



Microsoft Azure Administrator



Exam Ref AZ-104

Harshul Patel



Exam Ref AZ-104 Microsoft Azure Administrator

Harshul Patel

Exam Ref AZ-104 Microsoft Azure Administrator

Published with the authorization of Microsoft Corporation by:
Pearson Education, Inc.

Copyright © 2022 by Pearson Education, Inc.

All rights reserved. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, request forms, and the appropriate contacts within the Pearson Education Global Rights & Permissions Department, please visit www.pearson.com/permissions

No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-013-680538-0

ISBN-10: 0-136-80538-8

Library of Congress Control Number: 2021936223

ScoutAutomatedPrintCode

TRADEMARKS

Microsoft and the trademarks listed at <http://www.microsoft.com> on the “Trademarks” webpage are trademarks of the Microsoft group of companies. All other marks are property of their respective owners.

WARNING AND DISCLAIMER

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author, the publisher, and Microsoft Corporation shall have neither liability nor responsibility to any person or entity with respect to any loss

or damages arising from the information contained in this book or from the use of the programs accompanying it.

SPECIAL SALES

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

CREDITS

EDITOR-IN-CHIEF

Brett Bartow

EXECUTIVE EDITOR

Loretta Yates

SPONSORING EDITOR

Charvi Arora

DEVELOPMENT EDITOR

Rick Kughen

MANAGING EDITOR

Sandra Schroeder

PROJECT EDITOR

Tracey Croom

COPY EDITOR

Rick Kughen

INDEXER

Cheryl Ann Lenser

PROOFREADER

Donna E. Mulder

EDITORIAL ASSISTANT

Cindy Teeters

COMPOSITOR

codeMantra

COVER DESIGNER

Twist Creative, Seattle

Contents at a glance

Acknowledgments

About the Author

Introduction

CHAPTER 1 Manage Azure identities and governance

CHAPTER 2 Implement and manage storage

CHAPTER 3 Deploy and manage Azure compute resources

CHAPTER 4 Configure and manage virtual networking

CHAPTER 5 Monitor and back up Azure resources

Index

Contents

Introduction

Organization of this book

Preparing for the exam

Microsoft certifications

Quick access to online references

Errata, updates, & book support

Stay in touch

Chapter 1 Manage Azure identities and governance

Skill 1.1: Manage Azure Active Directory (Azure AD) objects

Create users and groups

Manage user and group properties

Manage device settings

Perform bulk user updates

Manage guest accounts

Configure Azure AD Join

Configure self-service password reset

Skill 1.2: Manage role-based access control (RBAC)

Role-based access control

Create a custom role

Interpret access assignments

Skill 1.3: Manage subscriptions and governance

Configure Azure policies

Configure resource locks

Apply and manage tags on resources

Create and manage resource groups

- Manage Azure Subscriptions
- Configure management groups
- Configure cost management
- Configure cost center quotas

Thought experiment

Thought experiment answers

Chapter summary

Chapter 2 Implement and manage storage

Skill 2.1: Secure Storage

- Configure network access to the storage accounts
- Create and configure storage accounts
- Generate shared access signatures
- Manage access keys
- Configure Azure AD Authentication for a storage account
- Configure access to Azure Files

Skill 2.2: Manage storage

- Create an export from an Azure job
- Create an import into an Azure job
- Install and use Azure Storage Explorer
- Copy data by using AzCopy
- Implement Azure Storage replication
- Configure blob object replication

Skill 2.3: Configure Azure Files and Azure Blob Storage

- Create an Azure Fileshare
- Create and configure Azure File Sync service
- Configure Azure Blob Storage
- Configure storage tiers for Azure blobs
- Configure blob Lifecycle Management

Thought experiment

Thought experiment answers

Chapter summary

Chapter 3 Deploy and manage Azure compute resources

Skill 3.1: Automate deployment of virtual machines (VMs) by using Azure Resource Manager templates

 ARM Template Overview

 Modify an Azure Resource Manager template

 Configure a virtual hard disk template

 Deploy from a template

 Save a deployment as an Azure Resource Manager template

 Deploy virtual machine extensions

Skill 3.2: Configure VMs for high availability and scalability

 Configure high availability

 Deploy and configure scale sets

Skill 3.3: Configure VMs

 Configure Azure Disk Encryption

 Move VMs from one resource group to another

 Manage VM sizes

 Add data disks

 Configure networking

 Redeploy VMs

Skill 3.4: Create and configure containers

 Configure sizing and scaling for Azure Container Instances

 Configure container groups for Azure Container Instances

 Configure storage for Azure Kubernetes Service (AKS)

 Configure scaling for AKS

 Configure network connections for AKS

 Upgrade an AKS cluster

Skill 3.5: Create and configure Azure App Service

- Create an App Service Plan
- Configure scaling settings in an App Service plan
- Create an App Service
- Secure an App Service
- Configure custom domain names
- Configure backup for an App Service
- Configure networking settings
- Configure deployment settings

Thought experiment

- Scenario 1
- Scenario 2

Thought experiment answers

- Scenario 1
- Scenario 2

Chapter summary

Chapter 4 Configure and manage virtual networking

Skill 4.1: Implement and manage virtual networking

- Create and configure a virtual networks and subnets
- Create and configure VNet peering
- Configure private and public IP addresses and network interfaces
- Create and configure public IP addresses
- Configure network routes
- Configure endpoints on subnets
- Configure private endpoints
- Configure Azure DNS
- Configure custom DNS settings
- Configure private DNS zones

Skill 4.2: Secure access to virtual networks

- Create security rules

- Associate NSG to a subnet or network interface
- Evaluate effective security rules
- Implement Azure Firewall
- Implement Azure Bastion Service

