Provide access to enterprise systems:**
      - **Support SAP and IBM MQ.**

Logic App Example

- **The Scheduler logic app consists of:**
  - **Recurrence:**
    - **Controls scheduling**
  - **Put a message on a queue:**
    - **Designates the target Storage queue**

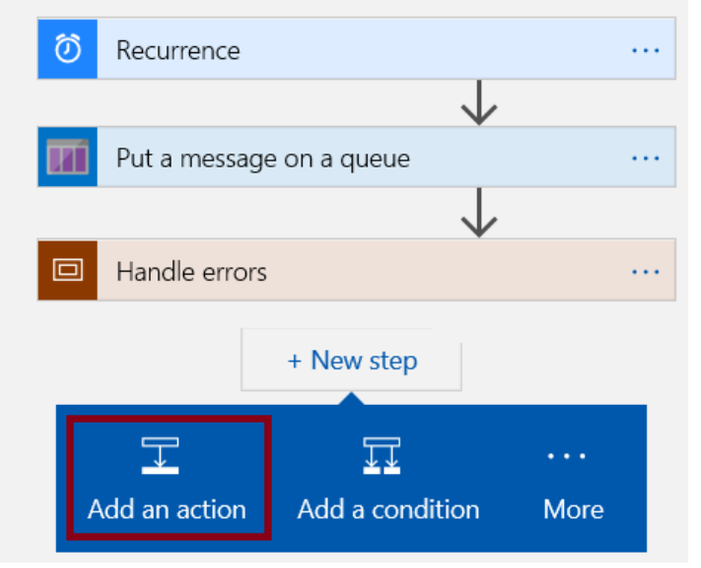


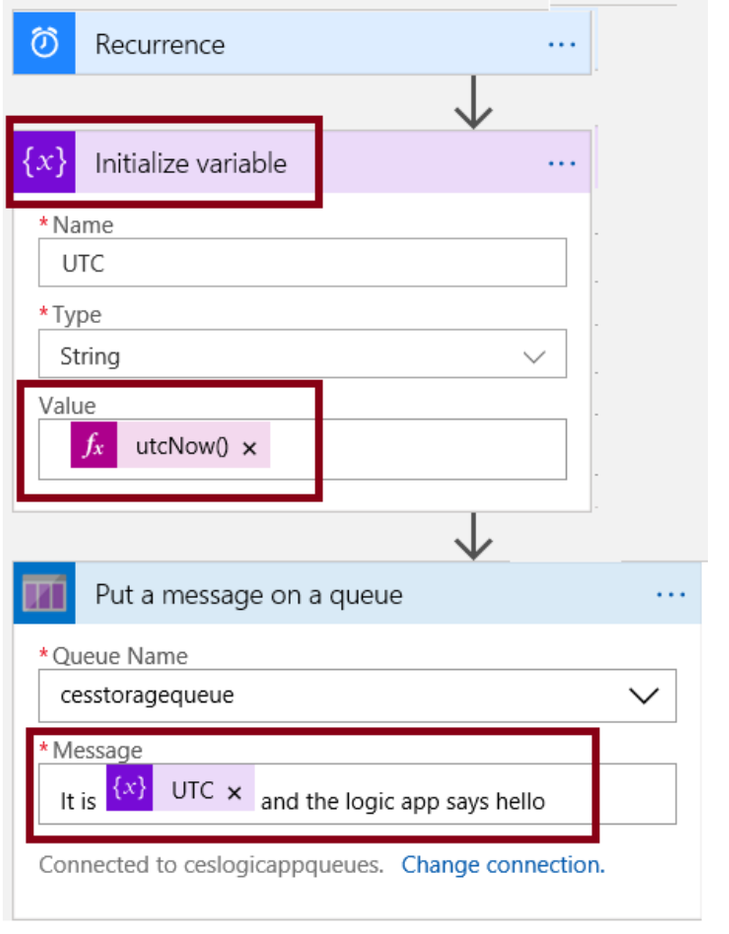
Scheduler - Add message to queue



Logic App Example (cont.)

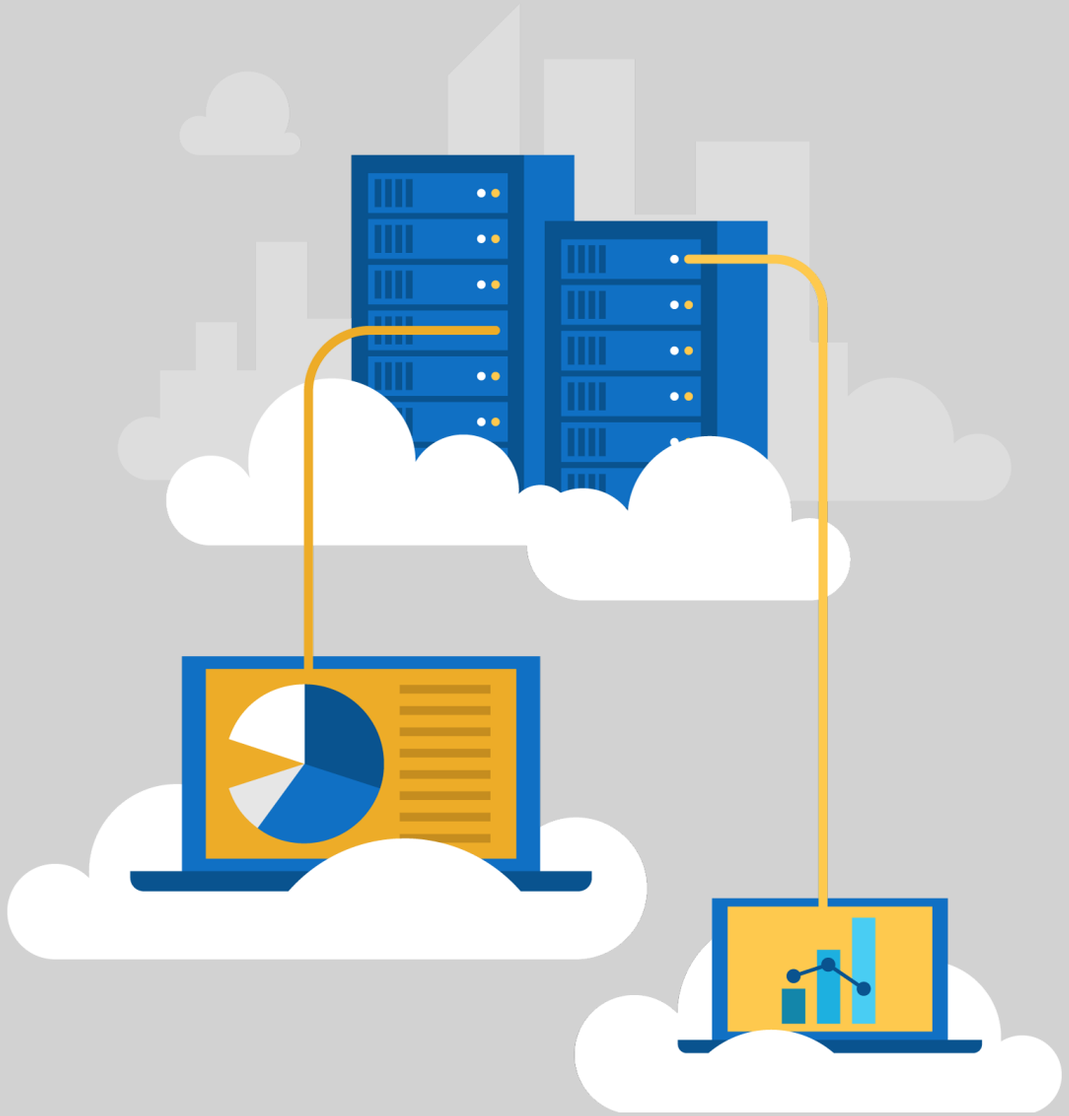
- **To customize the workflow:**
  - **Add a extra action:**
    - **Variable (Initialize Variable)**
  - **Modify the action sequence**
  - **Add variable to the message**





Microsoft

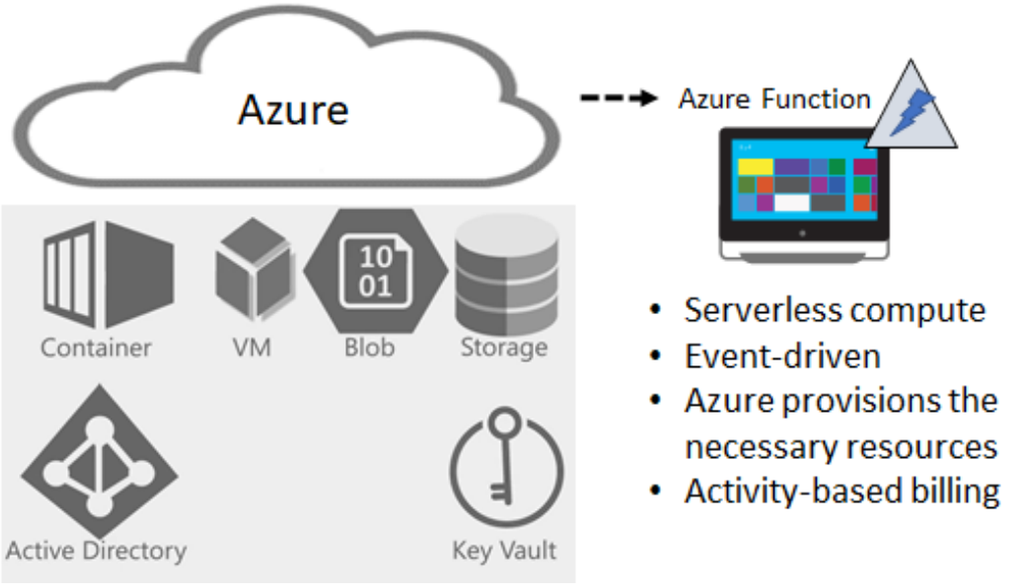
Managing Azure Functions



Overview of Azure Functions

Serverless compute service:

Eliminates the need to explicitly provision or manage infrastructure
Enables running code on-demand or in response to a trigger
Incurs charges only when active

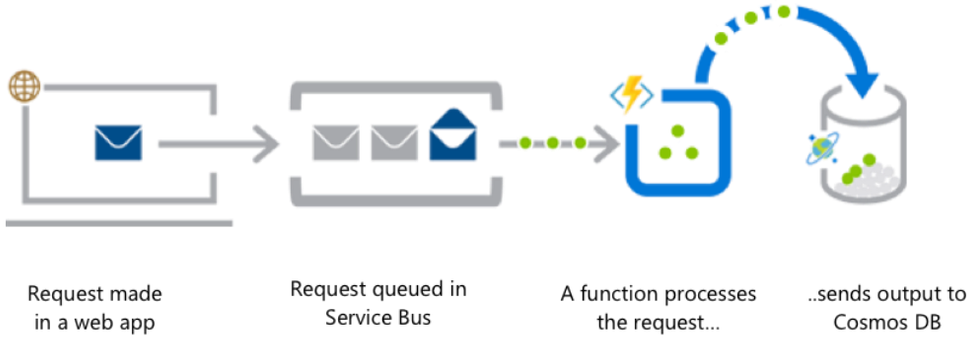


Features of Azure Functions

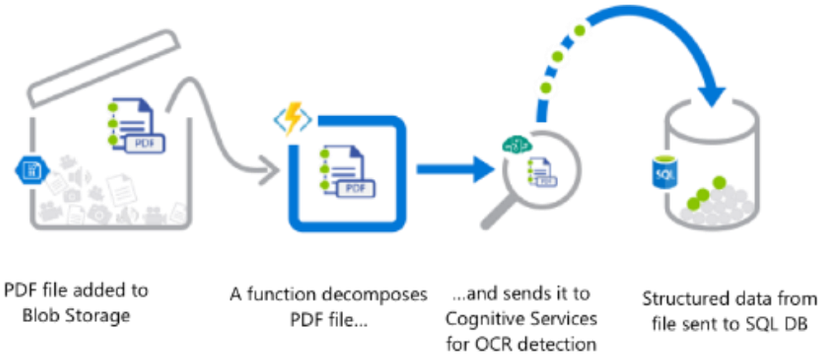
Support for a range of programming languages:
C#, F#, Node.js, Python, PHP, batch, bash, or any executable.
Pay-per-use pricing model
Support for custom dependencies:
NuGet and NPM-based libraries.
Integration with the most popular OAuth providers:
Azure AD, Facebook, Google, Twitter, and Microsoft Account.
Integration with other Azure services and SaaS apps.
Flexible development:
Directly from the Azure portal
Via continuous integration through GitHub, VSTS, and other supported development tools.
Open-source: available on GitHub.
Ease of code reuse:
Developers can reuse their functions in multiple applications.

Azure Functions (Examples)

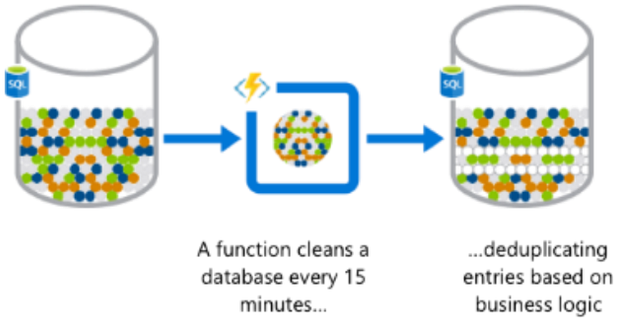
Web app backend



Real-time file processing



Automation of scheduled tasks



Function Service Plans

Hosting plans:

Consumption Plan:

Azure provides all the necessary compute resources on-demand

Automatic scaling of CPU and memory

Billing is based on:

number of executions

execution time

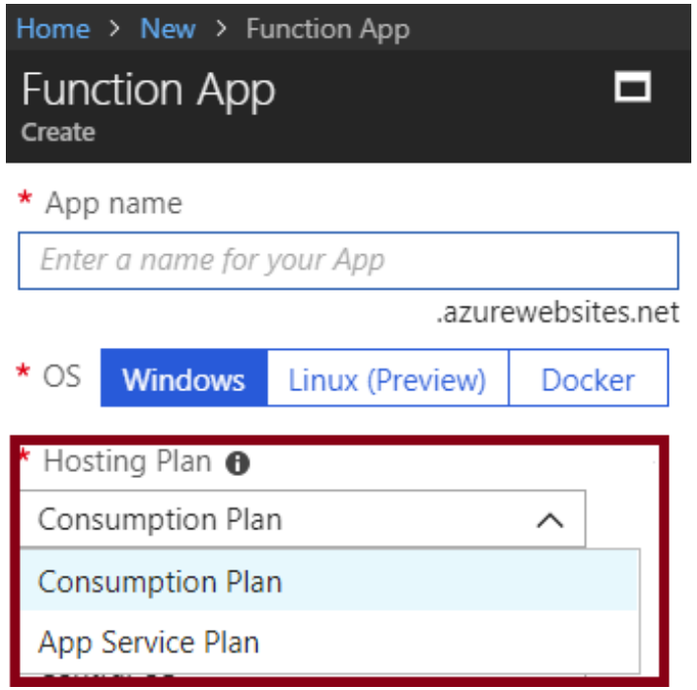
memory used.

App Service Plan:

The same resource and billing model as for web, mobile, and API apps

Allows for leveraging underutilized App Service deployments

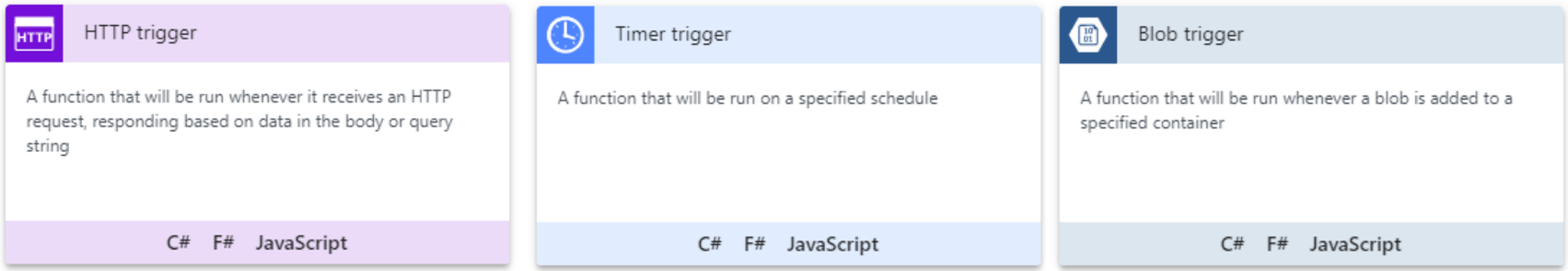
Required for Linux-based function apps



Function Templates

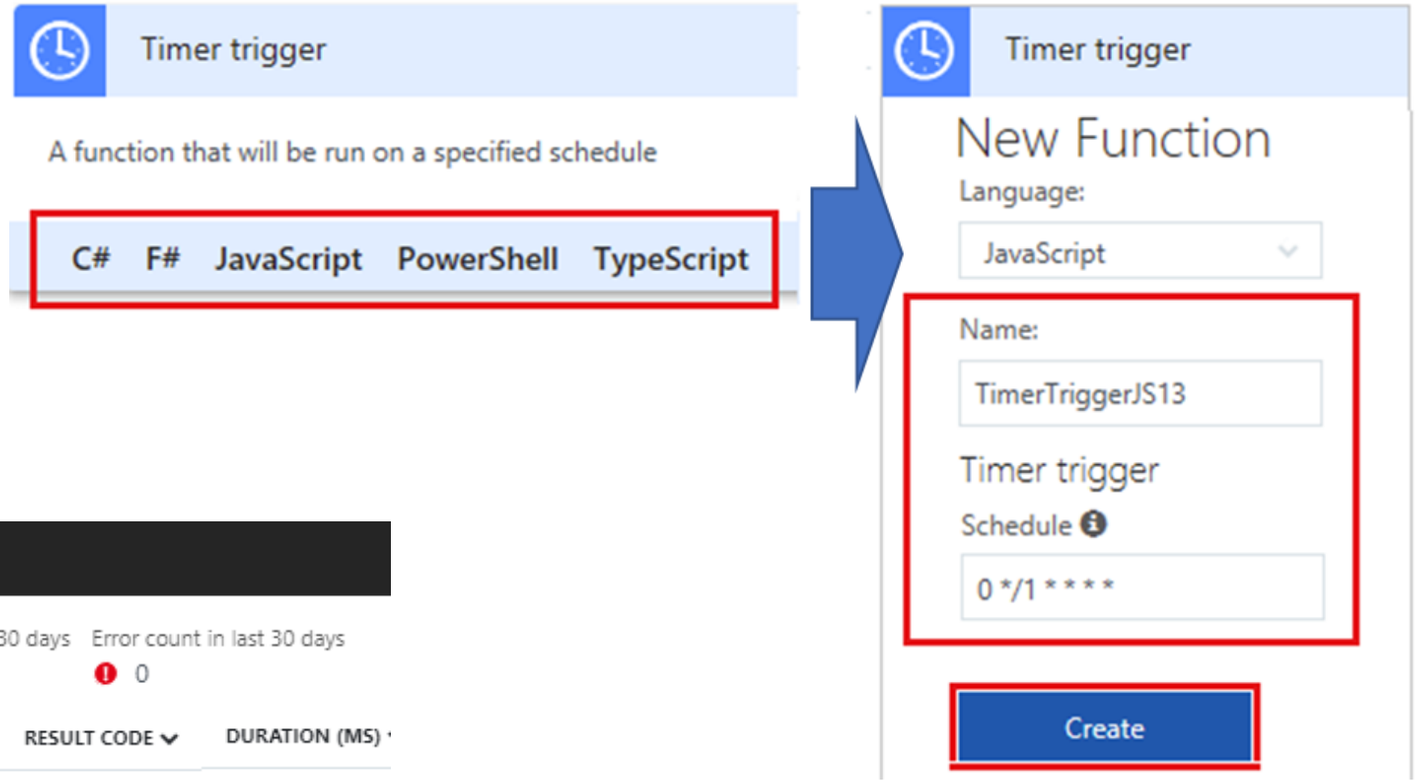
Simplify creating functions for a specific trigger type:

HTTP
Timer
Blob
Event Hub
GitHub
Webhook
Queue

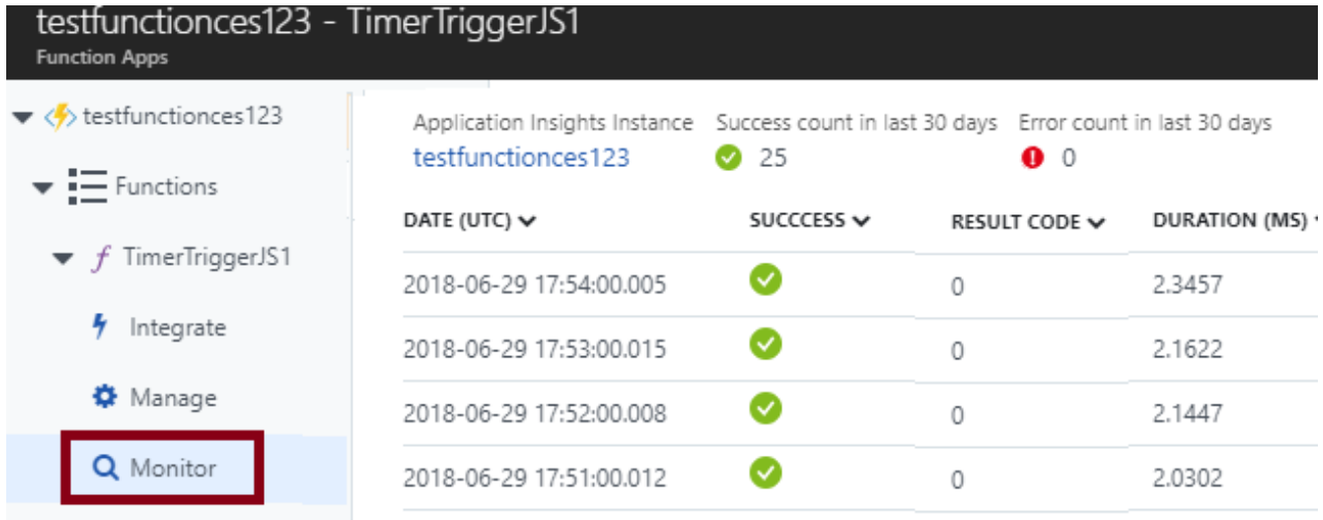


Implementing Functions

In-portal authoring



In-portal monitoring

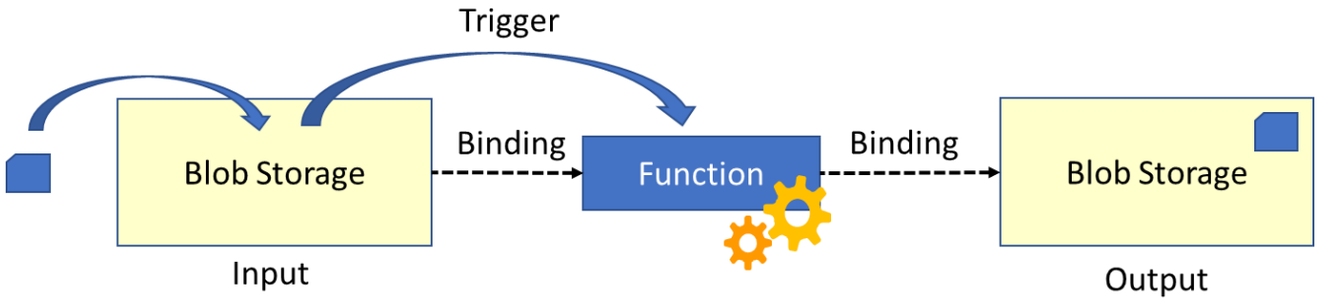


Bindings

Link functions to other services as:

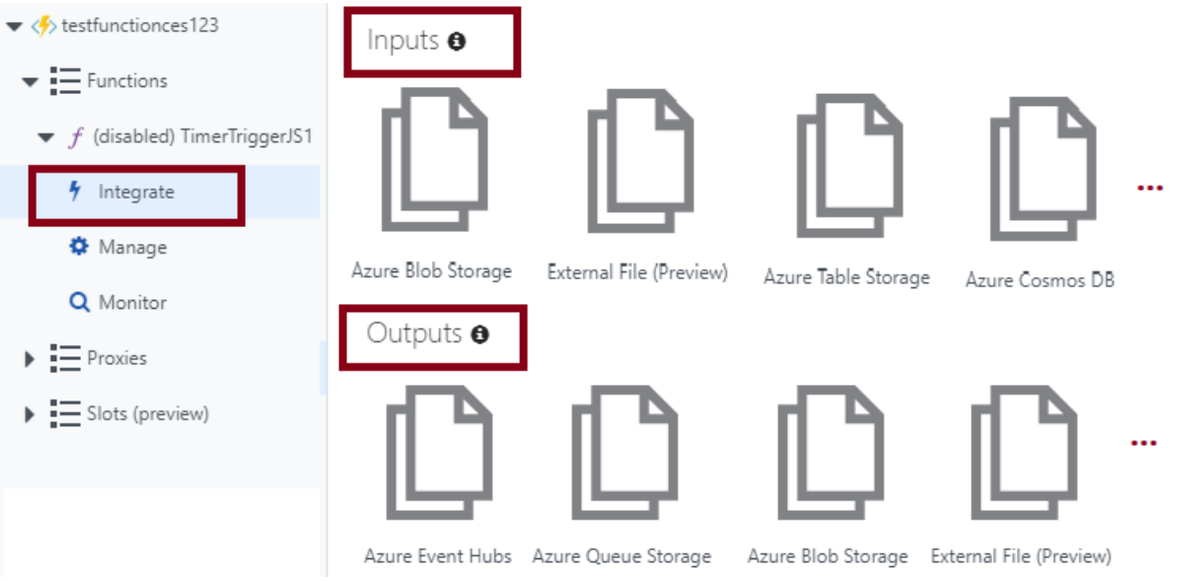
Inputs

Outputs



Configurable in the Azure portal:

Accessible via the Integrate link



Function Scaling

Relies on the platform-managed Scale Controller

Scale Controller:

Monitors the rate of events

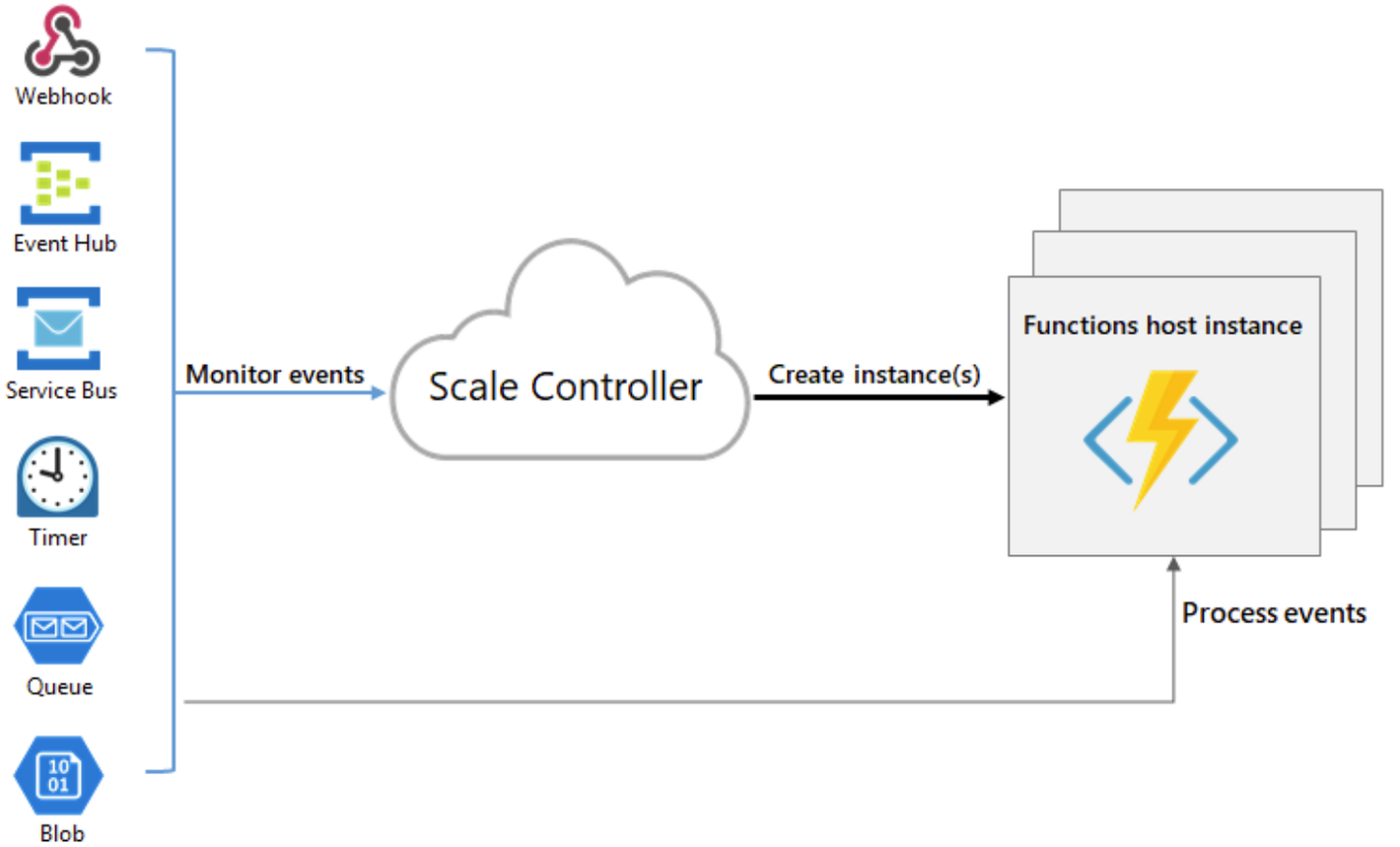
Controls horizontal scaling

Relies on heuristics:

E.g. for Queue Storage trigger:

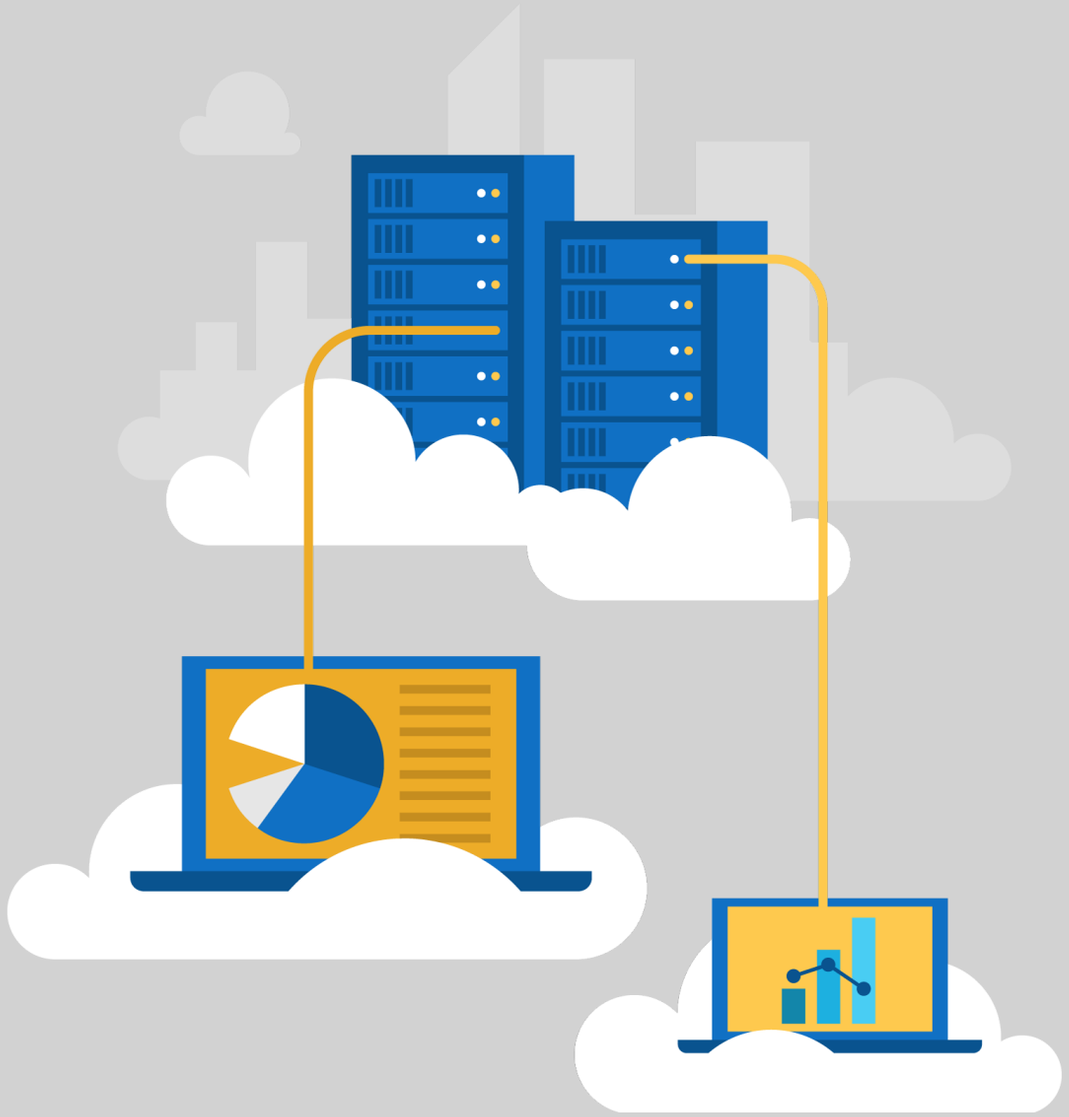
The queue length

The age of the oldest message



Microsoft

Manage Event Grid



Overview of Event Grid

Managed event routing service with a wide range of benefits:

Simplifies event delivery:

Connects multiple event sources and destinations, including virtually all Azure services and custom sources

Provides fully managed event delivery, intelligent filtering, and the ability to send events to multiple recipients at once

Eliminates polling and the associated cost and latency

Decouples event publishers from subscribers by using a pub/sub model and HTTP-based event delivery,

Simplifies building scalable serverless applications, microservices, and distributed systems.

Facilitates developing reliable cloud applications:

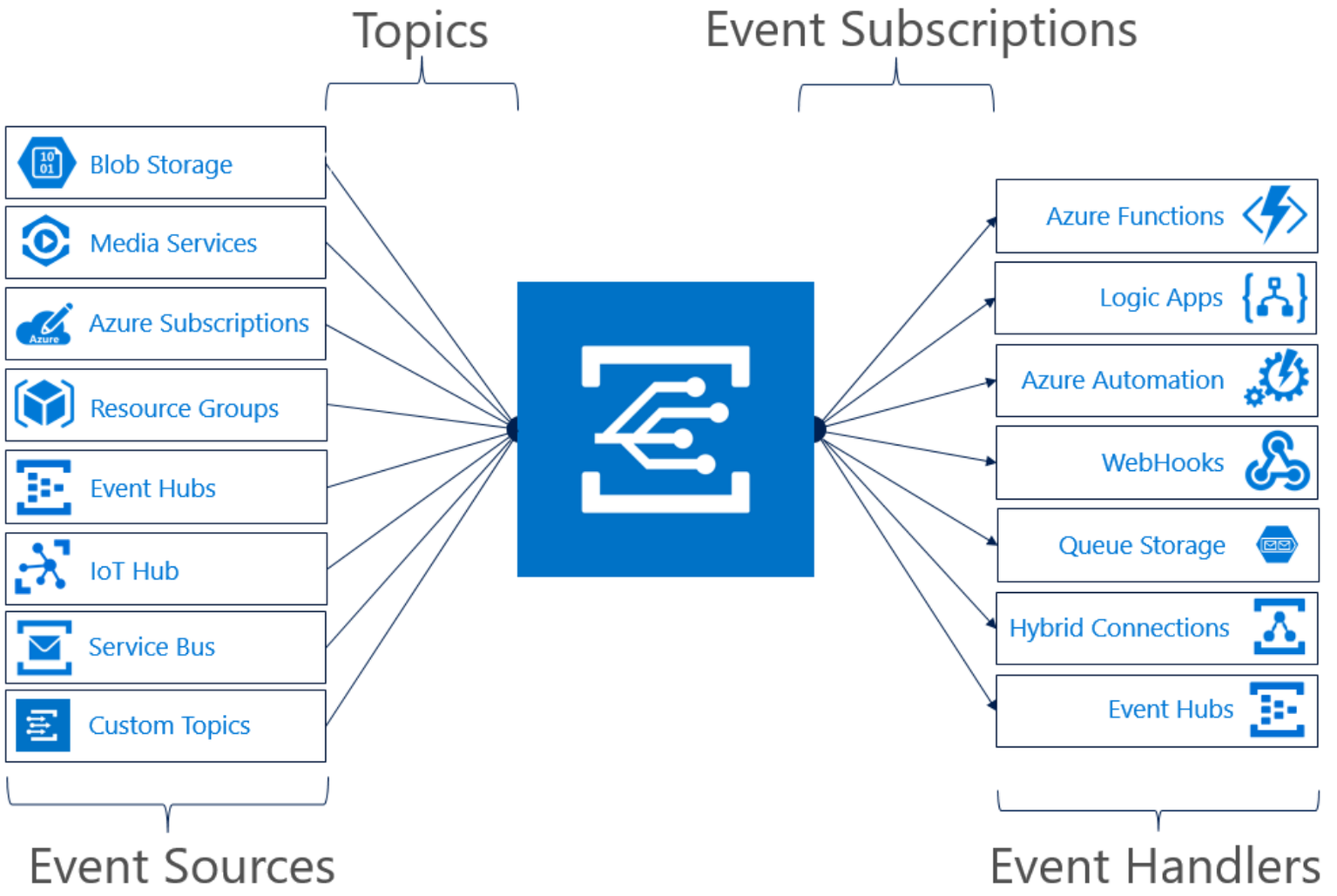
Supports massive auto-scaling with near-real-time notifications for changes

Implements reactive programming leveraging guaranteed event delivery

Promotes focus on product innovation

Event Grid Concepts

Event Sources
Topics
Event Subscriptions
Event Handlers

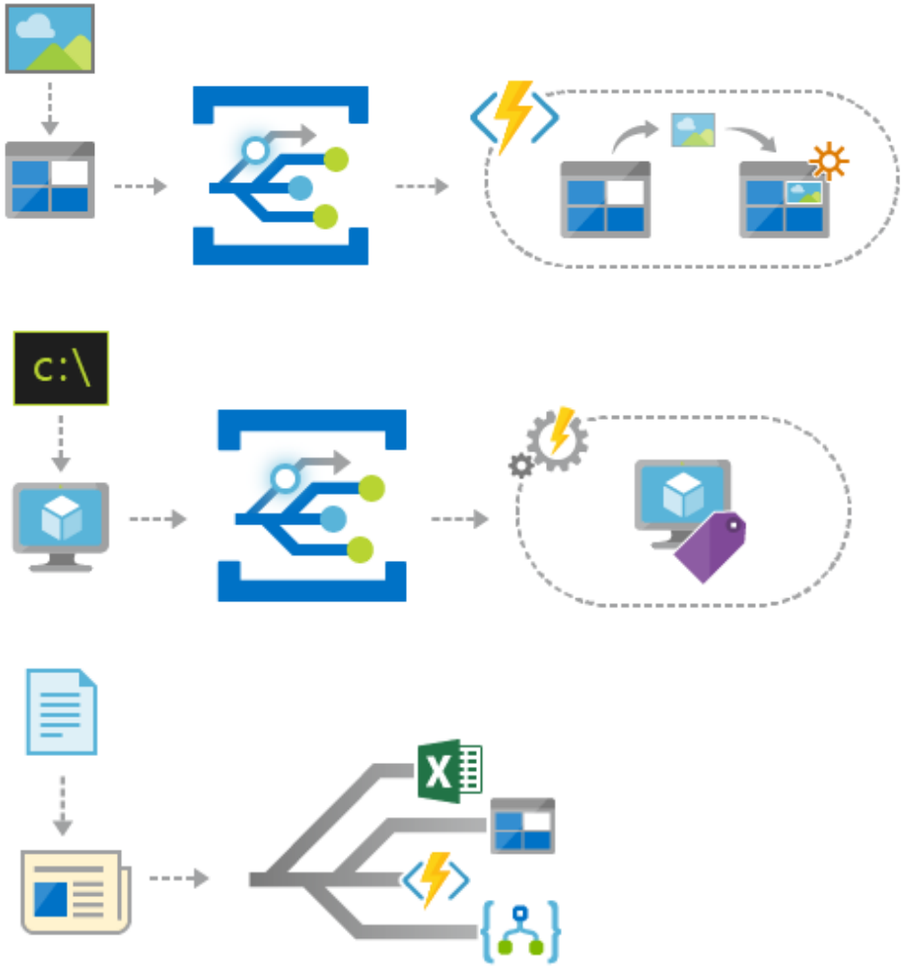


Event Grid Examples

Serverless application architecture

Ops automation

Application integration



Implementing Event Grid (Part 1)

Implementation components:

Event source: Azure Blob Storage

Event Handler: Azure Queue Storage

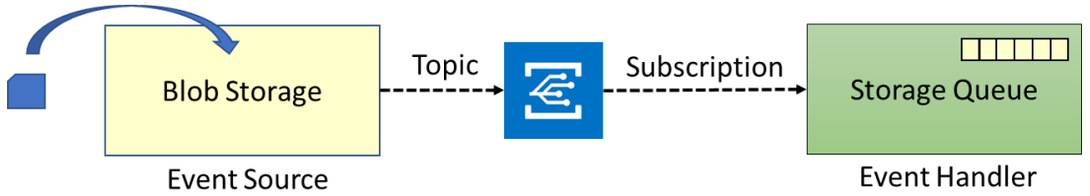
Implementation steps (part 1):

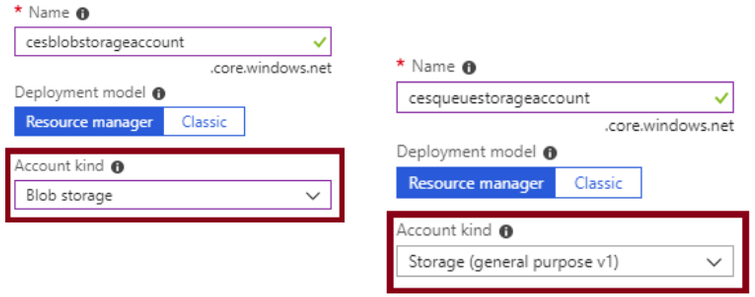
1. **Create an Azure Storage account (Blob storage)**
2. **Create an Azure Storage account (general purpose v1))**
3. **Create an Azure Storage queue**
4. **Create the event source:**

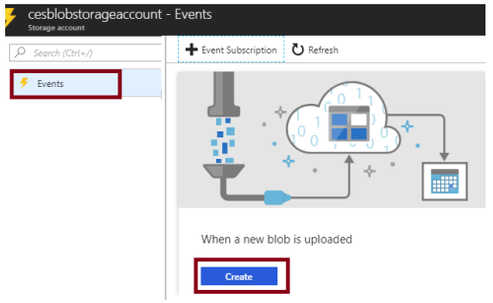
Use the When a new blob is uploaded template

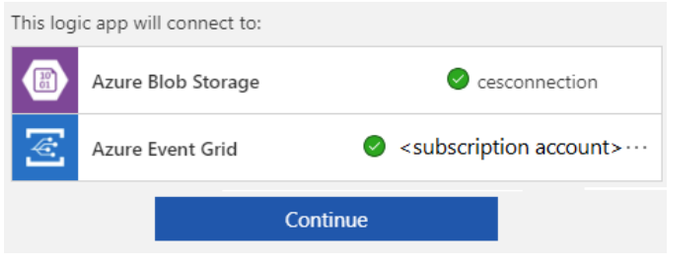
5. **Configure the event source:**

When prompted, provide the Blob storage account





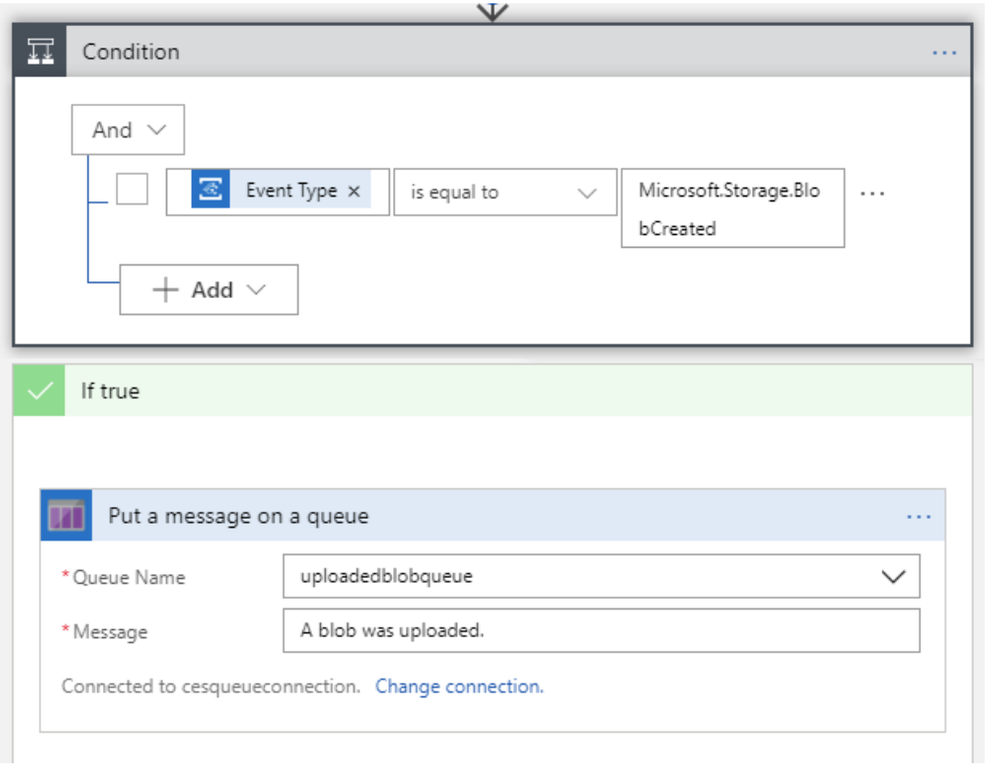




Implementing Event Grid (Part 2)

Implementation steps (part 2):