Skill 4.3: Configure load balancing

- Configure Azure Application Gateway
- Azure Load Balancer
- Configure an internal or public load balancer
- Troubleshoot load balancing

Skill 4.4: Monitor and troubleshoot virtual networking

- Monitor on-premises connectivity
- Use network resource monitoring
- Use Azure Network Watcher
- Troubleshoot external networking
- Troubleshoot virtual network connectivity

Skill 4.5: Integrate an on-premises network with an Azure virtual network

- Create and configure Azure VPN Gateway
- Create and configure Azure ExpressRoute
- Configure Azure Virtual WAN

Thought experiment

Thought experiment answers

Chapter summary

Chapter 5 Monitor and back up Azure resources

Skill 5.1: Monitor resources by using Azure Monitor

- Configure and interpret metrics
- Configure Azure Monitor logs
- Query and analyze logs
- Set up alerts and actions

Configure Application Insights

Skill 5.2: Implement backup and recovery

Create a Recovery Services Vault

Create and configure Backup Policy

Perform backup and restore operations by using Azure Backup Service

Perform site-to-site recovery by using Azure Site Recovery

Configure and review backup reports

Thought experiment

Thought experiment answers

Chapter summary

Index

Acknowledgments

I would like to acknowledge the flawless support I have received throughout the journey of book by Loretta and Charvi from the Pearson team. They have been very supportive and flexible, knowing the fact that I was dealing with multiple things at my end. I would also like to thank my wife, Divya, for her tremendous support in the making of this book. Despite her pregnancy, she played an instrumental role by encouraging and allowing me to complete the book on time. And last but not the least, the cuddle and cute smile of my little bundle of joy, Rivan, was a real energy booster during breaks in the middle of the night.

About the Author

HARSHUL PATEL is a technology enthusiast formerly from India who currently lives in Canada. He has been a cloud consultant with Microsoft Services for more than six years. He drives the adoption of Microsoft's cloud platforms for enterprise customers. He is thoroughly knowledgeable across various virtualization and cloud technologies. Harshul is an experienced author and an early adopter of many Microsoft products. He is a frequent speaker at various user group gatherings and a co-founder of a few global user groups.

Apart from work, Harshul is a happy-go-lucky guy. He loves to travel and spend time with his family and friends. Harshul and his wife, Divya, had a baby boy during the production of this book; they call him Rivan.

Introduction

The AZ-104 exam focuses on common tasks and concepts that an administrator needs to understand to deploy and manage infrastructure in Microsoft Azure. Manage Azure identities and Azure subscriptions is a key topic on the exam, which includes managing Azure AD objects (users, groups, and devices), use of Azure AD join and self-service password resets; it also covers role based access control, tagging, subscription level policies and resource organization using resource groups, subscription and management groups. Another topic covered is implement and manage storage, which includes creating and configuring storage accounts as well as configuring Azure files and understanding the services for importing and exporting data to Azure. A significant portion of the exam is focused on deploying and managing Azure compute resources, which includes configuring high availability of Azure VMs, creating and configuring virtual machine and their automated deployments as well as creating and configuring container solutions such as Azure Kubernetes Service (AKS) and Azure Container Instances (ACI); it also covers configuring web apps using app service and app service plans. This book also covers the creation and management of virtual networks, DNS, connectivity between virtual networks, configuring network security groups, Azure firewall and Azure bastion service; it also explains the load balancing solutions including configuration of application gateway. The final topic is monitor and backup Azure resources, which includes topics on how to monitor resources using Azure Monitor as well as how to implement back and recovery of Azure VMs including site to site recovery using Azure site recovery.

This book is geared toward Azure administrators who manage cloud services that span storage, security, networking and compute. It explains how to configure and deploy services across a broad range of related Azure services to help you prepare for the exam.

This book covers every major topic area found on the exam, but it does not cover every exam question. Only the Microsoft exam team has access to the exam questions, and Microsoft regularly adds new questions to the exam, making it impossible to cover specific questions. You should consider this book a supplement to your relevant real-world experience and other study materials. If

you encounter a topic in this book that you do not feel completely comfortable with, use the reference links provided throughout this book and take the time to research and study the topic. Great information is available on Microsoft Docs.

Organization of this book

This book is organized by the “Skills measured” list published for the exam. The “Skills measured” list is available for each exam on the Microsoft Learning website: <https://aka.ms/examlist>. Each chapter in this book corresponds to a major topic area in the list, and the technical tasks in each topic area determine a chapter’s organization. If an exam covers six major topic areas, for example, the book will contain six chapters.

Preparing for the exam

Microsoft certification exams are a great way to build your resume and let the world know about your level of expertise. Certification exams validate your on-the-job experience and product knowledge. Although there is no substitute for on-the-job experience, preparation through study and hands-on practice can help you prepare for the exam. This book is *not* designed to teach you new skills.

We recommend that you augment your exam preparation plan by using a combination of available study materials and courses. For example, you might use the Exam Ref and another study guide for your ”at home” preparation and take a Microsoft Official Curriculum course for the classroom experience. Choose the combination that you think works best for you. Learn more about available classroom training and find free online courses and live events at <http://microsoft.com/learn>. Microsoft Official Practice Tests are available for many exams at <http://aka.ms/practicetests>.

Note that this Exam Ref is based on publicly available information about the exam and the author’s experience. To safeguard the integrity of the exam, authors do not have access to the live exam.

Microsoft certifications

Microsoft certifications distinguish you by proving your command of a broad set of skills and experience with current Microsoft products and technologies. The exams and corresponding certifications are developed to validate your mastery

of critical competencies as you design and develop, or implement and support, solutions with Microsoft products and technologies both on-premises and in the cloud. Certification brings a variety of benefits to the individual and to employers and organizations.