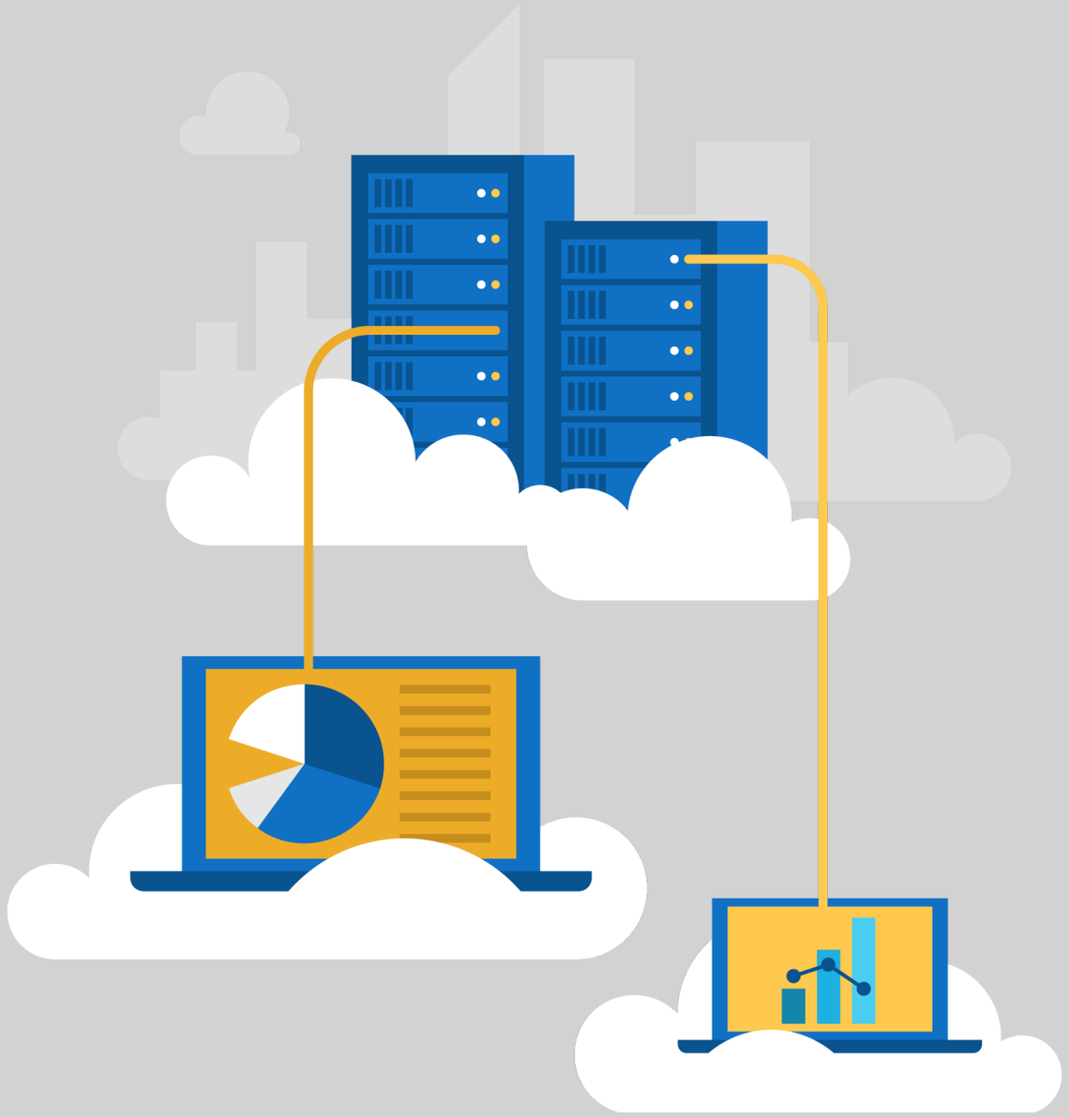
6. **Edit the Logic App:**
   - One condition: True whenever a blob is created
   - One action: Add a message to the queue.
7. **Save changes and start the Logic App**
8. **Create an storage container and upload a blob**
9. **View messages in the Storage queue**
10. **To avoid future charges, delete all resources**



ID	MESSAGE TEXT	INSERTION TIME	EXPIRATION TIME	DEQUEUE COUNT
5d9fe07a-efaa-47f1-a587...	A blob was uploaded.	Thu, 05 Jul 2018 23:55:36 GMT	Thu, 12 Jul 2018 23:55:36 GMT	0
8baedddd-44b2-4dc2-a1...	A blob was uploaded.	Thu, 05 Jul 2018 23:55:54 GMT	Thu, 12 Jul 2018 23:55:54 GMT	0

Microsoft

Managing Service Bus



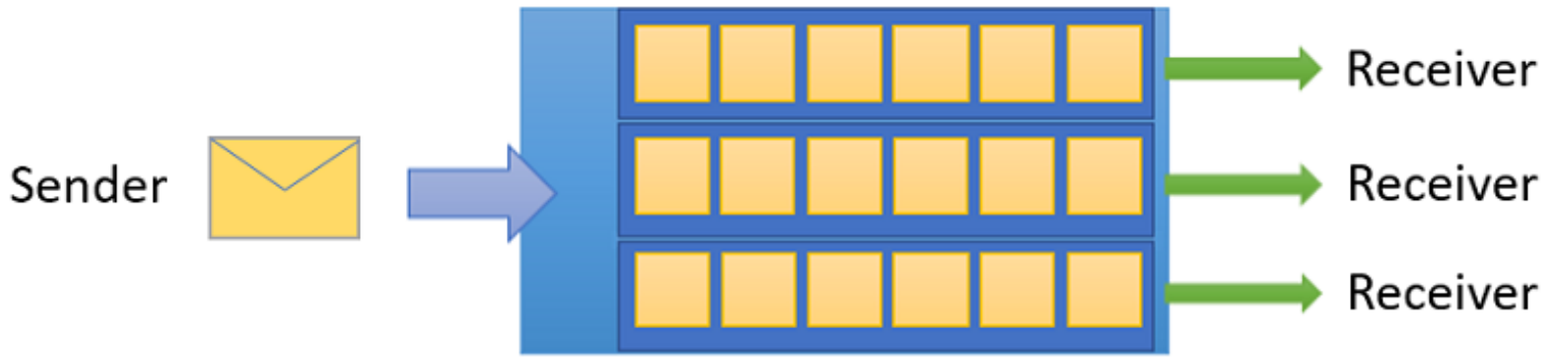
Queues

Azure Service Bus Queues provide:

Asynchronous, brokered messaging
Temporal decoupling of message senders and receivers
Structured message processing
Publish/subscribe capabilities
First In, First Out (FIFO) message delivery

Topics and Subscriptions

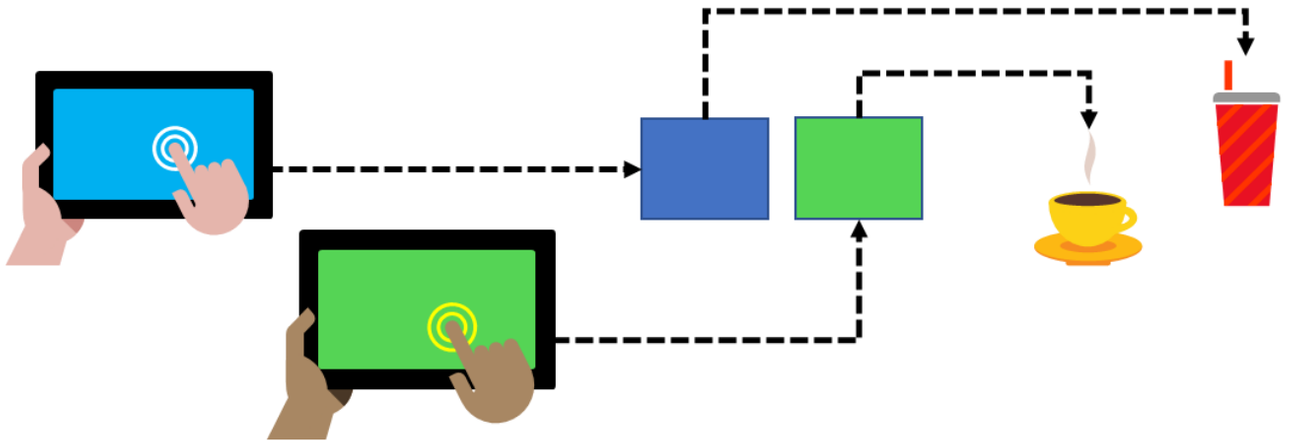
One-to-many communication in a publish/subscribe pattern
Useful for scaling large number of recipients
Each published message is available to topic-registered subscriptions
Subscriptions use filters to designate messages to receive



Topic with three Subscriptions
with Messages

Service Bus Features

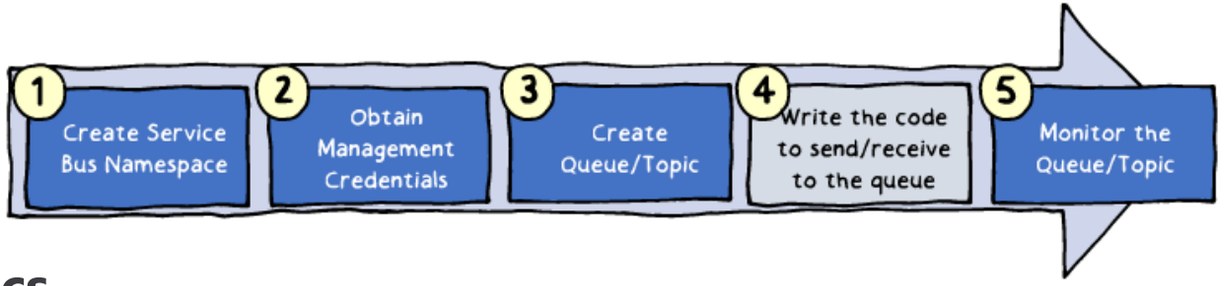
Load Leveling
Loose Coupling
Load Balancing

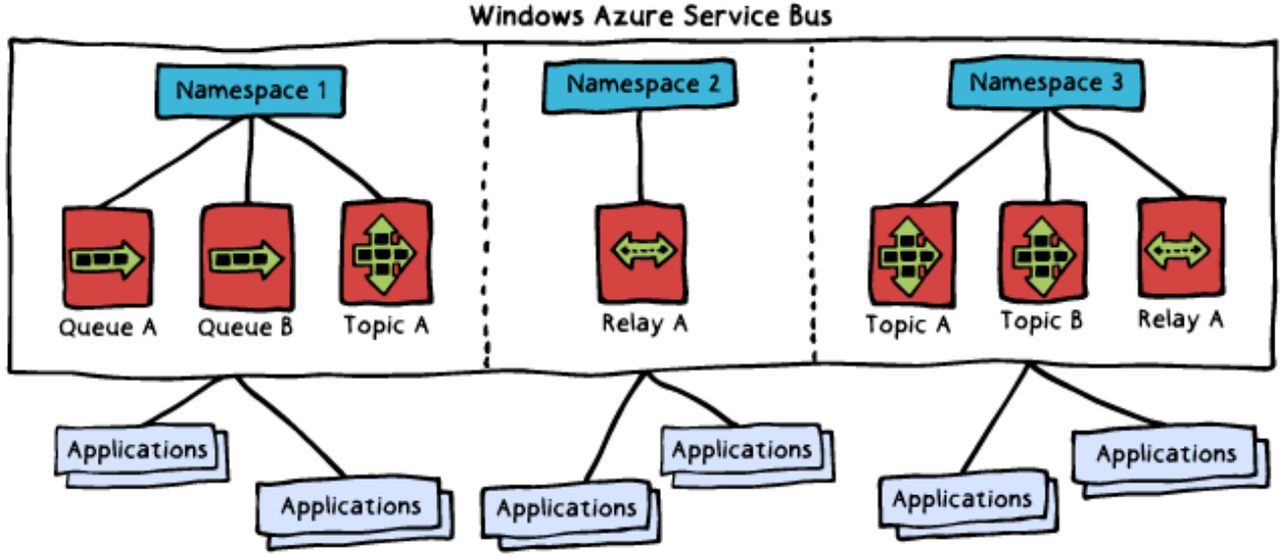


Implementing Service Bus

Implementation steps:

1. **Creating a Service Bus namespace**
2. **Obtaining management credentials**
3. **Creating queues and topics**
4. **Writing code to send/receive messages**
5. **Managing and monitoring queues and topics**





Creating the Namespace

Available directly from the Azure portal:

Name: globally identifies the namespace

Pricing tier: determines capacity and capabilities

Basic

Standard

Premium

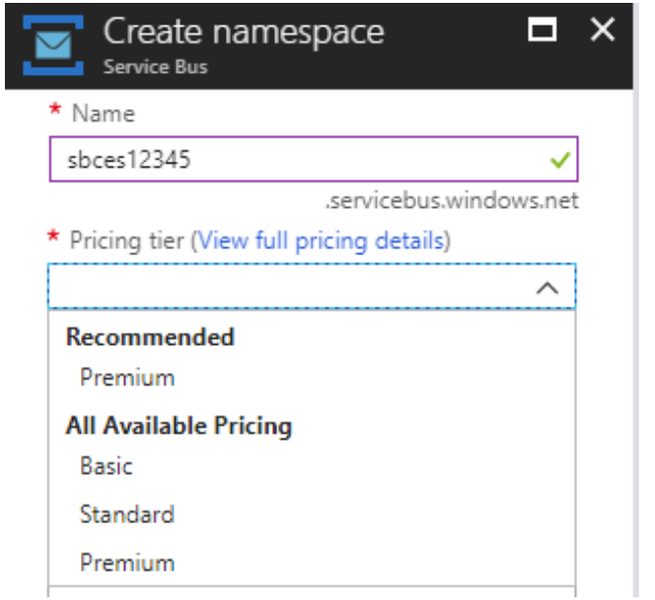
Generates a SAS policy:

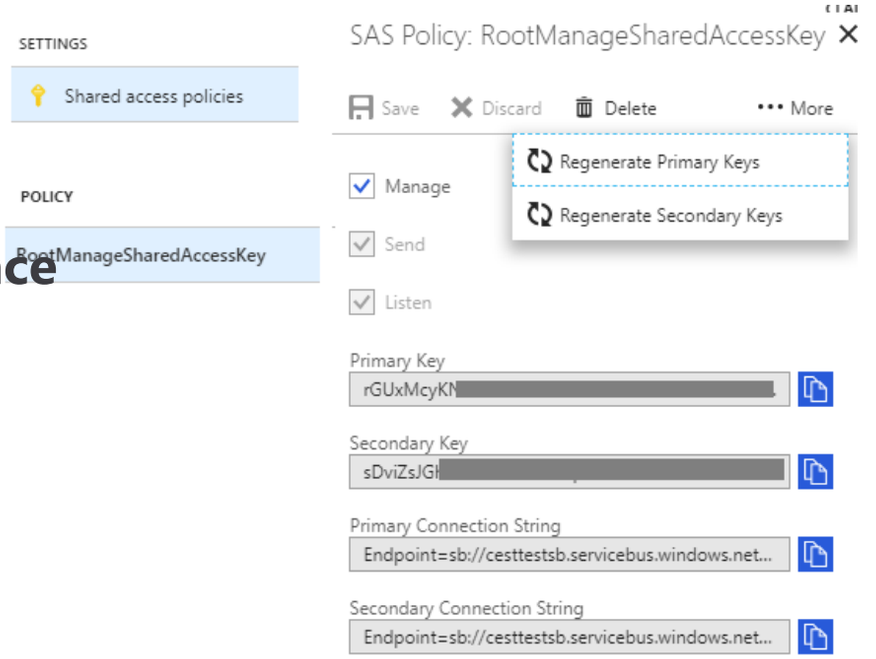
Includes:

Primary key

Secondary key

Grants full control to the namespace

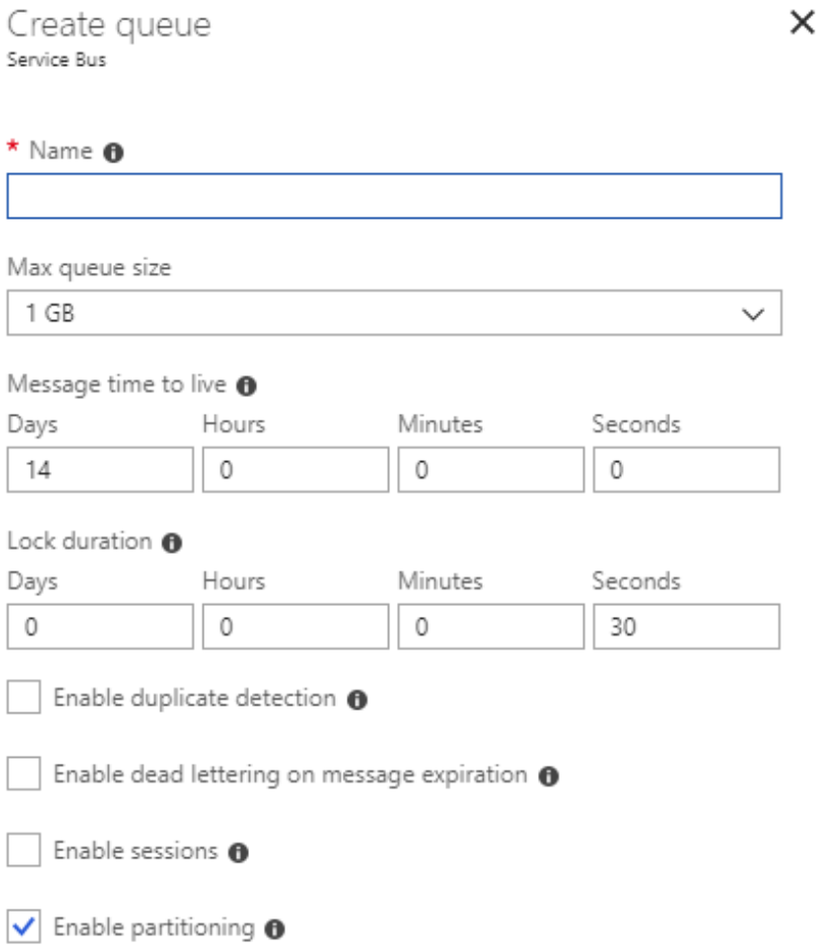




Create a Queue

Available directly from the Azure portal:

Name: identifies the queue within the namespace
Max queue size
Message time to live
Lock duration
Enable duplicate detection
Enable dead lettering
Enable sessions
Enable partitioning



Monitoring Service Bus

Available directly from the Azure portal:

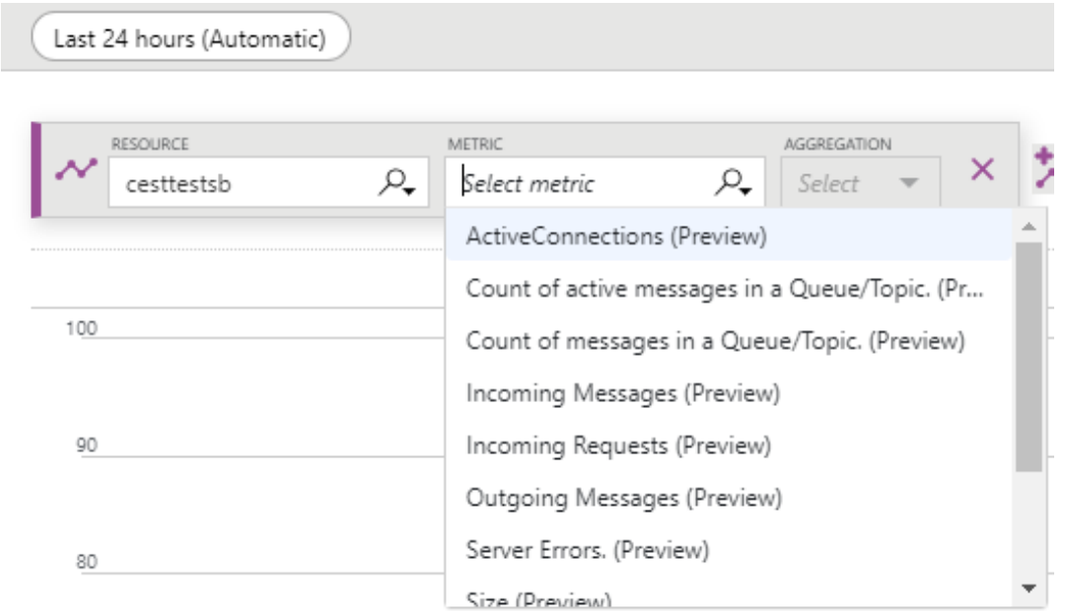
A rich set of metrics:

Namespace level

Queue/topic/message level

Diagnostic logs:

Capture all queue activities



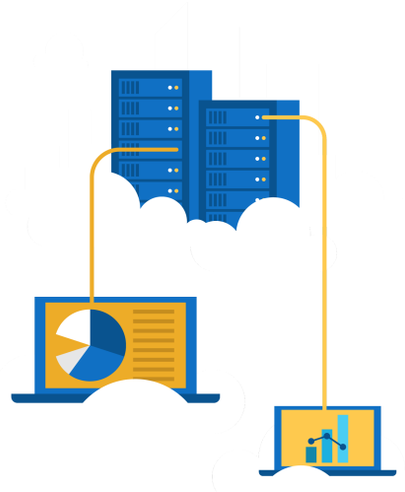
Comparing Service Bus and Storage Queues

Comparison Criteria	Storage Queues	Service Bus Queues
Ordering guarantee	No	Yes – FIFO
Delivery guarantee	At-Least-Once	At-Least-Once At-Most-Once
Lease/lock level	Message level	Queue level
Batch receive	Yes	Yes
Batch send	No	Yes
Scheduled delivery	Yes	Yes
Automatic dead lettering	No	Yes
Message auto-forwarding	No	Yes
Message groups	No	Yes
Duplicate detection	No	Yes

Microsoft Azure

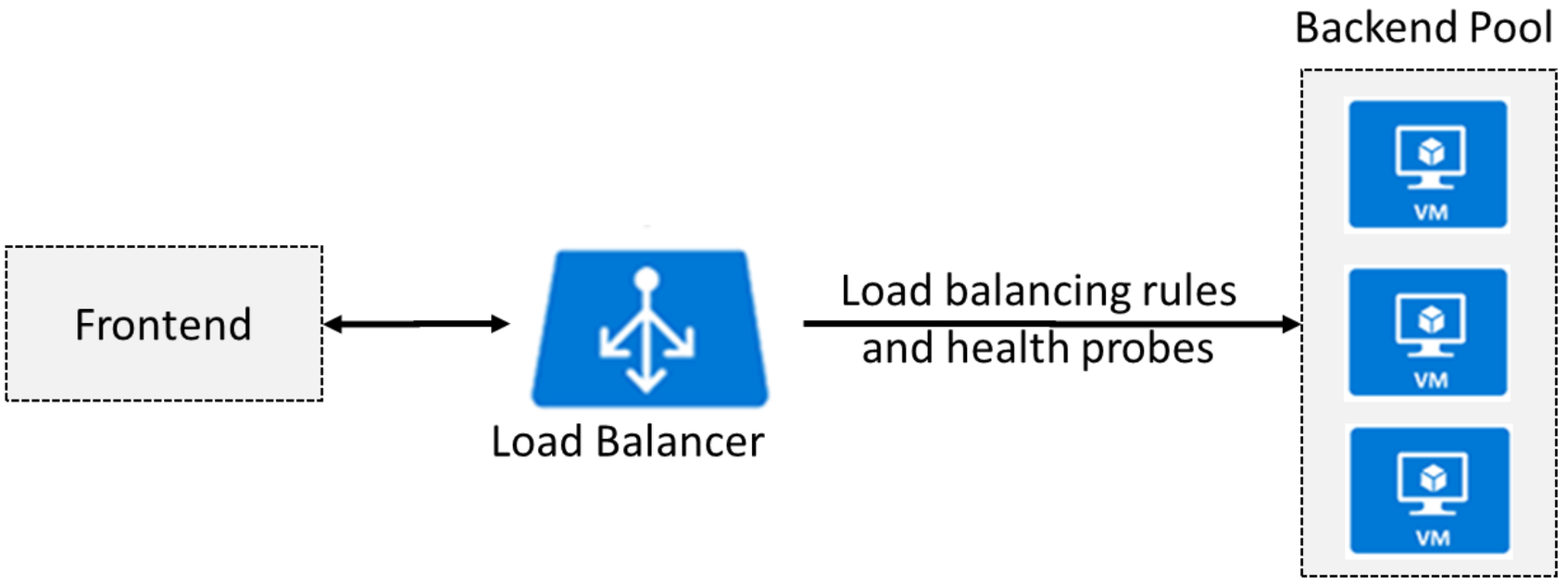
Implement application load balancing

Azure Load Balancer



Load Balancer

Operates on OSI Layer 4 (TCP/UDP)
Relies on health probes to determine status of backend pool
Distributes traffic according to load balancing rules

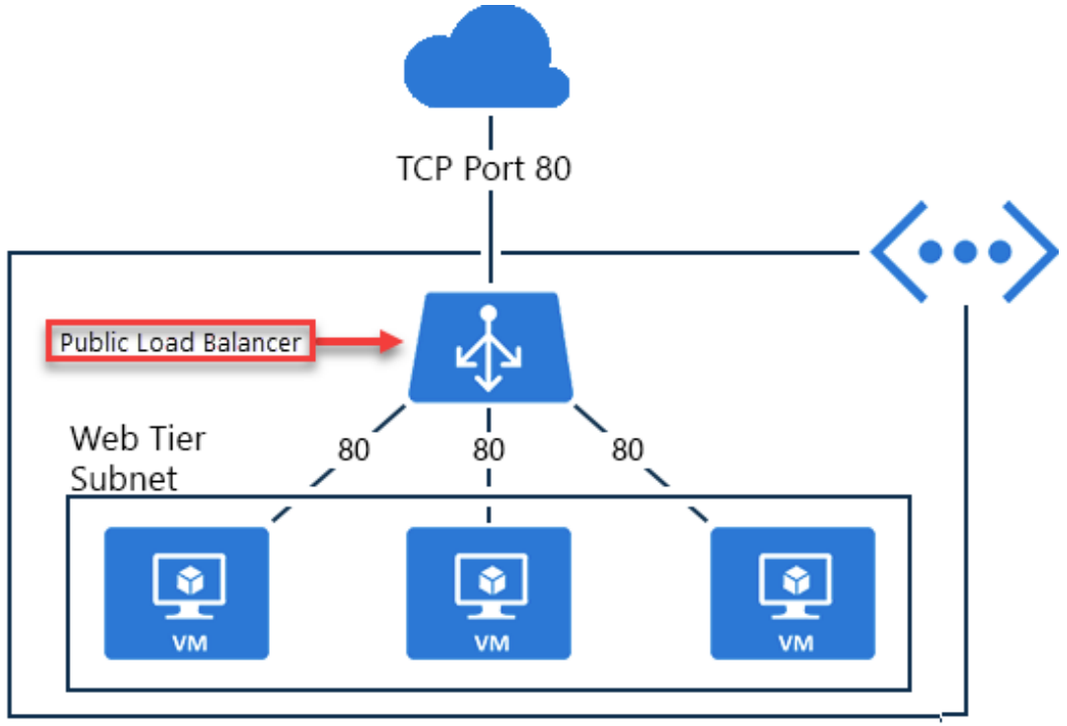


Public Load Balancer

Distributes traffic targeting a public IP address across backend VMs:

Frontend has one or more public IP addresses

Backend VMs have private IP addresses



Internal Load Balancer

Distributes traffic targeting a private IP address across backend VMs:

Frontend has one or more private IP addresses

Backend VMs have private IP addresses

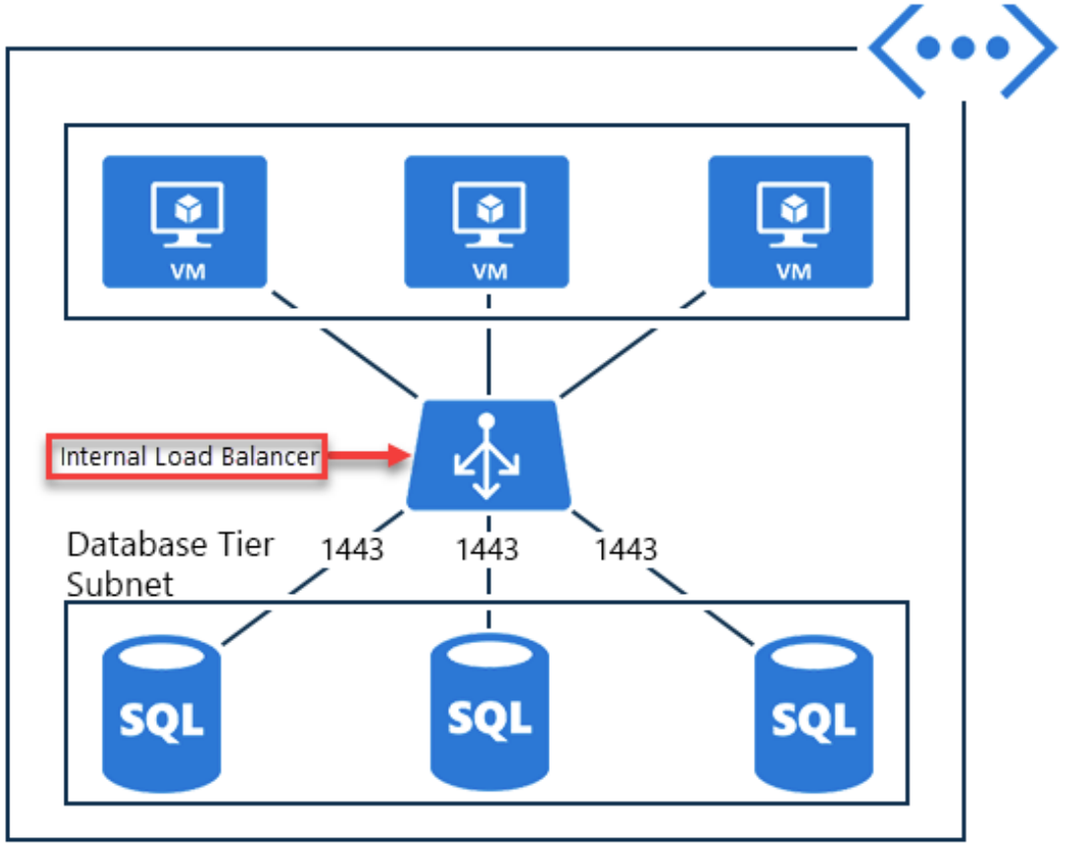
Supports load balancing:

Within a virtual network

For a cross-premises virtual network

For multi-tier applications

For line-of-business applications

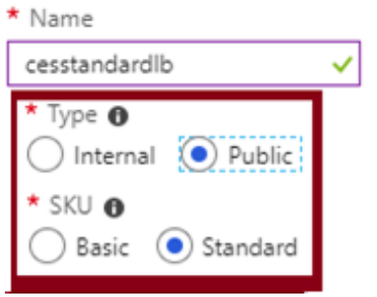


Load Balancer SKUs

Two SKUs:

Basic

Standard



Constraints and considerations:

SKUs are not mutable.

An Azure VM, Availability Set, or Azure VM Scale Set can reference one SKU, not both.

A Load Balancer rule cannot span two virtual networks.

There is no charge for the Basic load balancer.

The Standard load balancer is charged based on number of rules and data processed.

Backend Pool

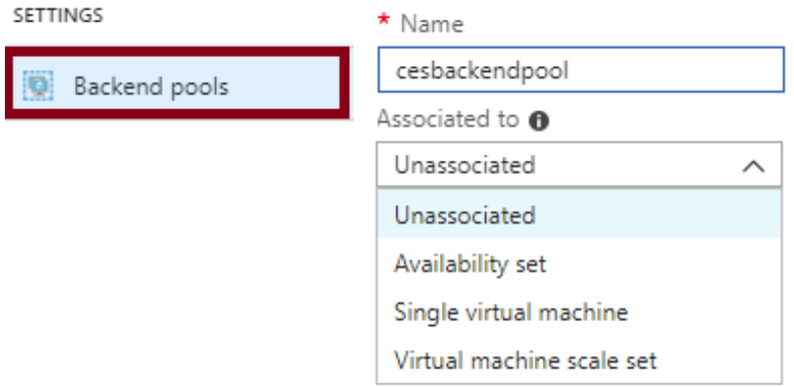
Configuration depends on the load balancer SKU:

Standard:

Up to 1,000 Azure VMs in the same virtual network, including VMs in availability sets and VM scale sets.

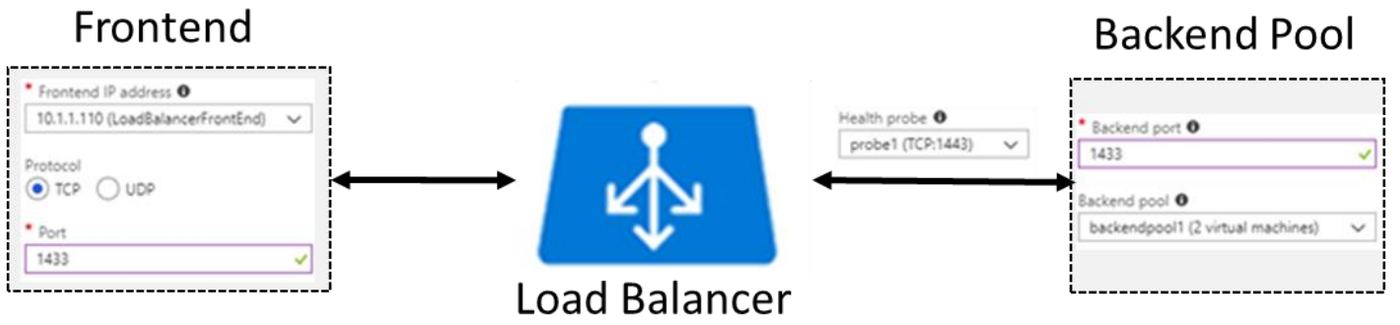
Basic:

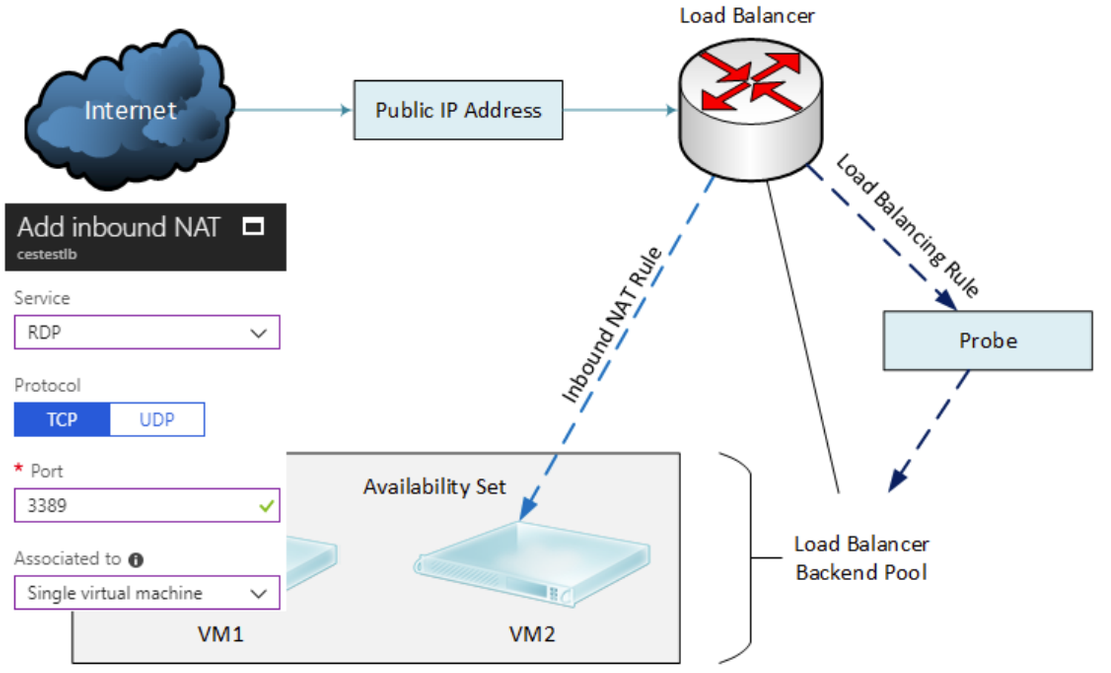
Up to 100 VMs in the same availability set or VM scale set.



Load Balancer Rules

Determine traffic distribution
Require existing:
Frontend IP
Backend pool
Health probe
Can be used with NAT rules:
Allow connections to specific backend VMs





Multiple Frontends

Azure Load Balancer supports load balancing on:

Multiple IP addresses

Multiple ports

Multiple IP addresses and ports

Two types of rules map frontend and backend pool configurations:

1. The default rule with no backend port reuse

2. The floating IP rule where backend ports are reused

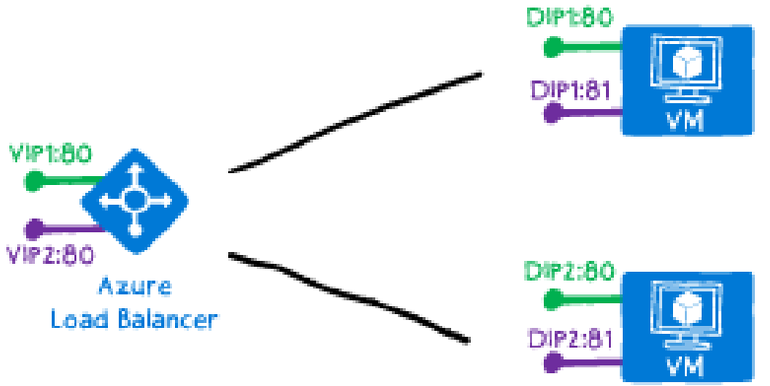
Rule type 1: No backend port reuse:

Frontends are configured with:

IP address, protocol, and port

Backend pool VMs expose each service on:

The same DIP and a unique port



Multiple Frontends (Rule 2)

Rule type 2: backend port reuse by using Floating IP

Requires enabling Floating IP in the rule definition:

Each backend pool VM has three network interfaces:

DIP. A Virtual NIC associated with the VM (IP configuration of Azure's NIC resource)

Frontend 1. A loopback interface within guest OS that is configured with IP address of Frontend 1

Frontend 2. A loopback interface within guest OS that is configured with IP address of Frontend 2

The destination of the inbound flow is the frontend IP address on the loopback interface in the VM.

By changing the destination IP address, you can enable port reuse on the same VM.

Session Persistence

Load balancing uses a hash to map traffic to backend pool VMs:

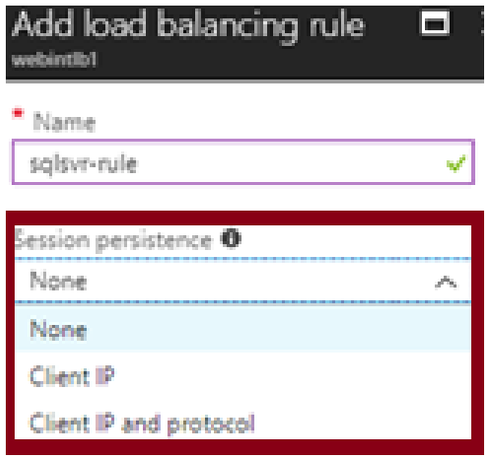
5-tuple (source IP, source port, destination IP, destination port, and protocol type)

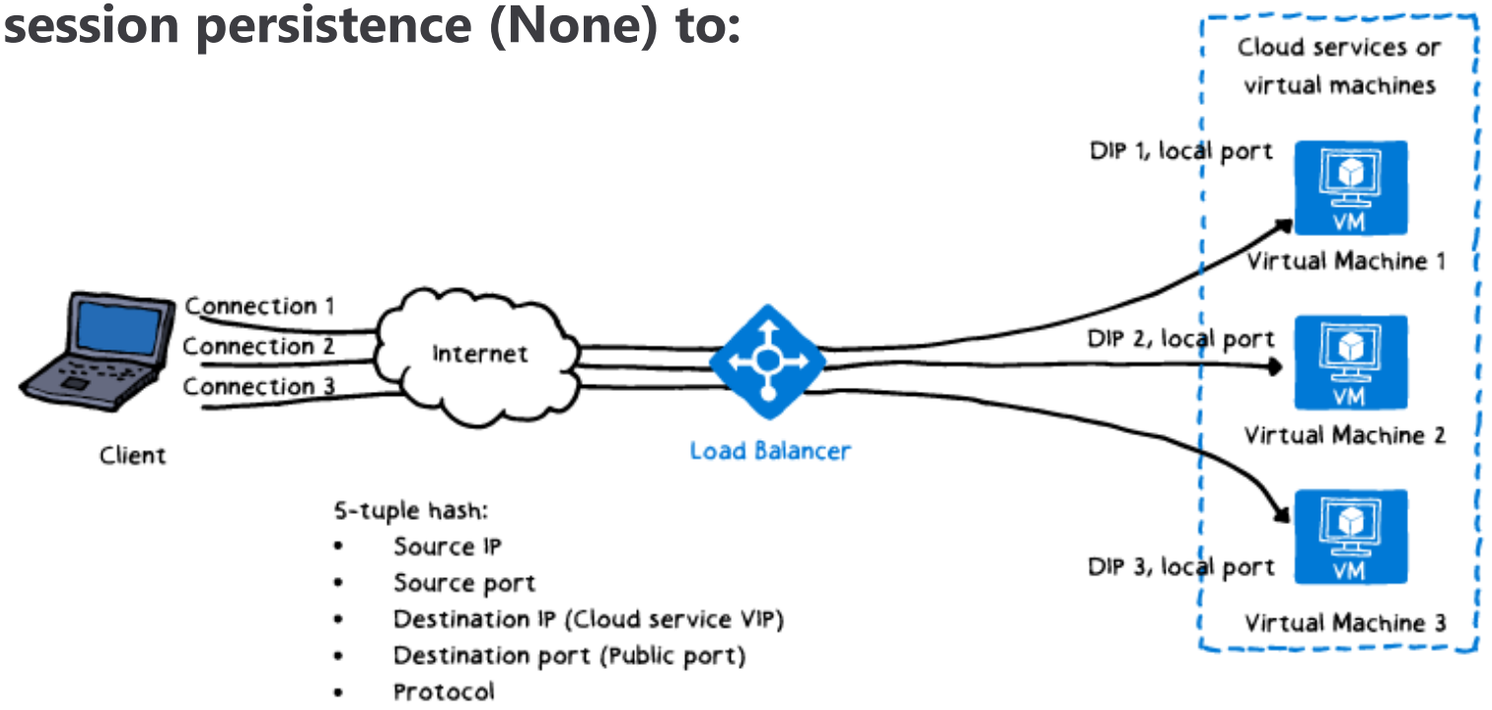
Stickiness applies only within a transport session.

You can change the default session persistence (None) to:

Client IP

Client IP and protocol





Health Probes

Evaluate status of load balanced workloads:

Unhealthy threshold set to 2 consecutive failures (default)

Interval set to 15 second (default)

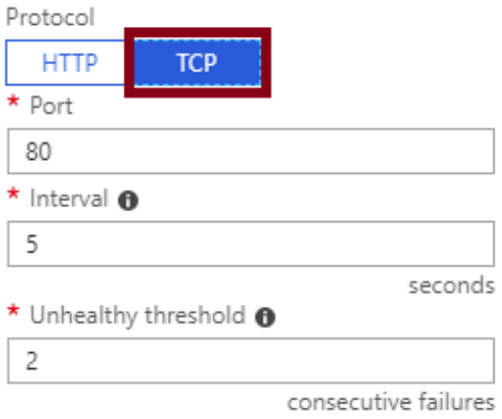
Support two protocols:

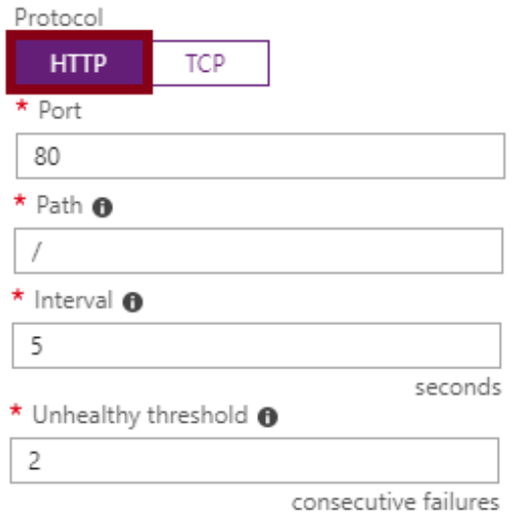
HTTP:

Expects HTTP 200 OK response

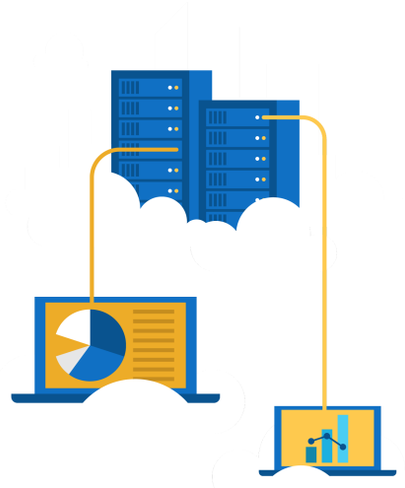
TCP:

Tests for a successful TCP session





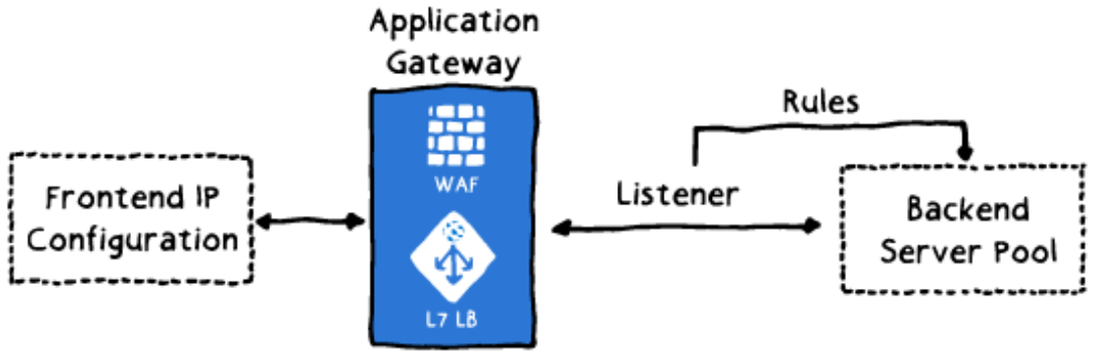
Azure Application Gateway



Application Gateway Components

Application Gateway is a load balancer operating on OSI Layer
Its components include:

Frontend IP configuration
Backend server pool
Listeners, including:
front-end port
protocol (HTTP or HTTPS)
SSL certificate (optional).
Rules
Web application firewall (WAF)



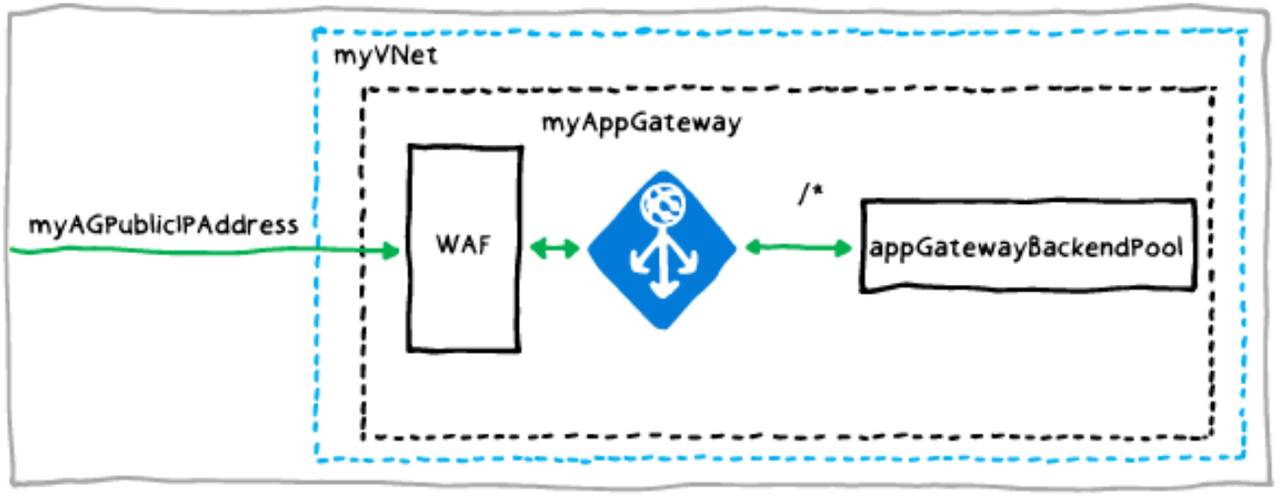
Web Application Firewall

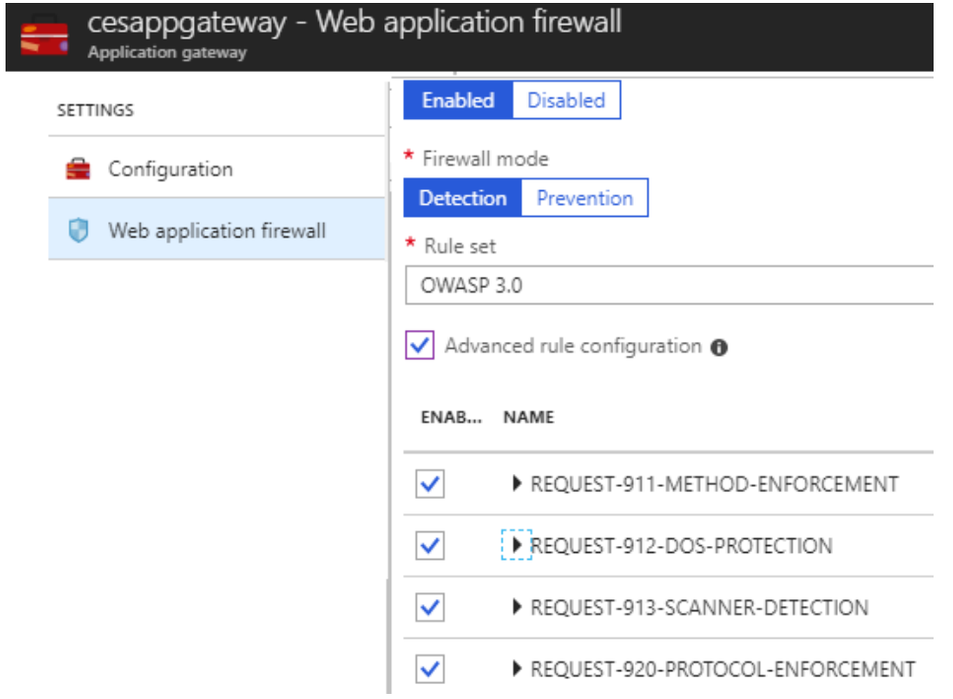
Provides protection for backend server pool workloads:

Protects against common cyber threats (SQL injection, cross-side scripting, etc.)