More Info All Microsoft Certifications

For information about Microsoft certifications, including a full list of available certifications, go to <http://www.microsoft.com/learn>.

Quick access to online references

Throughout this book are addresses to webpages that the author has recommended you visit for more information. Some of these links can be very long and painstaking to type, so we've shortened them for you to make them easier to visit. We've also compiled them into a single list that readers of the print edition can refer to while they read.

Download the list at <MicrosoftPressStore.com/ExamRefAZ104/downloads>

The URLs are organized by chapter and heading. Every time you come across a URL in the book, find the hyperlink in the list to go directly to the webpage.

Errata, updates, & book support

We've made every effort to ensure the accuracy of this book and its companion content. You can access updates to this book—in the form of a list of submitted errata and their related corrections—at:

<MicrosoftPressStore.com/ExamRefAZ104/errata>

If you discover an error that is not already listed, please submit it to us at the same page.

For additional book support and information, please visit
<MicrosoftPressStore.com/Support>.

Please note that product support for Microsoft software and hardware is not offered through the previous addresses. For help with Microsoft software or hardware, go to <http://support.microsoft.com>.

Stay in touch

Let's keep the conversation going! We're on Twitter:
<http://twitter.com/MicrosoftPress>.

Chapter 1

Manage Azure identities and governance

Microsoft has long been a leader in the identity space. This leadership goes back to the introduction of Active Directory (AD) with Windows 2000 before the cloud even existed. Microsoft moved into cloud identity with the introduction of Azure Active Directory (Azure AD), which is now used by more than 5 million companies around the world. The adoption of Office 365 led to this extended use of Azure AD. These two technologies, however, have very different purposes, with AD primarily used on-premises and Azure AD primarily used for the cloud.

Microsoft has poured resources into making AD and Azure AD work together. The concept is to extend the identity that lives on-premises to the cloud by synchronizing the identities. This ability is provided by Azure AD Connect. Microsoft has also invested in extending those identities to enable scenarios such as single sign-on by using Active Directory Federation Services (ADFS), which is deployed in many large enterprises. (Note that Azure AD Connect is not covered on the AZ-104 exam.)

Microsoft has continued pushing forward by developing options for developers to leverage Azure AD for their applications. Microsoft provides the ability for developers to extend a company's Azure AD to users outside of the organization. The first option is known as Azure AD B2C (Business-to-Customer). This allows customers to sign in to applications using their social media accounts, such as a Facebook ID. A complementary technology—Azure AD B2B (Business to Business)—extends Azure AD to business partners.

This area of the AZ-104 exam is focused on the management of identities using Azure Active Directory.

In the latter part of this chapter, you will also learn how to manage role-based access control (formally known as RBAC) for Azure resources, including the following topics:

- Understand how RBAC works
- Create a custom role assignment
- Provide access to Azure resources using different roles
- Interpret access assignment
- Manage multiple directories

Finally, you will learn how to manage Azure subscriptions and other resources. This includes how to

- Configure Azure policies
- Configure resource locks
- Apply and manage tags on resources
- Manage resource groups
- Manage subscriptions
- Configure management groups
- Manage costs

By understanding the controls that are available in Azure for subscription and resource management, you enable your organization for success across your Azure estate.

Skills covered in this chapter:

- [Skill 1.1: Manage Azure Active Directory \(Azure AD\) objects](#)
- [Skill 1.2: Manage role-based access control \(RBAC\)](#)
- [Skill 1.3: Manage subscriptions and governance](#)

Skill 1.1: Manage Azure Active Directory (Azure AD) objects

In an Azure AD tenant, there are users, groups, and devices that are controlled through the features of Azure AD discussed in this section. In this section, we focus on managing users and groups throughout their lifecycles, how to manage device settings, how to perform bulk updates to users using automation tooling such as PowerShell, and how to manage guest accounts.

In the latter part of this section, we will also discuss how to manage devices with Azure AD Join and how to configure user experience controls, such as self-service password reset (SSPR).

This skill covers:

- [Create users and groups](#)
- [Manage user and group properties](#)
- [Manage device settings](#)
- [Perform bulk user updates](#)
- [Manage guest accounts](#)
- [Configure Azure AD Join](#)
- [Configure self-service password reset](#)

Create users and groups

There are primarily two types of users in Azure AD – cloud-only users and users synchronized from an on-premises directory. Cloud-only users are created and managed exclusively in Azure AD, and their attributes can be updated directly in Azure AD.

You can create cloud-only users through the Azure portal, Azure PowerShell, and the Azure command-line interface (CLI). When creating new users, you must be assigned to the Global Administrator or User Administrator role. See [Skill 1.2](#) for more details about various roles and their assignments.

To create users from the Azure portal, search for **Azure Active Directory** in the search pane, or you can browse to all Azure services and select **Azure Active Directory** as a user with rights to create users, click **Users** to access the **Users** blade, and click **+New User**. An example of this blade is shown in [Figure 1-1](#). Note that you can also invite users (guest users) to your directory through the Azure portal.

When creating a new user, only the username (sign in address for Azure AD) and the user's name (given name and surname) are mandatory. You can configure additional settings, such as assigning specific groups and roles, blocking sign-ins from a specific location, and so on.

More Info

For more information, see:

<https://docs.microsoft.com/azure/governance/blueprints/tutorials/protect-new-resources>.

Groups are groups of objects that make role assignments and access permissions easier to manage. A group can contain groups, users, devices, or service principals. When using groups, you eliminate the need to individually assign roles or permissions. Creating groups is a similar experience and can be performed from the Azure portal, Azure PowerShell, the Azure CLI, and Microsoft Graph. To create a group in the Azure portal, search for **Azure Active Directory** or browse to all Azure services, select **Azure Active Directory**, click **Groups** to access the **New Group** blade, and click **+New Group**. The New Group blade is shown in [Figure 1-2](#).