Uses OWASP rules

Allows disabling rules that result in false positives





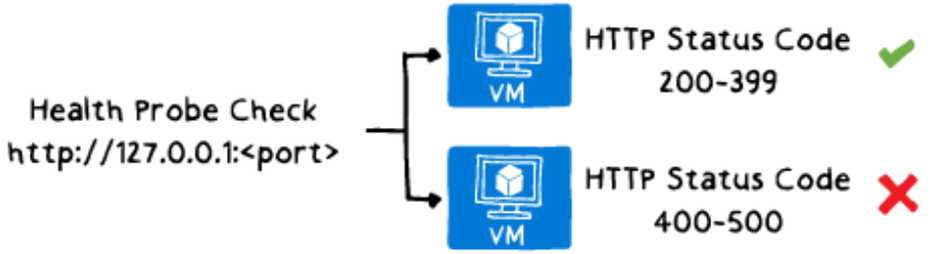
Health Probes

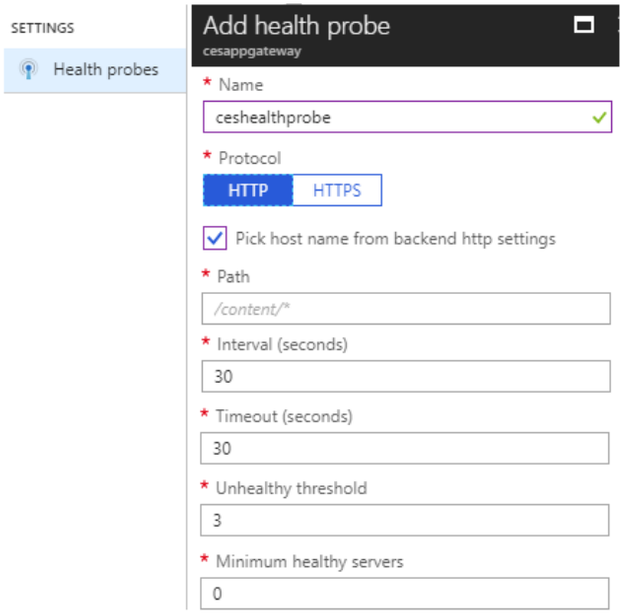
By default, health probes relies on healthy HTTP responses
Custom probes provide more control:

Facilitate more thorough health checks

Support custom values of:

Minimum healthy servers
Unhealthy threshold
Interval
Timeout
Path

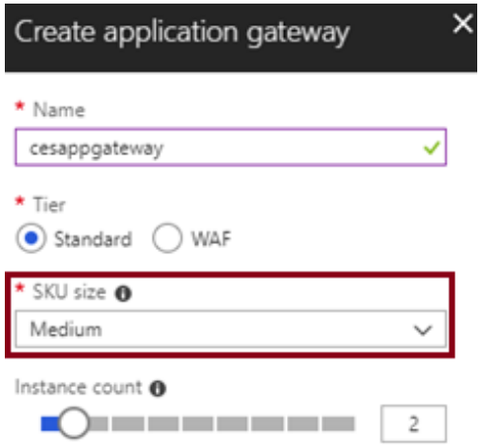




Application Gateway Sizing

Application Gateway is available in 3 SKUs:

Small: intended for development and testing only
Medium
Large



Average back-end page response size	Small	Medium	Large
6KB	7.5 Mbps	13 Mbps	50 Mbps
100KB	35 Mbps	100 Mbps	200 Mbps

Path-Based Routing

Directs traffic based on target URL, e.g.:

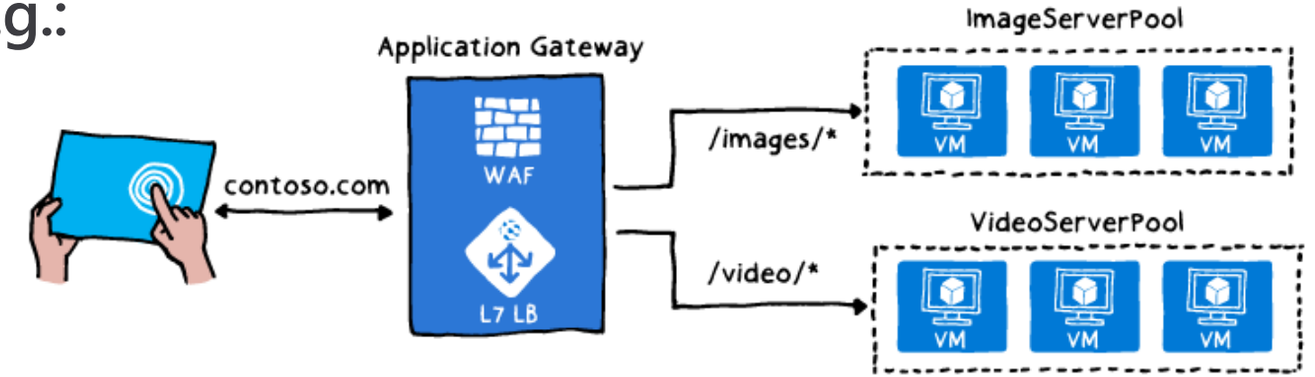
/images to one backend pool

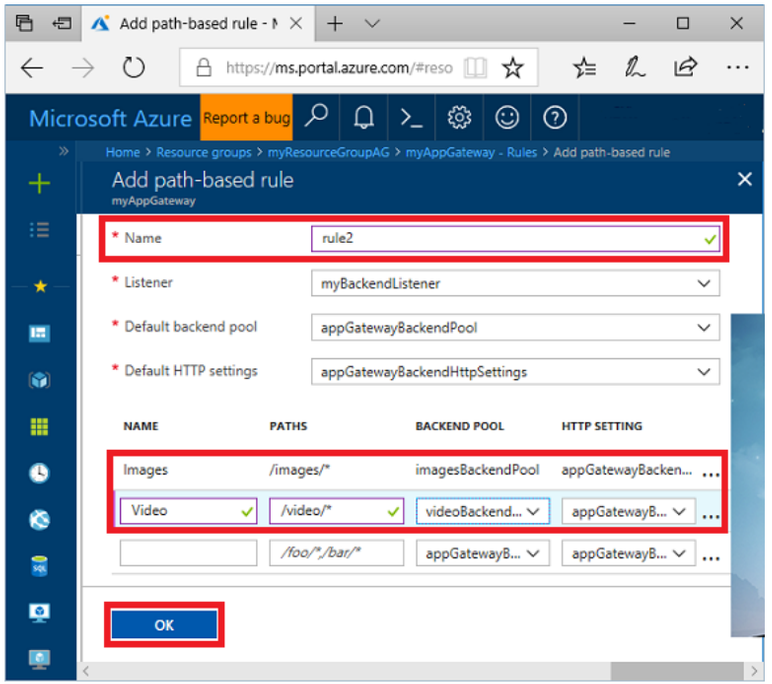
/video to another backend pool

To implement:

Specify the path pattern, e.g.:

/images/* and /video/*





Multiple Site Hosting

Allows multiple web sites on the same Application Gateway instance:

Each with its own backend pool

Up to the total of 20 web sites

To implement:

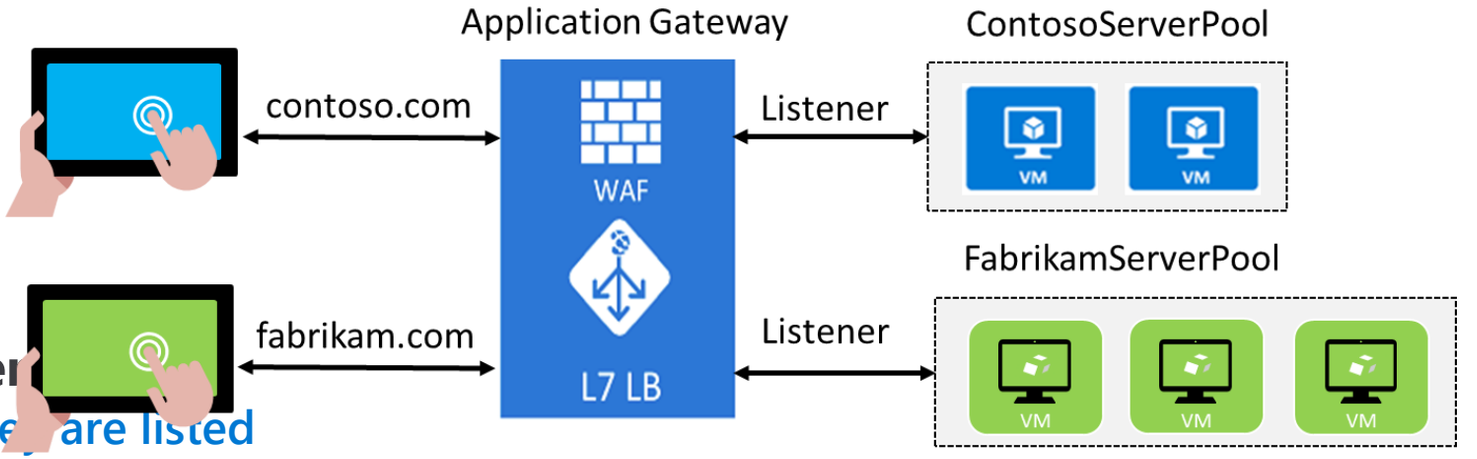
Create 2 backend pools

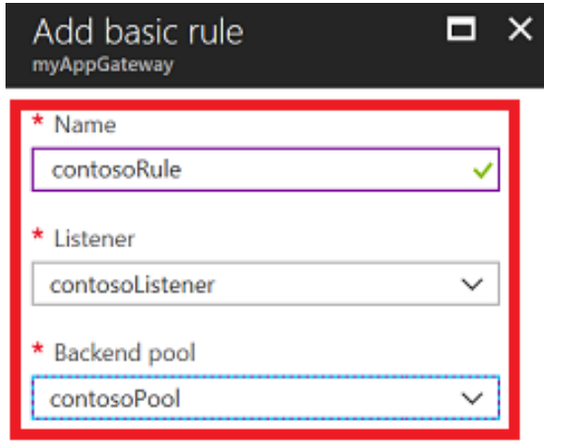
Create 2 listeners

Create 2 routing rules

Arrange the rules in the intended order

Rules are processed in the order in which they are listed





Secure Sockets Layer Offload

Provides SSL termination at the gateway:

Eliminates performance impact of decryption on the backend pool VMs

Requires uploading certificate and binding it to the appropriate listener

Redirection and Session Affinity

Redirection:

Protocol redirection:

Typically HTTP to HTTPS

Path-based redirection:

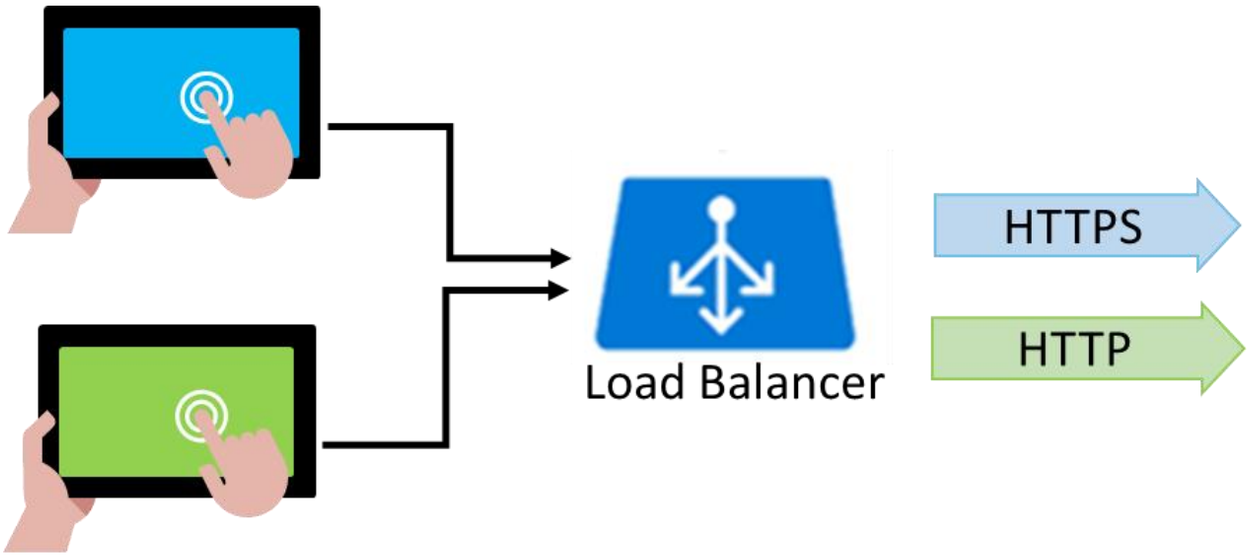
Apply protocol redirection for specific path only:
e.g. /cart/*

Redirection to external sites

Session affinity:

Cookie-based

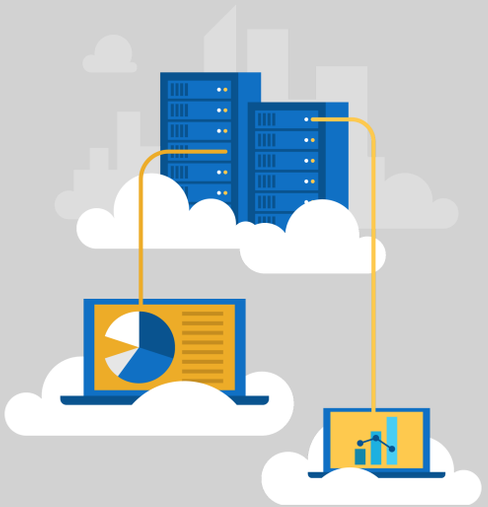
Directs traffic to the same backend pool VM



Microsoft Azure

Integrate on premises network with Azure virtual network

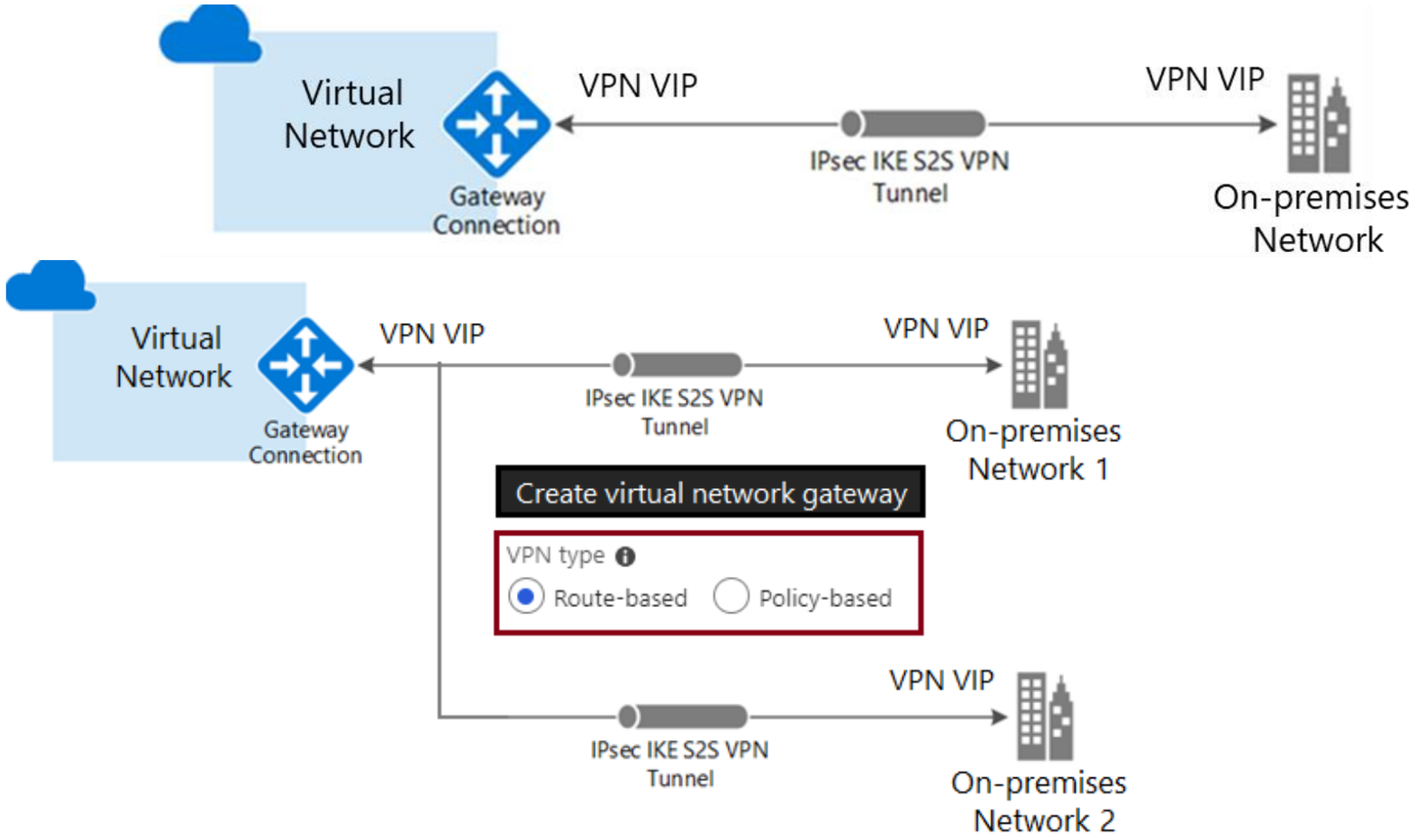
Site-to-Site VPN Connections



Site-to-Site VPN Connections

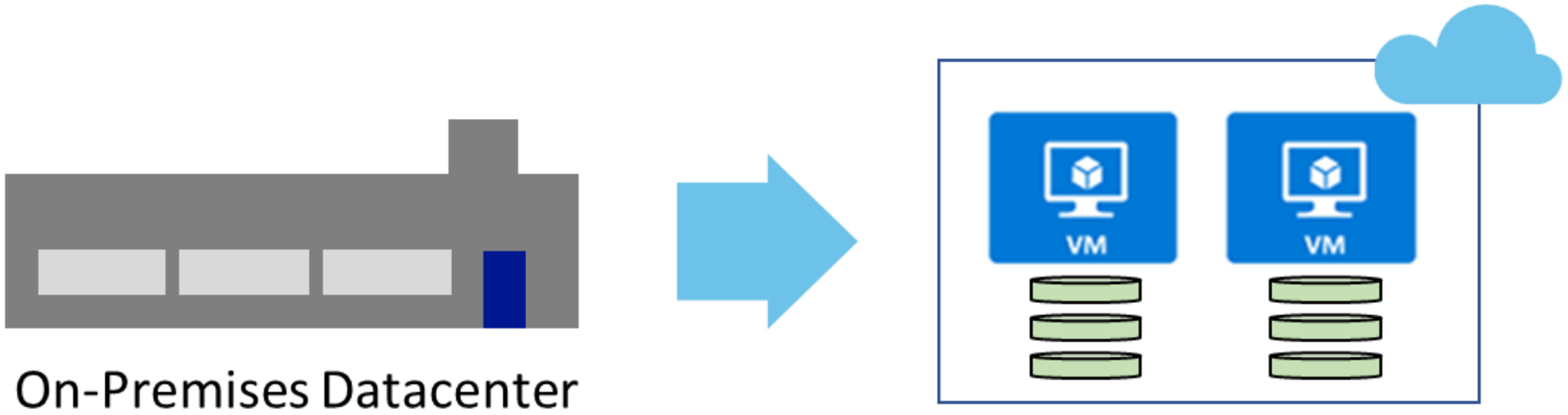
IPSec IKEv1 or IKEv2 VPN tunnels:

Cross-premises
VNet-to-VNet
Multi-site:
Requires a route-based VPN



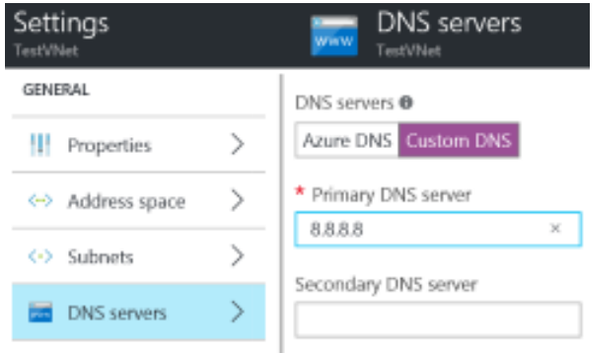
Site-to-Site Scenarios

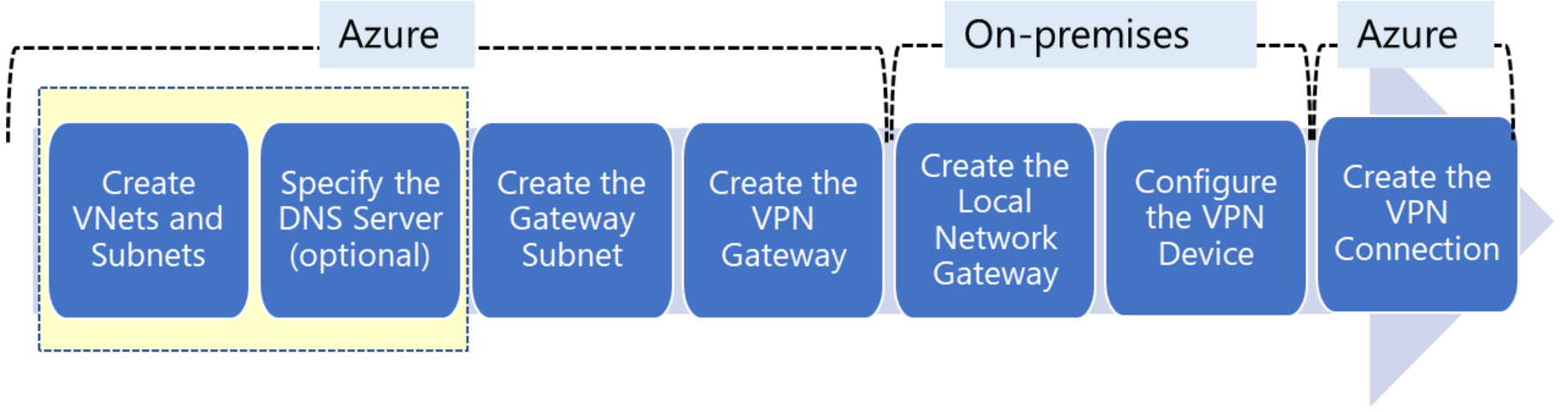
Capacity On-Demand
Strategic Migration
Disaster Recovery