When creating a new group, there are several factors that dictate the type of group that is created and how that group behaves in Azure AD and associated workloads, such as Office 365.

More Info

In 2020, Office 365 was renamed as Microsoft 365. The Azure portal UI still shows the old terminology, so it has been referred to as “Office 365” throughout this chapter. You can find details on how Microsoft 365 is integrated with Azure at <https://docs.microsoft.com/microsoft-365/enterprise/azure-integration>.

New user

Got feedback?



Create user

Create a new user in your organization.
This user will have a user name like
alice@msp131499.onmicrosoft.com.
[I want to create users in bulk](#)



Invite user

Invite a new guest user to collaborate with
your organization. The user will be emailed
an invitation they can accept in order to
begin collaborating.

[I want to invite guest users in bulk](#)

[Help me decide](#)

Identity

User name * ⓘ

Example: chris @ [REDACTED]
The domain name I need isn't shown here

Name * ⓘ

Example: 'Chris Green'

First name

Last name

Groups and roles

Groups

0 groups selected

Roles

User

Settings

Block sign in

Yes No

Usage location

[REDACTED]

Job info

Job title

Department

[Create](#)

FIGURE 1-1 New user blade in the Azure portal

First, you must select the type of group you are creating. You have two options: **Security** and **Office 365**. Security groups allow you to share Azure resources access to a group of users , devices, or service principals. An Office 365 group allows access to a shared mailbox, calendar, SharePoint site, and so on. Note that even if you are creating groups in an Azure AD tenant that is not associated with an Office 365 subscription, you will still see the option to create an Office 365 group.

New Group

Group type *

Security



Group name * ⓘ

Enter the name of the group

Group description ⓘ

Enter a description for the group

Membership type * ⓘ

Assigned



Owners

No owners selected

Members

No members selected

FIGURE 1-2 New Group blade in the Azure portal

Also, **Group Name** is a required field. While filling in a **Group Description** is not required, it is recommended that you always include a Group Description to make it easier to find and identify the purpose of a group later.

The **Membership Type** drop-down menu allows you to select from one of three values:

- **Assigned.** This value allows you to select one or more users and add them to the group. Adding and removing users is performed manually.
- **Dynamic User.** This value allows you to use dynamic group rules to automatically add and remove members.
- **Dynamic Device.** This value allows you use dynamic group rules to automatically add and remove devices.

Important Dynamic group requirement

You can only create a dynamic group if you have a Premium AD license. Otherwise, the **Membership Type** option is unavailable and is set to **Assigned**.

For both dynamic user and dynamic device-based groups, the rules associated with the group are evaluated on an ongoing basis. If a user or device has an attribute that matches the rule, that user or device is added to the group. If an attribute changes and the user or device no longer matches the criteria for group membership, the entity will be removed. Membership processing is not immediate. If an error occurs while processing a membership rule, an error is surfaced on the Group page in the Azure portal. You can always view the current processing status from the Group page.

It is important to note that you can create a dynamic group for users or devices, but you cannot create both at the same time. You also cannot use user attributes in a device-based rule. It is possible to change the membership type of a group after it has been created, which provides an opportunity to transition from a static (or assigned) membership model to a dynamic membership model or vice-versa.

When creating dynamic groups, rules can be edited in the simple rule format,

where you will build the query and conditions in the rule builder, where you can build complex rules with conditional logic. In the example shown in [Figure 1-3](#), a dynamic user group is being created, which will automatically update its membership based on the department attribute and its value in Azure AD.

Dynamic membership rules X

Save Discard | Got feedback?

[Configure Rules](#) [Validate Rules \(Preview\)](#)

You can use the rule builder or rule syntax text box to create or edit a dynamic membership rule. (Learn more)

And/Or	Property	Operator	Value	Edit
	department	Equals	Finance	

+ Add expression + Get custom extension properties (Learn more)

Rule syntax / Edit

```
(user.department -eq "Finance")
```

FIGURE 1-3 Dynamic membership rules

Dynamic groups require an Azure AD Premium P1 (or equivalent EM + S)

license.

Manage user and group properties

As users and groups are used, they might need updates to their attributes (or properties). For example, you might need to change a users' job title, or you might need to add or remove members from an existing group.

Users and groups can be updated using management tools such as Azure portal, Azure PowerShell, Azure CLI, and Microsoft Graph. [Figure 1-4](#) shows an example of the user profile in the Azure portal that can be accessed by browsing to your Azure AD tenant, selecting **Users**, choosing a user, and clicking **Edit**.

Groups can be managed through the Azure portal by browsing to your Azure AD tenant, selecting **Groups**, choosing a specific group, and then clicking **Properties**, **Members**, or **Owners**, depending on the type of update you want to make. When editing a group, you will not be able to change the Group Type (such as changing a Security group to an Office 365 group), but you will be able to update the Group Name, Group Description, and the Membership Type, as shown in [Figure 1-5](#). Changing a static group to dynamic group will remove all the members from the static group and apply dynamic membership rules. This change will also affect the access to the resources if the static group has any previously assigned access for its members.