Implementing Site-to-Site VPN

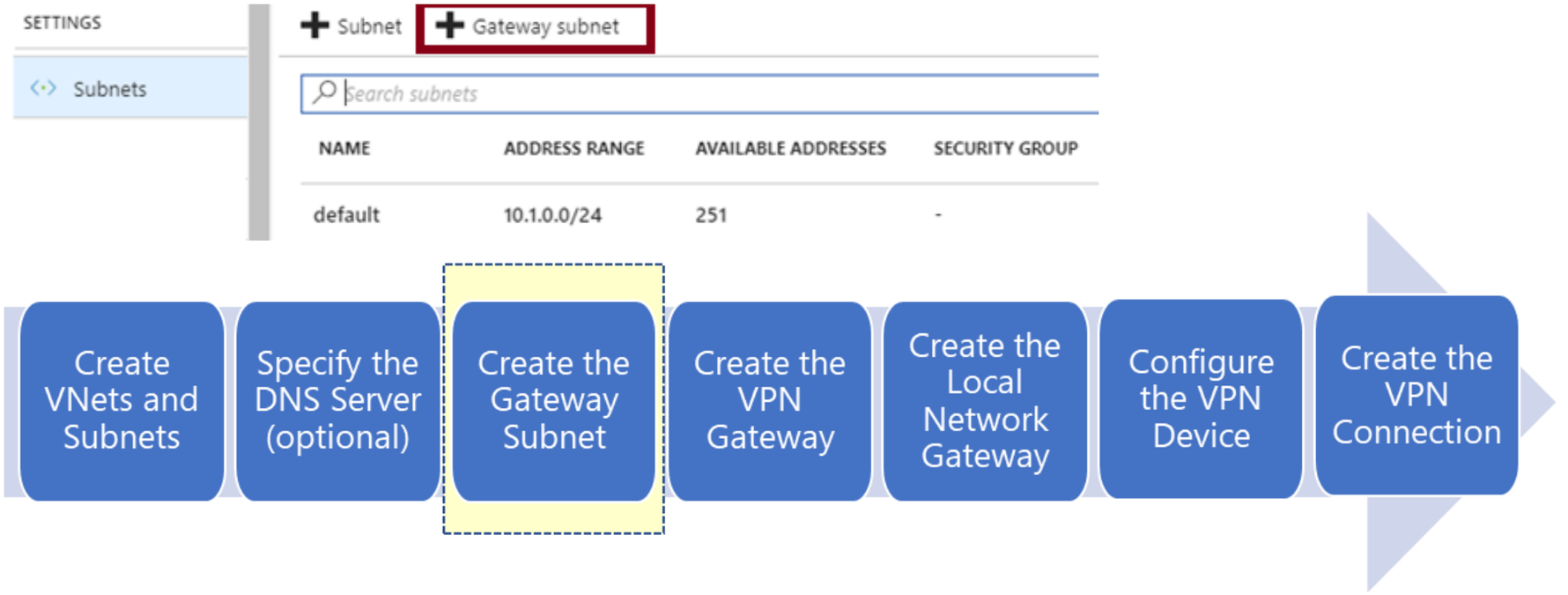
1. Create VNets and subnets
2. Specify the DNS server (optional)
3. Create the Gateway subnet
4. Create the VPN Gateway
5. Create the Local Network Gateway
6. Configure the VPN device
7. Create the VPN Connection





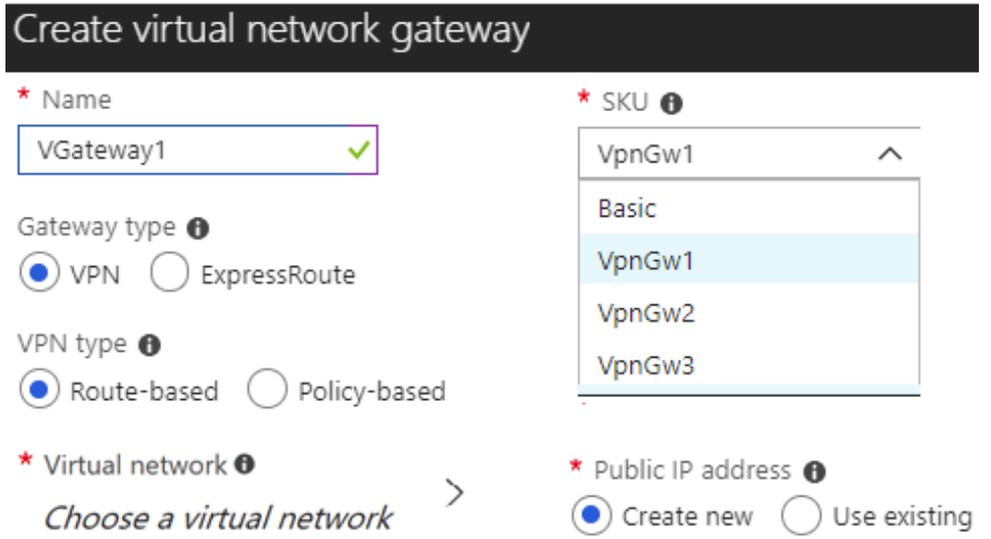
Gateway Subnet

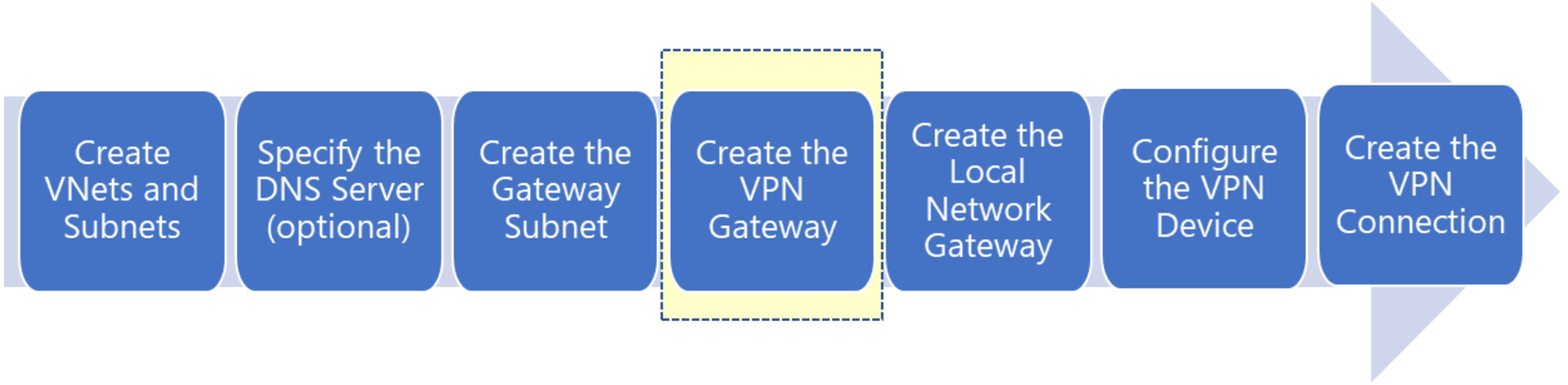
Requires an IP range of /28 or larger
Must be named GatewaySubnet
Should not be associated to any Network Security Groups



VPN Gateway

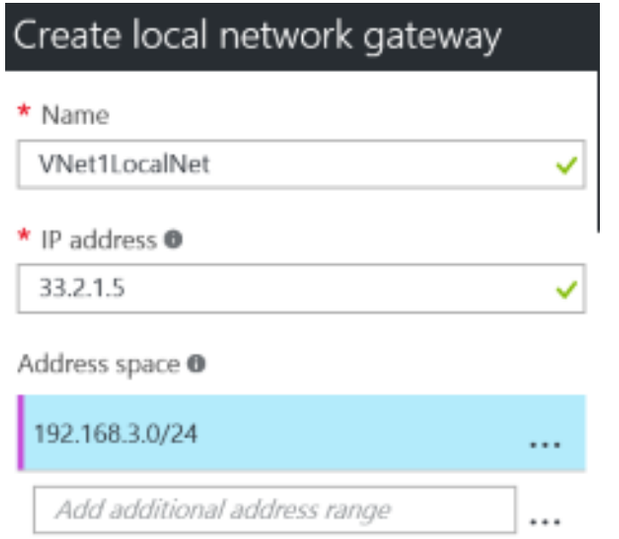
Establishes IPSec tunnel:
Handles traffic encryption
Supports multiple connections
One per GatewaySubnet
Supports two VPN types:
Route-based
Policy-based
Supports four SKUs
Basic, VpnGw1, VpnGw2, VpnGw3

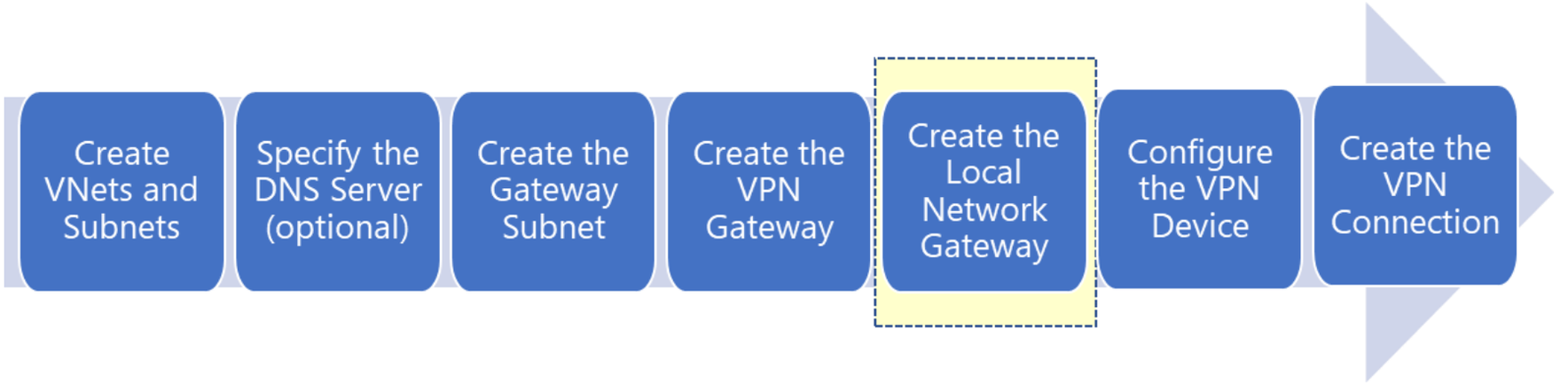




Local Network Gateway

In hybrid scenarios, represents on-premises VPN device
To configure, specify:
Name
IP address: the public IP address of the VPN device
Address space: one or more on-premises IP address ranges





Configure the VPN Device

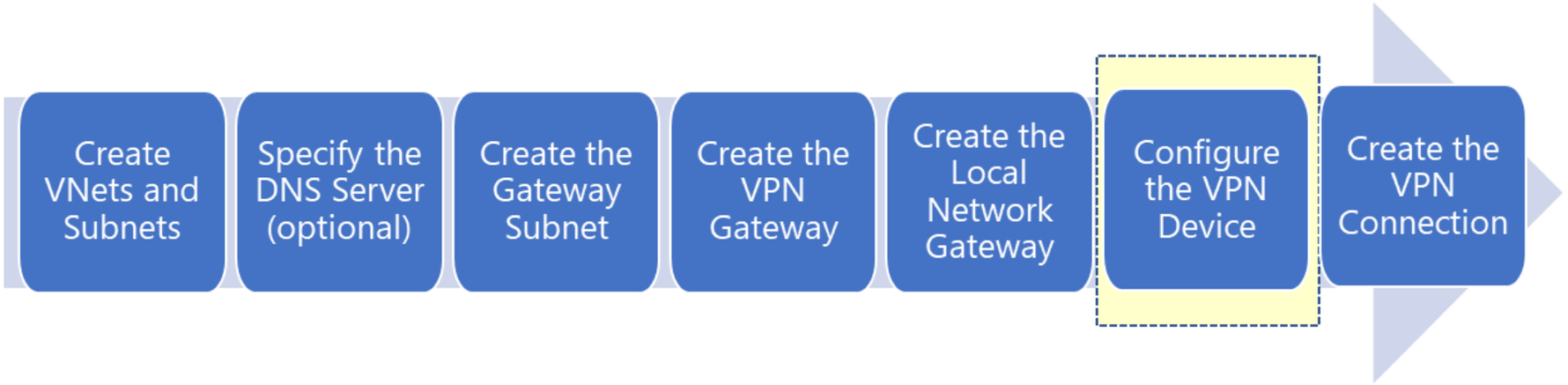
Follow instructions provided by Microsoft or device manufacturer:

Microsoft provides a list of the validated VPN devices

The configuration settings include:

A shared key

The public IP address of the VPN gateway



Configure the VPN Connection

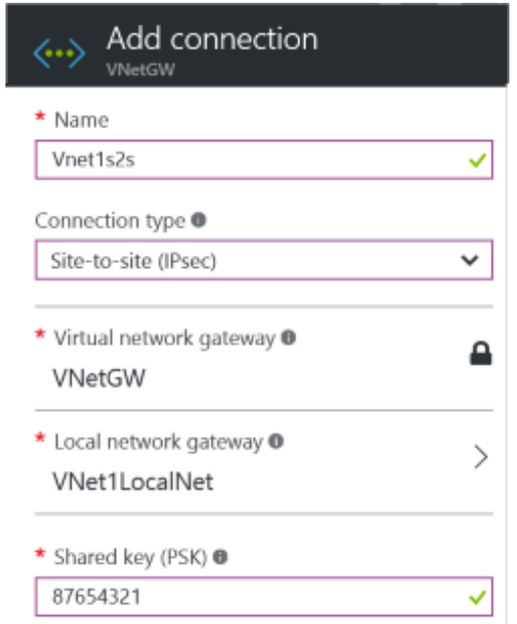
Establishes IPSec tunnel between:

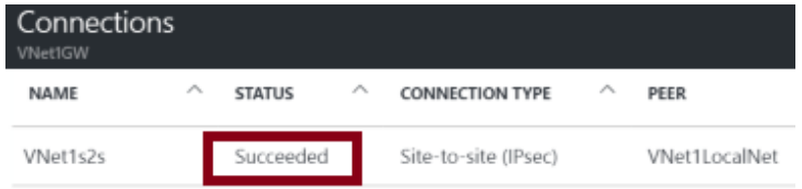
Azure VPN gateway

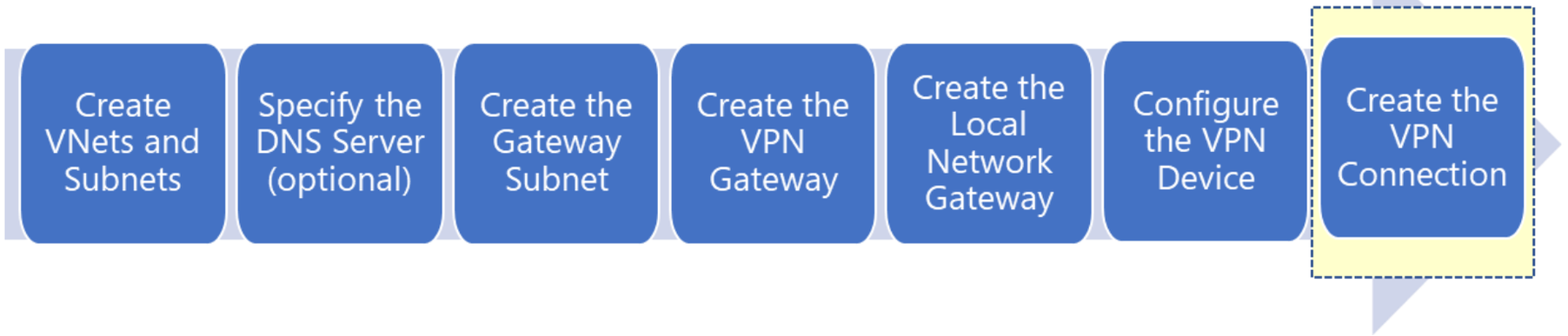
Local network gateway

Requires:

Shared key matching the one you used in the previous step







Verify the VPN Connection

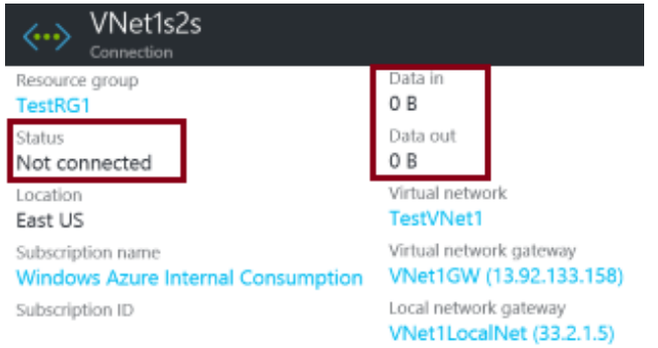
The Azure portal:

Identify the connection status:

Connected

Succeeded

Note the Data in and Data out values



Azure Powershell:

Run the Get-AzureRmVirtualNetworkGatewayConnection cmdlet

Examine the values of:

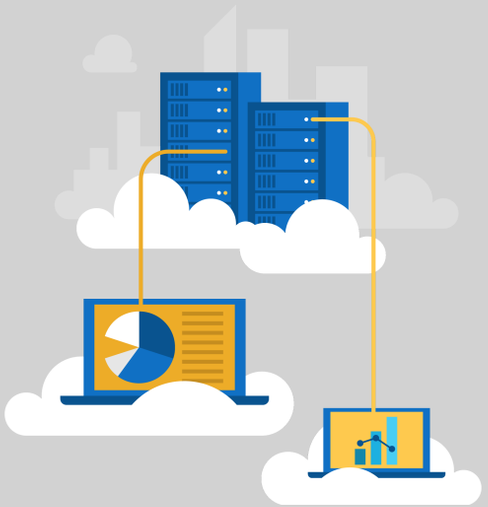
connectionStatus

ingressBytesTransferred

egressBytesTransferred

```
"connectionStatus": "Connected",
"ingressBytesTransferred": 33509044,
"egressBytesTransferred": 4142431
```

ExpressRoute



ExpressRoute

Private connection that extends on-premises network to:

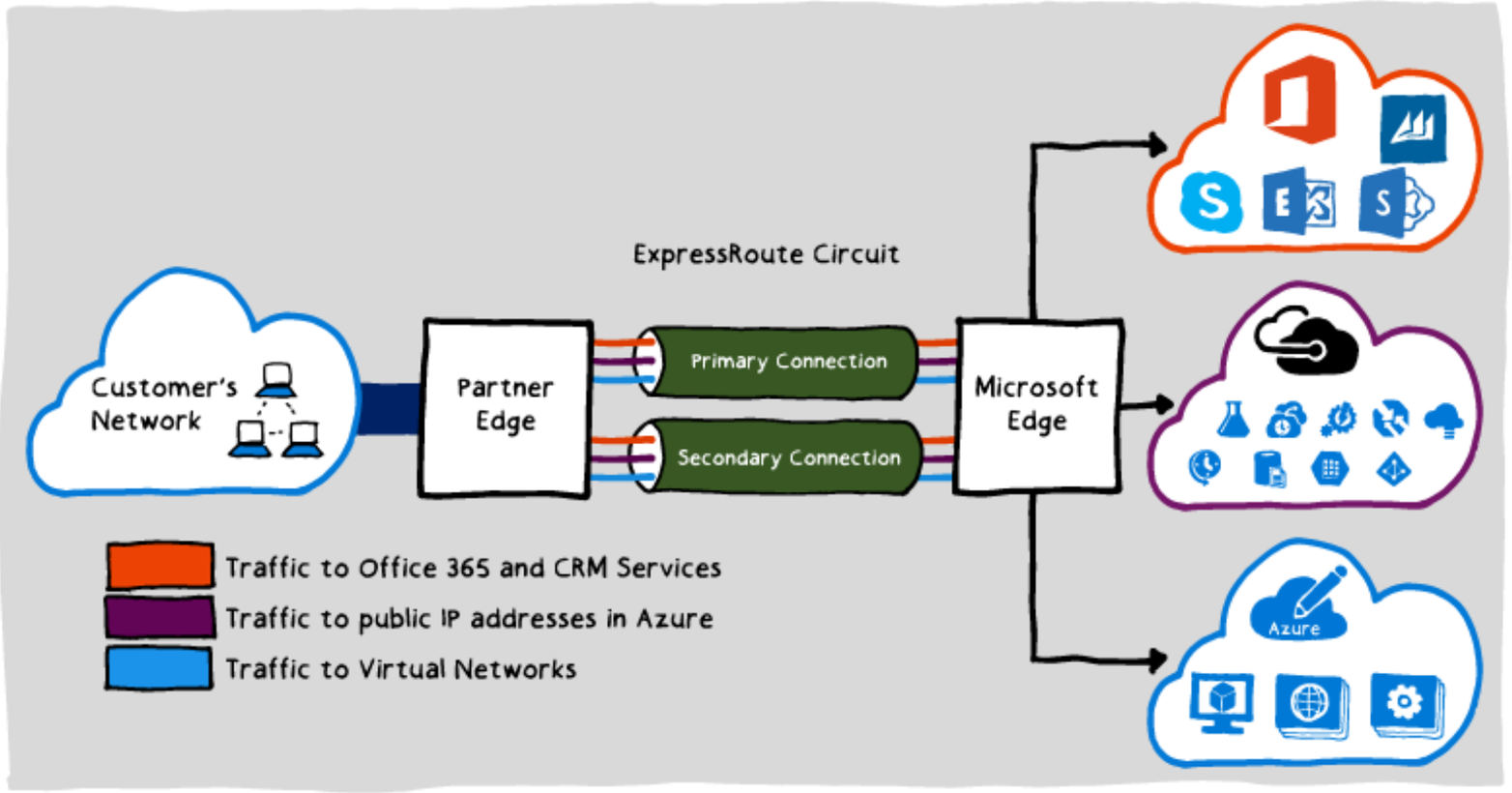
Microsoft Azure
Office 365
Dynamics 365

Benefits:

Enhanced reliability
Higher bandwidth
Lower latency
Increased security

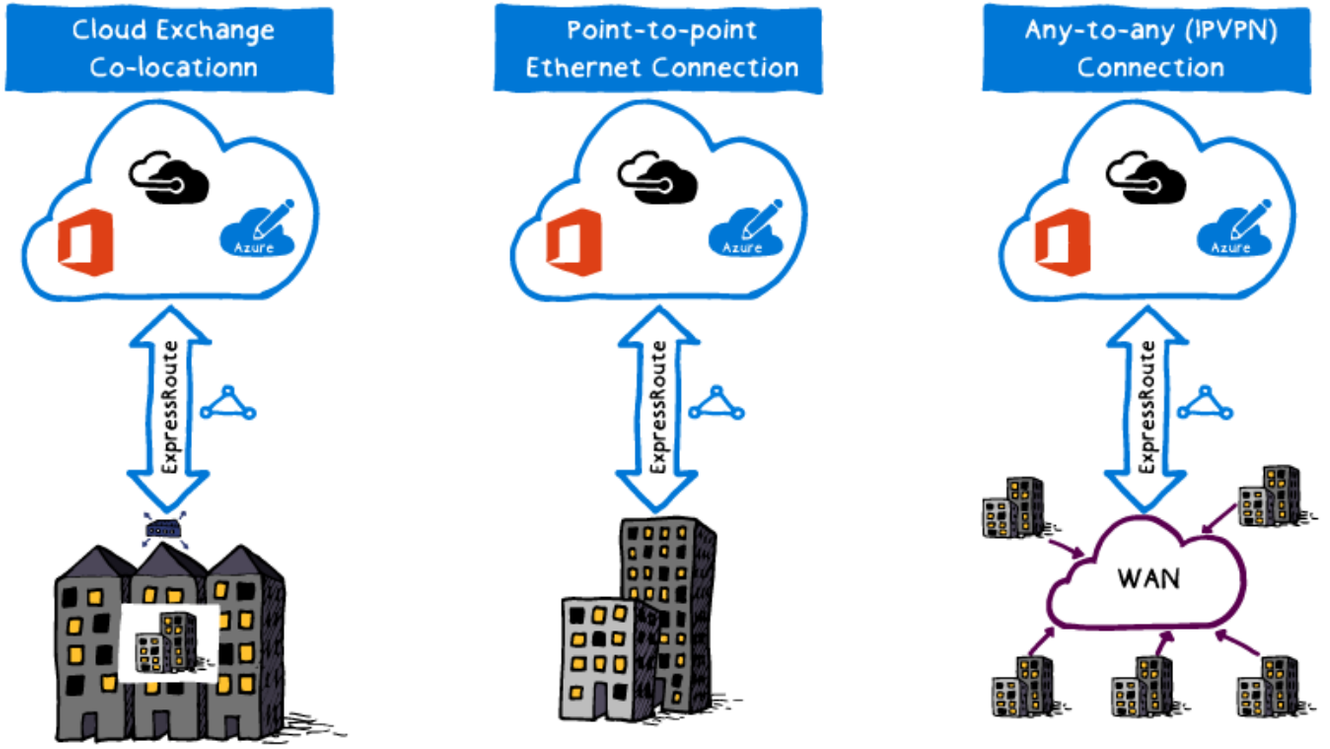
Common scenarios:

Data migration
Business continuity



ExpressRoute Connection Options

Cloud Exchange Co-location
Point-to-point Ethernet Connection
Any-to-any (IPVPN) Connection



Site-to-Site and ExpressRoute Coexisting Connections

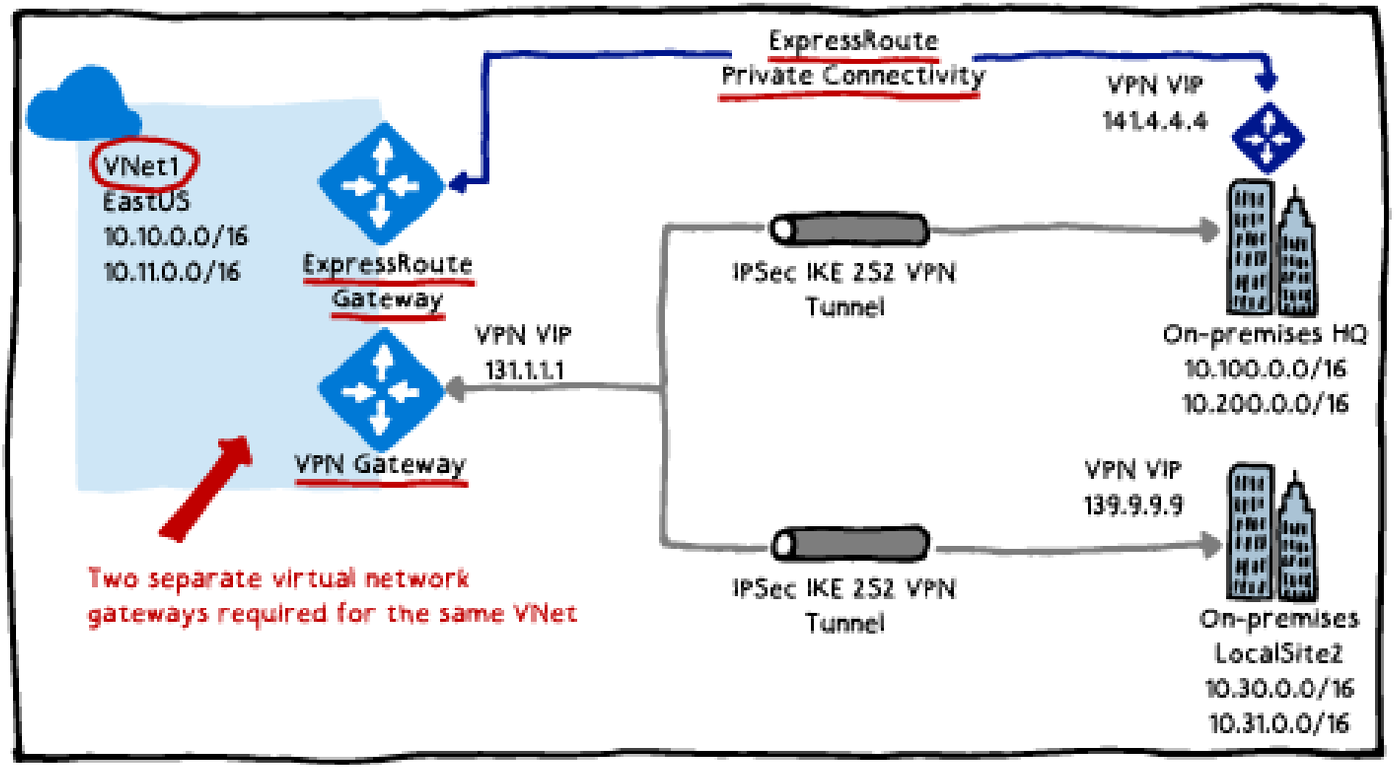
Configuration:

VPN gateway and ExpressRoute gateway are deployed to the same GatewaySubnet

Benefits:

Extended scope of connectivity

A cost effective failover

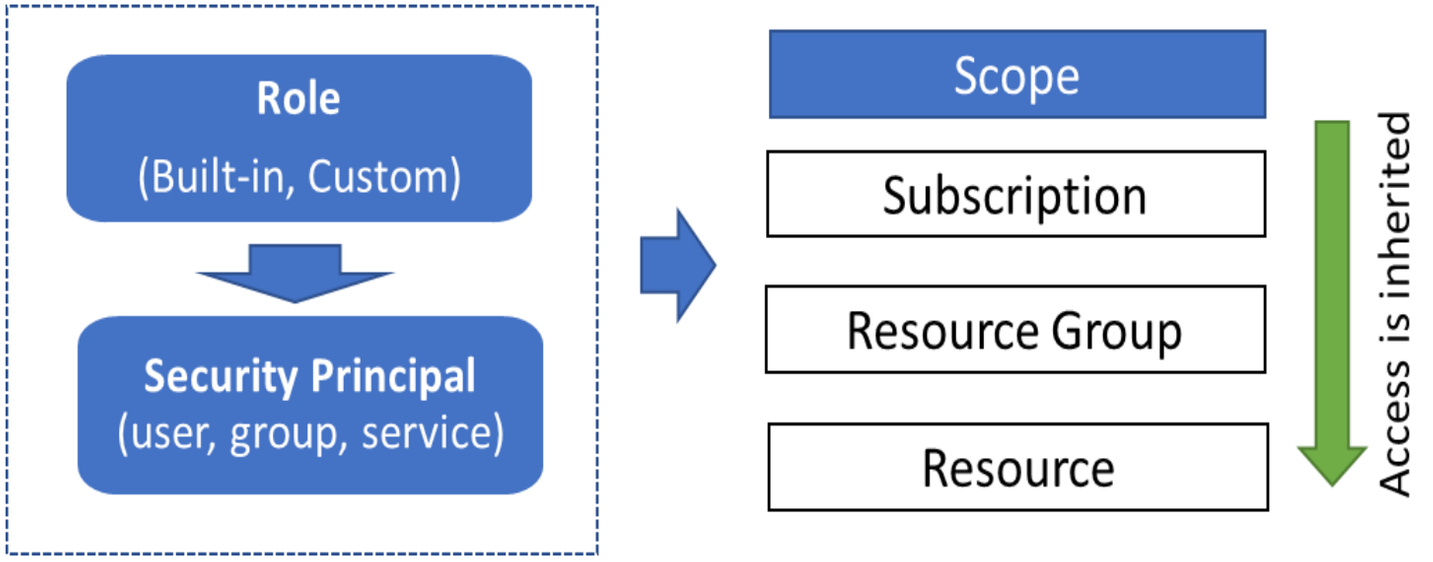


Microsoft Azure

Manage role-based access control (RBAC)

Role-Based Access Control

1. Select a role (the definition of what actions are allowed and/or denied)
2. Associating the role with a security principal (user, group, or service)
3. Scope to a subscription, a resource group, or a specific resource



Role-Based Access Control

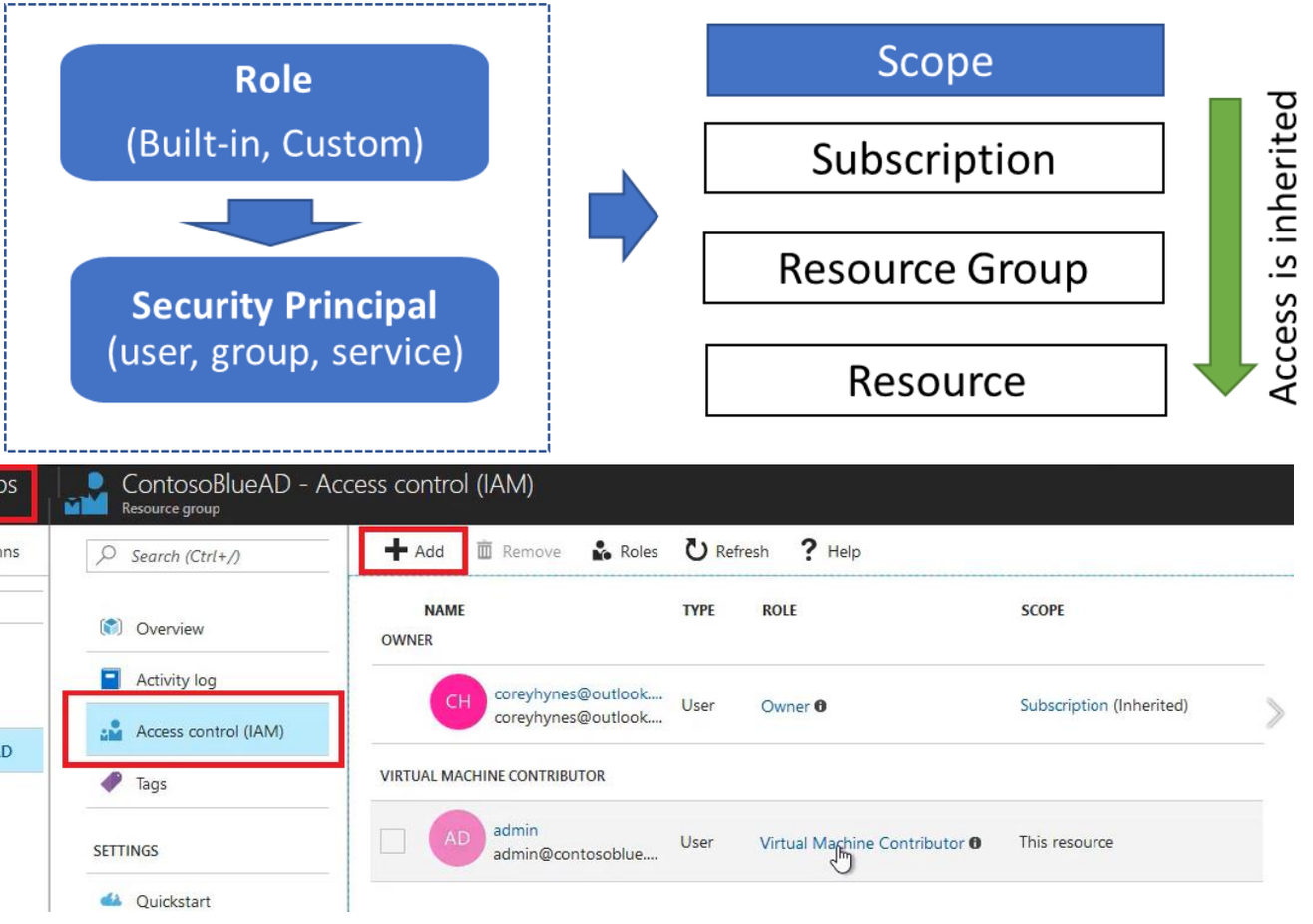
Facilitates granular access to Azure resources:

Consists of:

Role definition
Role assignment
Scope of assignment

Can be managed by using:

The Azure portal
Azure PowerShell
Azure CLI
ARM templates



Built-in Roles

A role represents a set of permissions to carry out specific actions

Azure AD provides many built-in roles:

Built-in, not resource specific, including:

Owner: full access to all resources including the right to delegate access to others.

Contributor: creating and managing all resources but without the ability to delegate access to others.

Reader: viewing all resources (except for secrets).

Built-in, resource specific (e.g. Virtual Machine Contributor)

A role is described by a JSON-formatted role definition:

Name, Id, Actions, Not Actions, AssignableScopes

Can be retrieved by Get-AzureRmRoleDefinition

```
Get-AzureRmRoleDefinition -Name Owner
Name             : Owner
Id               : 8e3af657-a8ff-443c-a75c-2fe8c4bcb635
IsCustom         : False
Description      : Lets you manage everything, including
                   access to resources.
Actions          : {*}
NotActions       : {}
AssignableScopes : {/}
```

Role Definitions

Actions and NotActions:

Include or exclude actions associated with the role

AssignableScopes:

/subscriptions/[subscription id]
/subscriptions/[subscription id]/resourceGroups/[resource group name]
/subscriptions/[subscription id]/resourceGroups/[resource group name]/[resource]

Built-in Role	Action	NotActions
Owner (allow all actions)	*	
Contributor (allow all actions except writing or deleting role assignment)	*	Microsoft.Authorization/*/Delete, Microsoft.Authorization/*/Write, Microsoft.Authorization/elevateAccess/Action
Reader (allow all read actions)	*/read	

Azure PowerShell and CLI

To automate role management, you can use:

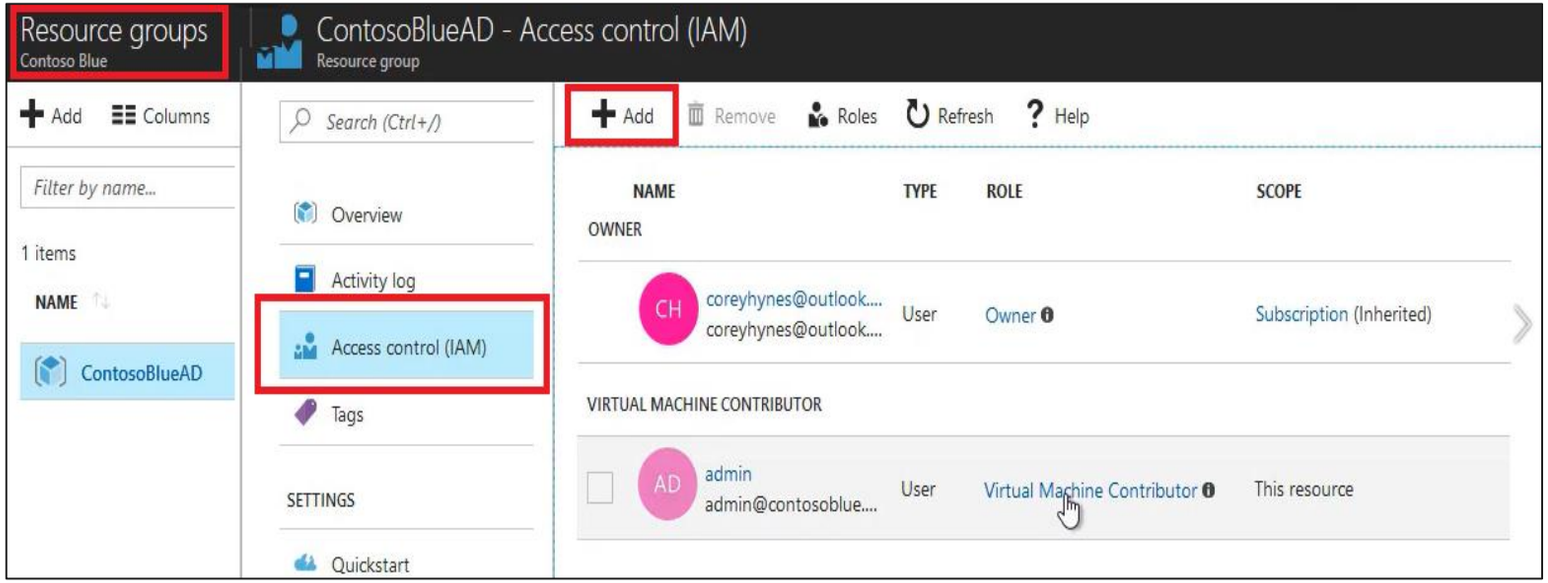
Azure PowerShell:

```
New-AzureRmRoleDefinition -InputFile .\sysops.json
New-AzureRmRoleAssignment -RoleDefinitionName $roleName `
                          -SignInName $assigneeName `
                          -ResourceGroupName $resourceGroupName
```

Azure CLI:

```
az role definition create --role-definition "./sysops.json"
az role assignment create --role $roleName \
                          --assignee $assigneeName \
                          --resource-group $resourceGroupName
```

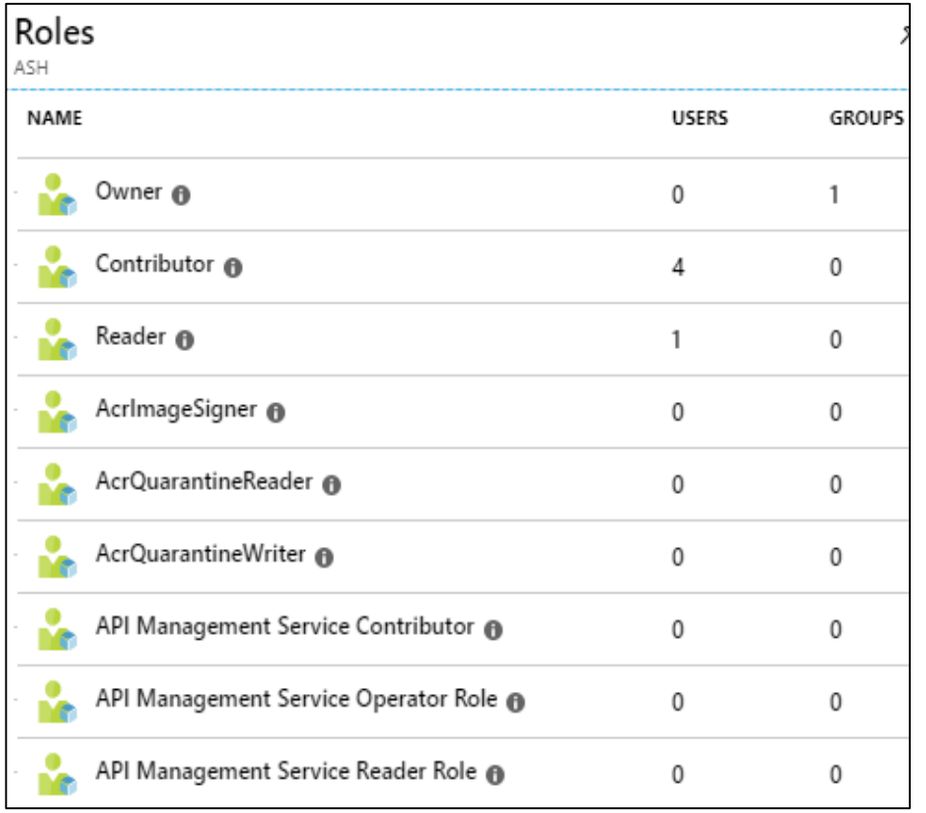
RBAC in the Portal



- You can use the Azure Portal to make your role assignments
- You can add or remove roles as you need
- You can add synced users and groups to Azure roles, which enables organizations to centralize the granting of access

Built-in Roles

- Azure AD has many built-in roles
- Owner has full access to all resources including the right to delegate access to others.
- Contributor can create and manage all types of Azure resources but can’t grant access to others
- Reader can view existing Azure resources



Role Definitions

- Each role has a role definition defined in a JSON file
- The **Actions** and **NotActions** properties allow or deny actions
- The **AssignableScopes** property specifies the affected subscriptions, resource groups, or resources

```
Name: Owner
ID: 8e3af657-a8ff-443c-a75c-2fe8c4bcb65
IsCustom: False
Description: Manage everything, including access to resources
Actions: {*}
NotActions: {}
AssignableScopes: {/}
```

Role Assignments (PowerShell and CLI)

- For large numbers of role assignments, use PowerShell or the CLI

  #Role assignment properties

  $roleName = “Contributor”

  $assigneeName = josh@microsoft.com

  $resourceGroupName = “contosoblue”

- Azure PowerShell

  New-AzureRmRoleAssignment -RoleDefinitionName $roleName -SignInName $assigneeName -ResourceGroupName $resourceGroupName

- CLI

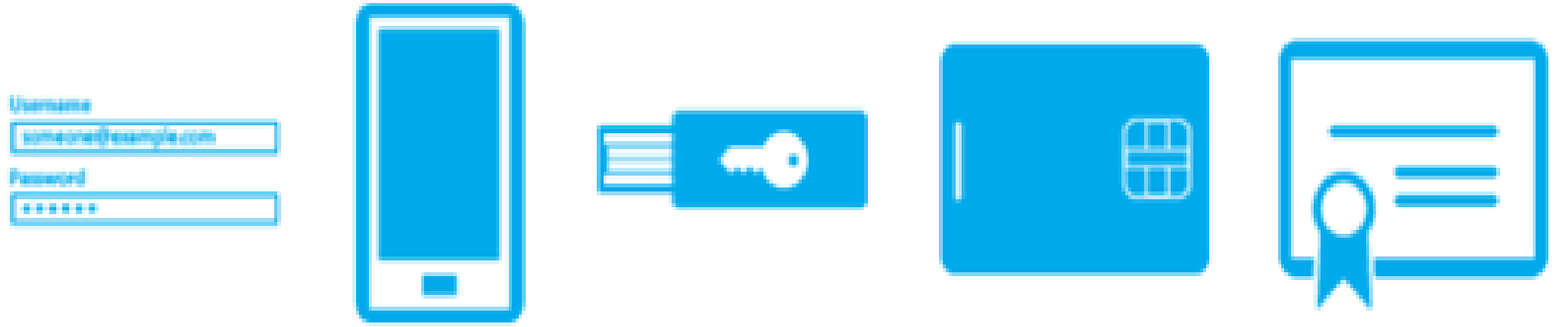
  az role assignment create –role $roleName –assignee $assigneeName –resource-group $resourceGroupName

Microsoft Azure

Implement Multi-Factor Authentication (MFA)

Azure MFA Concepts

- The security of MFA two-step verification lies in its layered approach
- Authentication methods include:
  - Something you know (typically a password)
  - Something you have (a trusted device that is not easily duplicated, like a phone)
  - Something you are (biometrics)



Azure MFA Features

- **Get more security with less complexity**
- **Mitigate threats with real-time monitoring and alerts**
- **Deploy on-premises or on Azure**
- **Use with Office 365, Salesforce, and more**
- **Add protection for Azure administrator accounts**

MFA Licensing and Pricing

- There are three pricing methods for Azure MFA.
- **Consumption based billing**
  - **Per user.** You can pay per user. Each user has unlimited authentications. Use this model if you know how many users you have and can accurately estimate your costs.
  - **Per authentication**. You can pay for a bundle (10) of authentications. Use this model when you are unsure how many users will participate in MFA authentication.
- **MFA licenses included in other products**
- **Direct and Volume licensing**

Microsoft Authenticator App

- Prevent unauthorized access to accounts
- Stop fraudulent transactions by giving you an additional level of security
- Use either as a second verification method or as a replacement for your password when using phone sign-in
- The app can work in one of two ways:
  - **Notification**. The app sends a notification to your device, then Verify or Deny
  - **Verification code**. Open the app and copy the verification code onto the sign-in screen

MFA for Global Admins

- Free
- Add
mac
- Secc
auth
- Use

s, like virtual

e

the portal to enable MFA for administrators

What are You Trying to Secure?