Chris Green | Profile

User

[View](#) [Save](#) [Discard](#) | [Got feedback?](#)

[Diagnose and solve problems](#)

Manage

- [Profile](#)
- [Assigned roles](#)
- [Administrative units \(Preview\)](#)
- [Groups \(Preview\)](#)
- [Applications](#)
- [Licenses](#)
- [Devices](#)
- [Azure role assignments](#)
- [Authentication methods](#)

Activity

- [Sign-ins](#)
- [Audit logs](#)

Troubleshooting + Support

- [New support request](#)

Chris Green

chris@msp131499.onmicrosoft.com



User Sign-ins Group memberships 0

Select a file Select a thumbnail image (max size 100KB)

Mar 22 Mar 29 Apr 5 Apr 12 Apr 19

Identity

Name	First name	Last name
Chris Green	Chris	Green
User name	User type	
chris@msp131499.onmicrosoft.com	Member	
Object ID	Source	
89d84f1e-3a7c-4e7c-9a5d-7033c66a108c	Azure Active Directory	

Job info

Job title	Department	Manager	Remove Change
Director	Finance		

Settings

Block sign in	Usage location
Yes No	Canada

Contact info

Street address	State or province	Country or region	Office
City	ZIP or postal code	Office phone	Mobile phone

Authentication contact info

Use the [Authentication methods](#) page to manage authentication contact info for a user

Minors and consent

Age group	Consent provided for minor	Legal age group classification
Undefined	None	Undefined

FIGURE 1-4 A user profile in the Azure portal

 Business Development | Properties

Group

«  Save  Discard

 Overview (Preview)

 Diagnose and solve problems

Manage

 Properties

 Members (Preview)

 Owners (Preview)

 Administrative units (Preview)

 Group memberships (Preview)

 Applications

 Licenses

 Azure role assignments

General settings

Group name * ⓘ

Business Development 

Group description ⓘ

Welcome to Biz Dev Team 

Group type

Office 

Membership type * ⓘ

Assigned 

Object Id

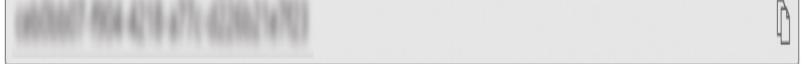


FIGURE 1-5 Edit group properties blade in the Azure portal

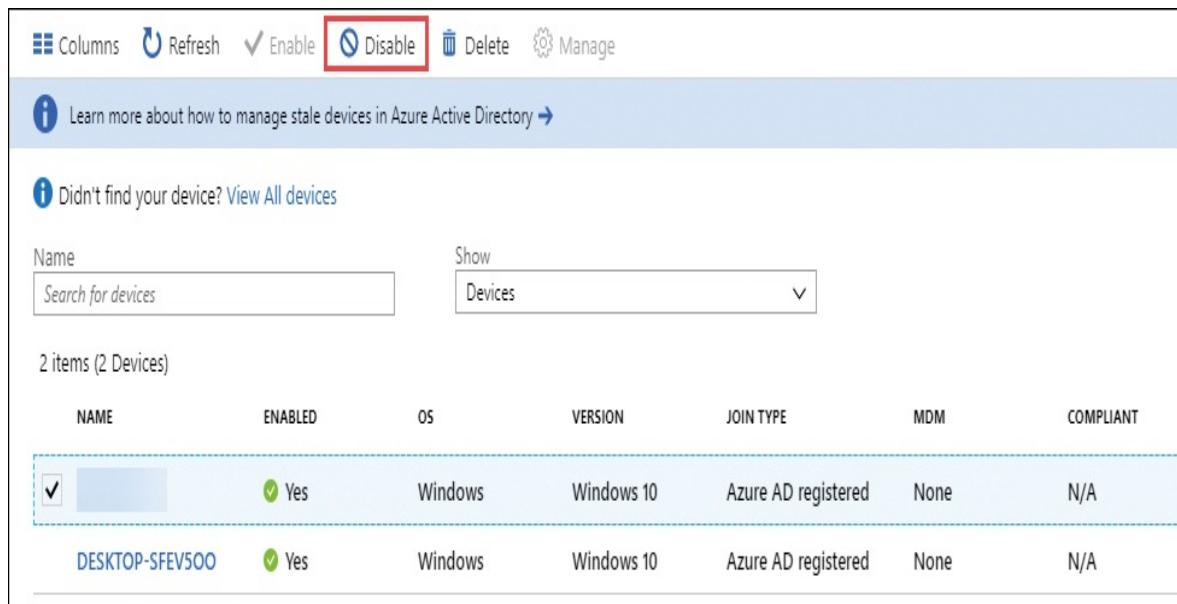
Manage device settings

Registered and joined devices in Azure AD can be managed in two areas in the Azure portal.

- The first is by browsing to your Azure AD tenant in the Azure portal and selecting **Devices**. **All Devices** is the default view, but you can also choose other views, such as **Device Settings**, **BitLocker Keys**, and so on).
- The second is through the **Devices** blade for an individual user.

With either option, you will be able to search for devices using the device name as a filter, view a detailed overview of any registered and joined devices, and perform common device-management tasks.

To enable and disable devices, you must be a Global Administrator. Disabling a device prevents it from accessing Azure AD resources. Note that this does not prevent the user from accessing resources in general; it only prevents the user from accessing resources from that disabled device. [Figure 1-6](#) shows the **Disable** button.



The screenshot shows the 'All Devices' blade in the Azure portal. At the top, there are buttons for 'Columns', 'Refresh', 'Enable' (with a checkmark), 'Disable' (highlighted with a red box), 'Delete', and 'Manage'. Below this is a blue banner with an info icon and text about managing stale devices. A message says ' Didn't find your device? [View All devices](#)'. There are search fields for 'Name' and 'Show' (set to 'Devices'). The main area shows a table with 2 items (2 Devices). The columns are NAME, ENABLED, OS, VERSION, JOIN TYPE, MDM, and COMPLIANT. Two rows are listed:

NAME	ENABLED	OS	VERSION	JOIN TYPE	MDM	COMPLIANT
DESKTOP-SFEV500	Yes	Windows	Windows 10	Azure AD registered	None	N/A
DESKTOP-SFEV500	Yes	Windows	Windows 10	Azure AD registered	None	N/A

FIGURE 1-6 Disable button from the All Devices blade in the Azure portal

Deleting devices is similar to enabling or disabling a device. Again, the user performing the update must be a Global Administrator. Deleting a device prevents a device from accessing your Azure AD resources and removes all details that are attached to the device (including BitLocker keys for Windows devices). Deleting a device represents a non-recoverable activity and is not recommended unless it is required for an activity such as device decommissioning.