What are you trying to secure	Azure MFA	MFA Server
First-party Microsoft apps	●	●
SaaS apps in the app gallery	●	
Web applications published through Azure AD App Proxy	●	
IIS applications not published through Azure AD App Proxy		●
Remote access such as VPN, RDG	●	●

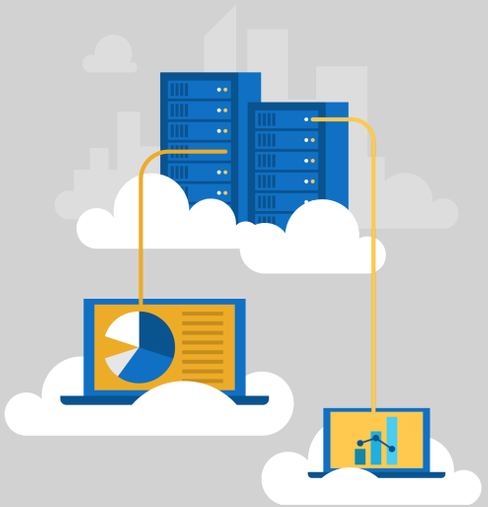
Where Are Your Users Located?

User Location	Azure MFA	MFA Server
Azure Active Directory	●	
Azure AD and on-premises AD using federation with AD FS	●	●
Azure AD and on-premises AD using Azure AD Connect - no password hash sync or pass-through authentication	●	●
Azure AD and on-premises AD using Azure AD Connect - with password hash sync or pass-through authentication	●	
On-premises Active Directory		●

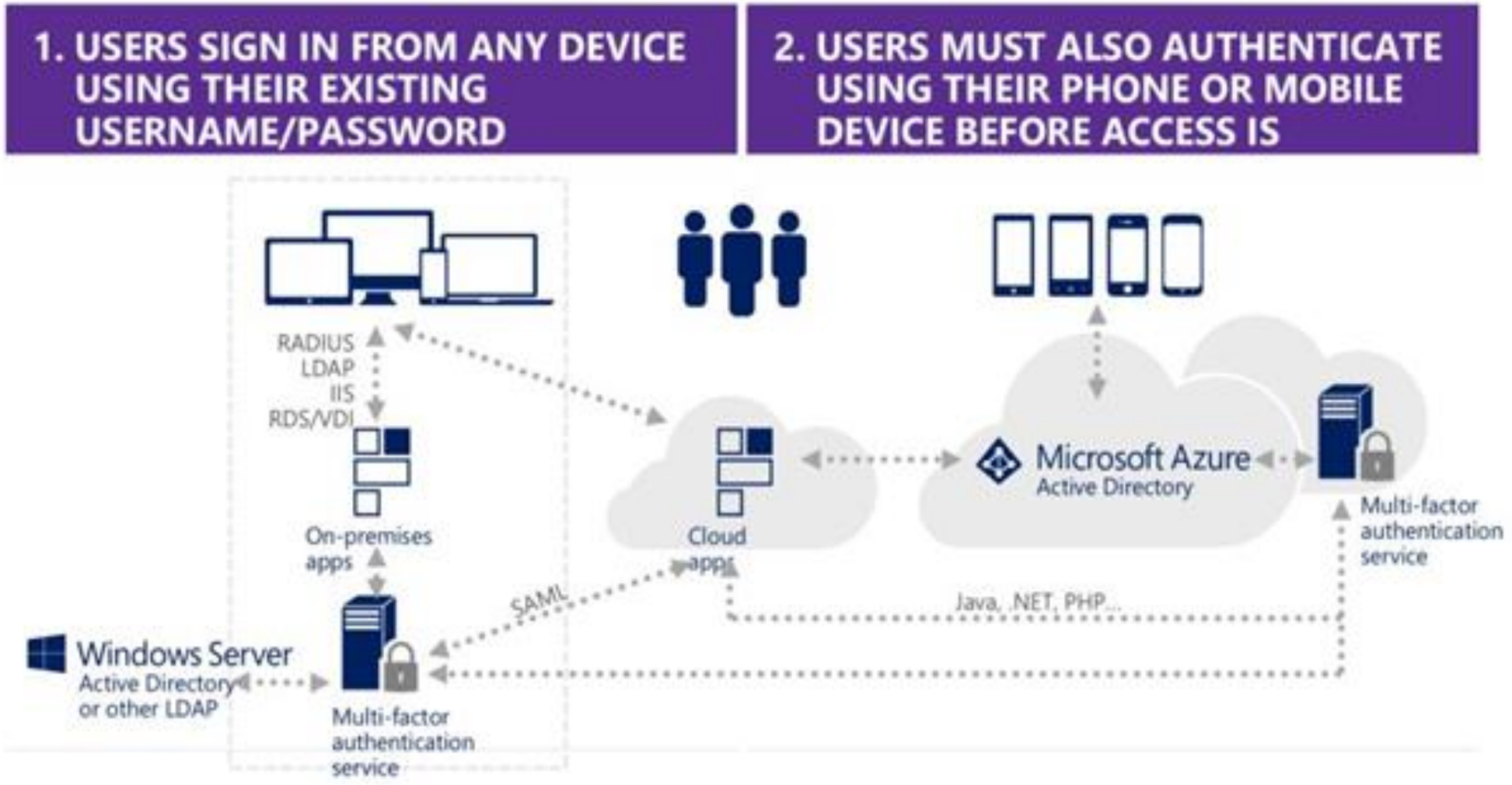
What Features Do You Need?

Feature	Azure MFA	MFA Server
Mobile app notification and mobile app verification code as a second factor	●	●
Mobile app verification code as a second factor	●	●
Phone call or one-way SMS as second factor	●	●
Hardware Tokens as second factor		●
PIN mode		●
Fraud alert and MFA reports	●	●
Remember MFA for trusted devices	●	
Conditional access	●	●

Implementing Multi-Factor Authentication

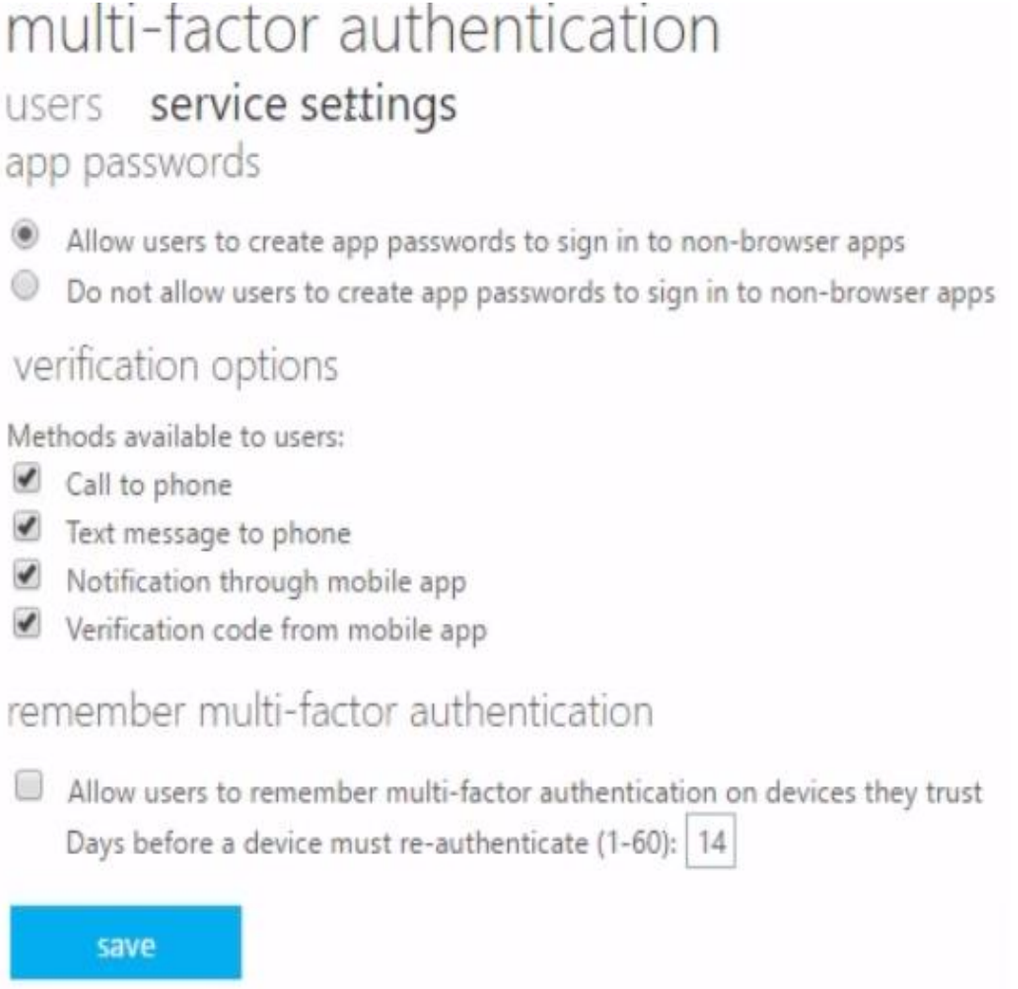


Video: How MFA Works



MFA User Settings

- Phone Call
- Text Message. A six-digit code is sent to the user's cell phone.
- Mobile App Notification
- Mobile app verification code. A six-digit code is sent to the user mobile app
- Cache passwords (1 to 60 days)

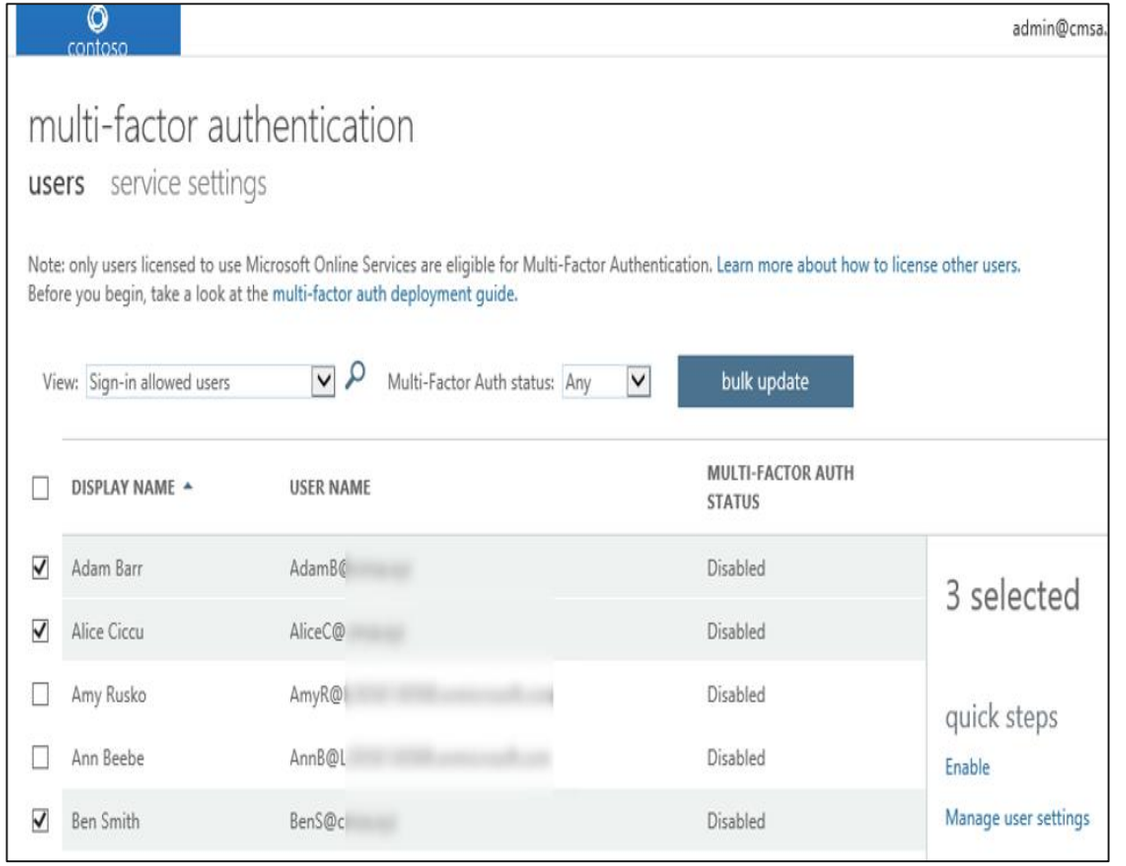


MFA vs SSPR

Authentication Method	Usage
Password	MFA and SSPR
Security questions	SSPR Only
Email address	SSPR Only
Microsoft Authenticator app	MFA and Public Preview for SSPR
SMS	MFA and SSPR
Voice call	MFA and SSPR
App passwords	MFA only in certain cases

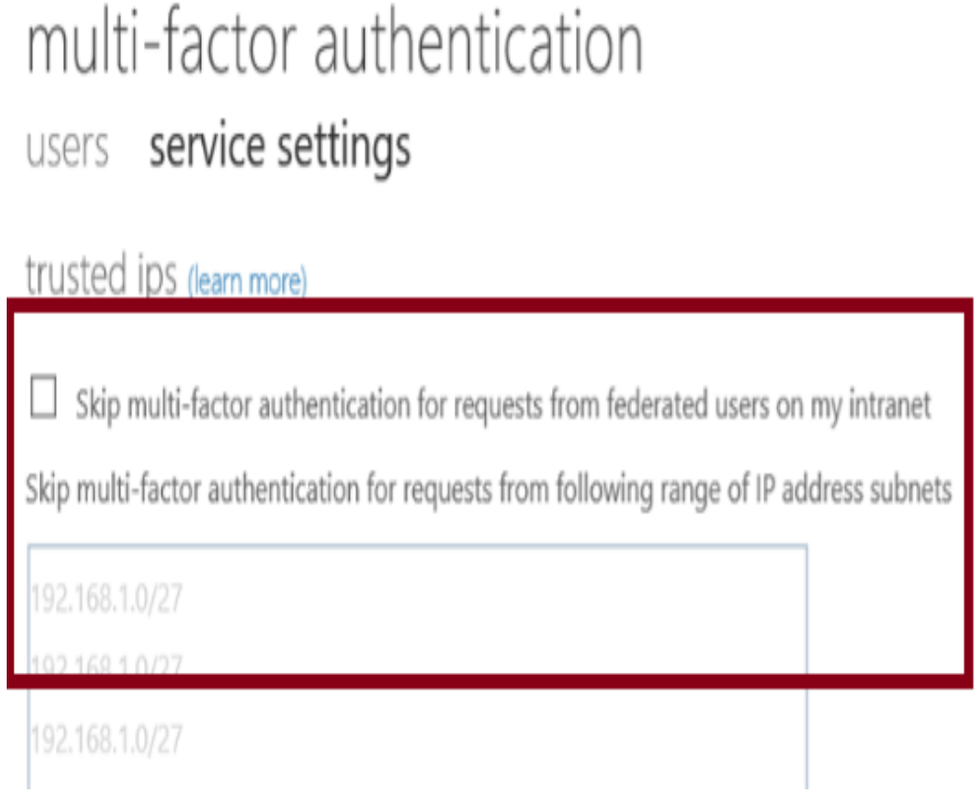
Enabling MFA

- Select the users that you want to modify and enable for MFA
- Can also bulk enable groups of users with PowerShell
- On first-time sign-in, after MFA has been enabled, users are prompted to configure their MFA settings

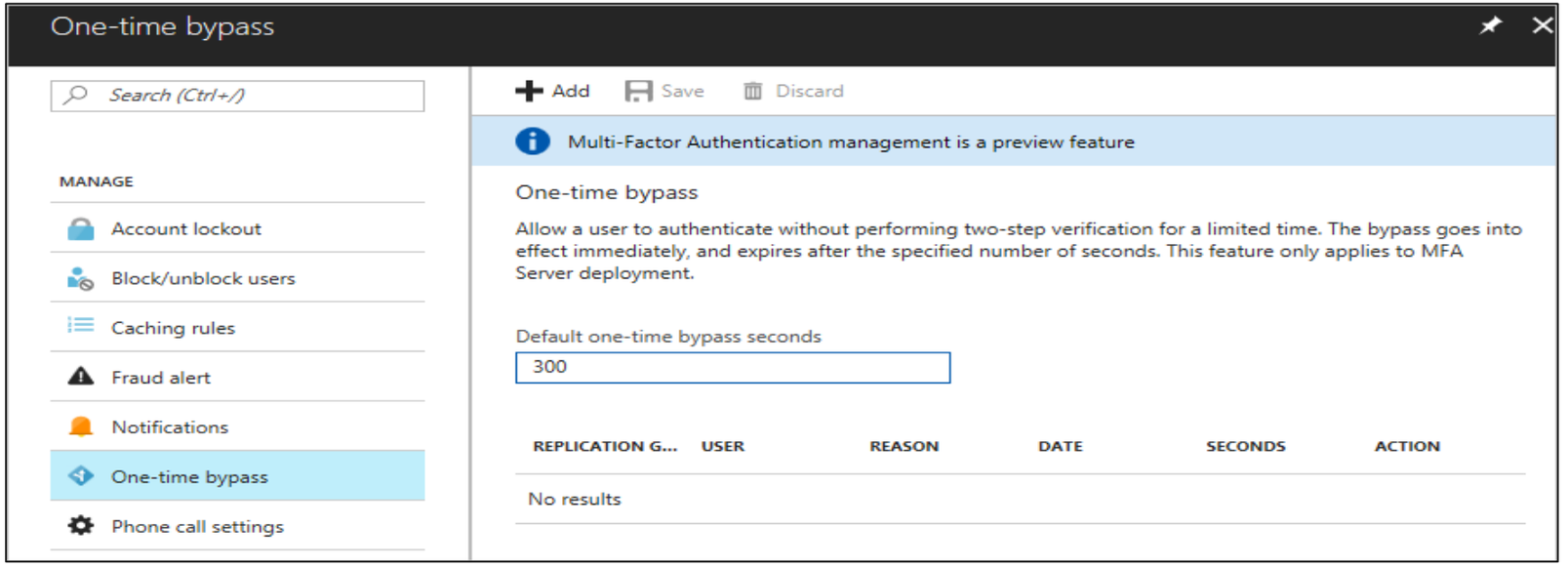


Trusted IPs

- Allows federated users or IP address ranges to bypass two-step authentication
- For managed tenants, you can specify IP ranges that can skip MFA
- For federated tenants, you can specify IP ranges and you can also exempt AD FS claims users



One-time Bypass



- 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.

Conditional Access

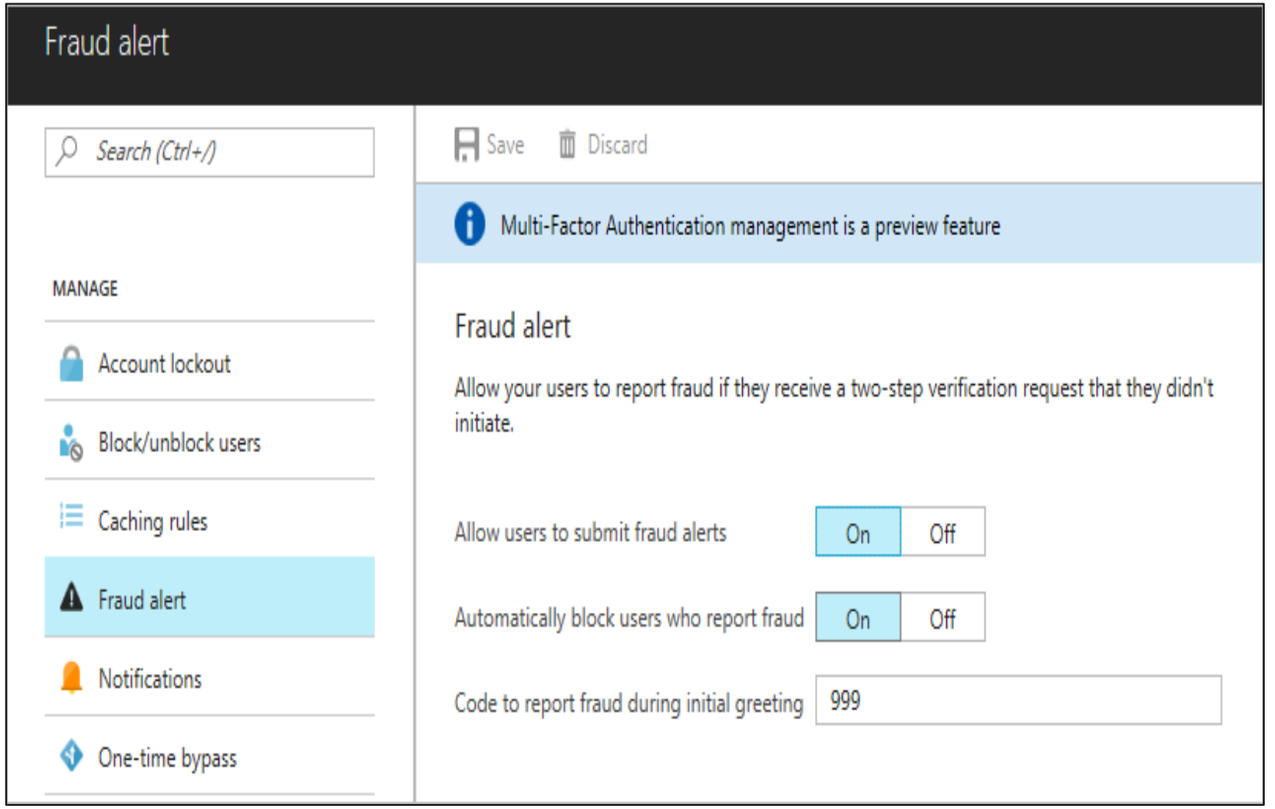
- Enables you to enforce controls on access to apps based on specific conditions
- The combination of your conditions with your access controls represents a conditional access policy

Conditions – “When this happens”

Access controls – “Then do this”

Fraud Alert

- Users can report fraudulent attempts to access their resources
- Report fraud attempts by using the mobile app or through their phone
- Block user when fraud is reported



Practice: Conditional Access

- Requirements
  - Access to an Azure AD Premium edition
  - A test account called Isabella Simonsen
- Tasks
  - Create the required conditional access policy
  - Evaluate a simulated sign in
  - Test the conditional access policy

Practice: MFA Authentication Pilot

- Enable Azure Multi-Factor Authentication
- Test Azure Multi-Factor Authentication

Review Questions

What are the mechanisms used by MFA two-step verification to authenticating users at sign-in? What is the cost of Azure MFA for global administrators?

Review Questions

What functionality does Trusted IPs provide? How do you select its different options?

Review Questions

What three questions should you consider to help you determine whether on-premises or cloud-based MFA is needed?

Questions?

Microsoft Azure

Homework Assignment

https://aka.ms/AZ300


Open
Mic …