Perform bulk user updates

Previously, the Azure portal was only helpful for single updates to users, which meant we had to rely on custom automation solutions (mostly using PowerShell) for updating users in bulk. Because of recent updates, you can now perform bulk operations (such as creating, inviting, and deleting users in batches) using the Azure portal.

You can access this functionality by navigating to your Azure AD tenant in the Azure portal and then clicking **Users**. You will see these functionalities at the top of the blade, as shown in [Figure 1-7](#).



FIGURE 1-7 Bulk updates from the Users blade in the Azure portal

After clicking the **Bulk Create** button, you will see the **Bulk Create User** blade, which is shown in [Figure 1-8](#).

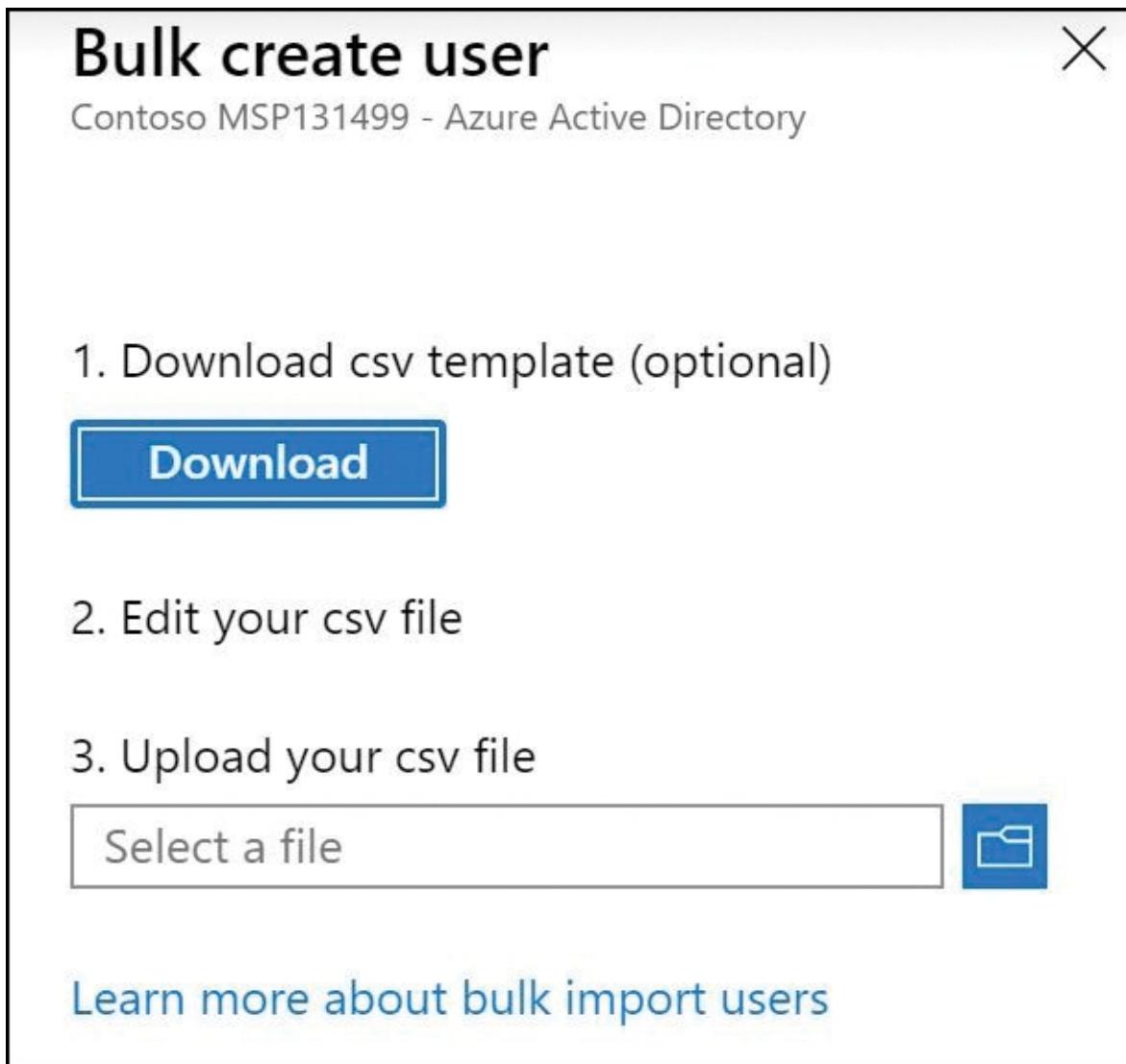


FIGURE 1-8 Bulk create user blade in the Azure portal

Bulk operations are three-step process:

1. Download a CSV (comma-separated values or comma-delimited) template (`UserCreateTemplate.csv`) by clicking the **Download** button on the **Bulk Create User** blade. This is a standard template with mandatory attributes, such as **Name**, **User Name**, **Initial Password**, and **Block Sign In**. You can also specify optional attributes such as **First Name**, **Last Name**, **Job Title**, and so on.
2. Edit the CSV file with bulk update values. You just need to update appropriate values and save the changes. The sample mandatory values are

already included in the template for reference.

3. Upload the updated CSV file and submit the operation.

After submitting the operation, you can check the status of the bulk operation by navigating to **Bulk Operation Results** option under the **Activity** section of the **Users** blade (see [Figure 1-9](#)).

The screenshot shows the 'Bulk operation results' page within the Azure Active Directory 'Users' blade. The left sidebar lists navigation options: All users (Preview), Deleted users, Password reset, User settings, Diagnose and solve problems, Activity (which is selected and highlighted in grey), Sign-ins, Audit logs, Bulk operation results (which is also highlighted in grey), Troubleshooting + Support, and New support request. The main content area displays a table of bulk operation results. The table has columns: File name, Upload time, Completion time, Status, # Success, # Failure, and Total requests. There are two entries:

File name	Upload time	Completion time	Status	# Success	# Failure	Total requests
UserCreateTemplate.csv	4/19/2020, 7:02:38 PM	4/19/2020, 7:02:41 PM	Completed with errors	0	2	2
UserCreateTemplate.csv	4/19/2020, 7:04:36 PM	4/19/2020, 7:04:38 PM	Completed with no errors	2	0	2

FIGURE 1-9 Bulk operation results blade in the Azure portal

Manage guest accounts

To create guest users from the Azure portal, browse to your Azure AD tenant as a user with rights to create users and select the **Users** blade and choose **+New Guest User**. An example of this blade is shown in [Figure 1-10](#). A guest user can be anyone who is invited to collaborate with your organization. Once created, the guest user should receive an invitation in their mailbox.

Guest users can be created and managed similarly to how normal user accounts are created and managed. Guest users can be invited to the directory, group, or application. As soon as we invite the guest user, that account is created in Azure AD with the User Type set to Guest. The guest user will receive an email invitation immediately after creation. The guest user must accept the invitation along with the first-time consent process in order to access the assigned resources.

By default, all users and admins can invite guests. You can restrict the way guest users can be invited by accessing the **Manage External Collaboration Settings** on the **Users** blade under **User Settings**. The External Collaboration Settings are shown in [Figure 1-11](#). You can also access these settings from Azure AD tenant by clicking the **User Settings** on the left menu and then choosing **Manage External Collaboration Settings** in the **External Users** section.

New user
Contoso MSP131499

Got feedback?

Create user

Create a new user in your organization.
This user will have a user name like
alice@msp131499.onmicrosoft.com
[I want to create users in bulk](#)

Invite user

Invite a new guest user to collaborate with
your organization. The user will be emailed
an invitation they can accept in order to
begin collaborating.
[I want to invite guest users in bulk](#)

[Help me decide](#)

Identity

Name <small>(i)</small>	<input type="text" value="Example: 'Chris Green'"/>
Email address <small>*</small> <small>(i)</small>	<input type="text" value="Example: chris@contoso.com"/>
First name	<input type="text"/>
Last name	<input type="text"/>

Personal message

Groups and roles

Groups	0 groups selected
Roles	User

Settings

Block sign in	<input type="button" value="Yes"/> <input checked="" type="button" value="No"/>
Usage location	<input type="text"/>

Job info

Job title	<input type="text"/>
Department	<input type="text"/>

FIGURE 1-10 New User blade in the Azure portal

External collaboration settings

 Save  Discard

Guest users permissions are limited 

 Yes  No

Admins and users in the guest inviter role can invite 

 Yes  No

Members can invite 

 Yes  No

Guests can invite 

 Yes  No

Enable Email One-Time Passcode for guests (Preview) 

[Learn more](#)

 Yes  No

Collaboration restrictions

- Allow invitations to be sent to any domain (most inclusive)
- Deny invitations to the specified domains
- Allow invitations only to the specified domains (most restrictive)

FIGURE 1-11 External Collaboration Settings blade in the Azure portal

When a guest user is added, the guest user has a Consent Status (viewable in PowerShell) as `PendingAcceptance`. This value will be changed to **Accepted** immediately after the guest user accepts the invitation. The guest user will appear as an “invited user” in the Azure portal until the user accepts the invitation.

Configure Azure AD Join

Azure Active Directory includes the ability to manage device identity, which enables single sign-in to devices and the applications and services managed through Azure Active Directory that are accessed from that device. Managed devices include both enterprise and bring-your-own-device (BYOD) scenarios. This allows users to work from any device, including personal devices, all while protecting corporate intellectual property with the necessary regulatory and compliance controls.

Azure AD Join allows you to control these devices, the applications installed and accessed from them, and how those applications interact with your corporate data.

When associating devices with Azure AD, you have three options: registering a device, joining a device, and using hybrid AD joined. Registration of devices would be appropriate for personal devices, while joining devices is useful for corporate-owned devices. Hybrid AD joined devices are joined to your on-premises Active Directory and are registered with your Azure AD tenant.

Associating a device with Azure AD allows you to manage a device’s identity by implementing features like single sign on (SSO) and securing access using conditional access. Note that this identity can be managed independently of a user’s identity. This provides a great degree of flexibility because devices can be enabled or disabled without affecting a user account. Azure AD Join is an extension of device registration that changes the local state of the device. When a device is Azure AD-Joined, users can sign in to the device using an organizational account instead of a personal account.

Also, registration of devices in Azure AD can be combined with a mobile

device management solution, such as Microsoft Intune, Microsoft Endpoint Configuration Manager, Mobile Application Management (MAM), and Group Policy if it is hybrid Azure AD Joined. This allows for additional device attributes—such as device OS version and device state (including whether the device is rooted or jailbroken)—to be tracked in Azure AD. Those attributes can then be used to build and enforce conditional access policies, which can further secure corporate data.

Device registration is configured in Azure AD under **Devices > Device Settings**. From this screen, you can set the configuration for an entire Azure AD tenant, as seen in [Figure 1-12](#).

 Devices | Device settings
Contoso MSP131499 - Azure Active Directory

« Save X Discard | Got feedback?

All devices

Device settings

Enterprise State Roaming

Diagnose and solve problems

Activity

Audit logs

Troubleshooting + Support

New support request

Users may join devices to Azure AD ⓘ

All Selected None

Selected >
No member selected

Additional local administrators on Azure AD joined devices ⓘ

Selected None

Selected >
No member selected

Users may register their devices with Azure AD ⓘ

All None

 Learn more on how this setting works

Require Multi-Factor Auth to join devices ⓘ

Yes No

Maximum number of devices per user ⓘ

20 ▾

Enterprise State Roaming

Manage Enterprise State Roaming settings

FIGURE 1-12 Configure device registration settings

From this screen, you can configure the following settings:

- **Users May Join Devices To Azure AD.** This setting allows you to select the users and groups that can join devices to Azure AD. This setting only applies to Azure AD Join on Windows 10 devices. The default value is **All** and can be changed to **Selected** or **None**.
- **Additional Local Administrators On Azure AD Joined Devices.** With Azure AD Premium or with the Enterprise Mobility Suite, you can choose which users are granted Local Administrator rights to the device. Global Administrators and the device owner are granted Local Administrator rights by default. The default value is **None** and can be changed to **Selected**. If the value is set to **Selected**, any users added here are also added to the Device Administrators role in Azure AD.
- **Users May Register Their Devices with Azure AD.** Allow users to register their devices with Azure AD (Workplace Join). Enrollment with Microsoft Intune or Mobile Device Management for Office 365 requires Device Registration. If you have configured either of these services, **ALL** will be selected, and the button associated with the setting will be disabled.
- **Require Multi-Factor Auth To Join Devices.** Multifactor authentication is recommended when adding devices to Azure AD. When set to **Yes**, users who are adding devices from the Internet must first use a second method of authentication. Prior to enabling this setting, you must ensure that multifactor authentication is configured for the users who are able to register devices and that those users have gone through MFA set up.
- **Maximum Number Of Devices Per User.** This setting designates the maximum number of devices that an individual user can have in Azure AD. If the quota is reached, the user will not be able to add a device until one of their existing devices is removed. Valid values for this setting are **5, 10, 20, 50, 100**, and **Unlimited**.

Note Hybrid AD Joined Devices

The Multi factor authentication and maximum number of devices per user settings are not applicable to hybrid AD joined devices.

- **Manage Enterprise State Roaming Settings.** Clicking this link at the bottom of the blade takes you to new blade where you will see the **Users May Sync Settings And App Data Across Devices** setting. With Azure AD Premium or EMS, you can select a subset of your users by choosing the **Selected** value and enabling the **Enterprise State Roaming** feature, which can be enabled or for **All** users or **None**. This is only applicable for Windows 10 devices.

After the directory has been configured, you can begin registering devices. For Azure AD Join, there are several requirements for devices, including Windows versions. The requirements for Windows versions are driven by the type of Azure AD Join: hybrid or non-hybrid. Non-hybrid Azure AD Join is applicable to devices that are not joined to an on-premises Active Directory, while hybrid Azure AD Join is applicable to devices that are joined to an on-premises directory. For hybrid Azure AD Join, an IT administration must perform the join to Azure AD.

For non-hybrid Azure AD Join, Windows 10 Professional and Windows 10 Enterprise devices can be joined to a directory. For hybrid Azure AD Join scenarios, you can join current Windows devices, such as Windows 10 and Windows Server 2016. Also, there is support for a hybrid join with down-level devices, including Windows 7, Windows 8.1, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2.

Configure self-service password reset

The password reset is one of the highest cost-incurring activities for many organizations, and many organizations have dedicated front-line help desks to handle such requests. Self-service password reset (SSPR) allows users to reset their own passwords in Azure AD, including the ability to optionally write the password back to an on-premises environment when properly licensed and configured by using password writeback and Azure AD Connect. SSPR allows users to change their passwords, reset their passwords when they cannot sign in, and unlock their accounts, all without the intervention of an IT department.

Each scenario above addresses both cloud-only and hybrid users. Also, licensing requirements vary. [Table 1-1](#) details each scenario, the type of user it applies to, and any required licenses.

TABLE 1-1 Self-service password reset license requirements

Scenario	User Type	License Requirements
Password Change	Cloud-only user.	Included in all editions of Azure AD
Password Reset	Cloud-only user	Microsoft 365 Business Standard, Microsoft 365 Business Premium, Azure AD Premium P1, Azure AD Premium P2
Password Change/Unlock/Reset	Hybrid user	Microsoft 365 Business Premium, Azure AD Premium P1, Azure AD Premium P2

SSPR can be enabled through the Azure portal by browsing to your Azure AD tenant and then selecting **Password Reset**. When enabling SSPR, you can scope the functionality to a group, which will allow you to roll out the feature in waves as users are onboarded into the service. As a part of configuration, you will also select the **Authentication Methods** for SSPR: **Mobile App Notification, Mobile App Code, Email, Mobile Phone, Office Phone, and/or Security Questions** (as shown in [Figure 1-13](#)). Finally, you will configure registration options using the **Registration** blade, such as whether registration is required to use SSPR and the number of days for reconfirmation.

Additionally, you can also control how notifications are triggered to users and admins using the **Notifications** blade. There is an option available to provide customize helpdesk link to notify the administrator directly, which can be configured using the **Customization** blade. If on-premises integration is enabled, you can also control write back passwords to your on-premises directory and allow users to unlock accounts without resetting their passwords using the **On-Premises Integration** blade.

More Info

You can find details on self-service password reset writeback using following link: <https://docs.microsoft.com/azure/active-directory/authentication/howto-sspr-writeback>

directory/authentication/concept-sspr-writeback.

Password reset | Authentication methods

Contoso MSP131499 - Azure Active Directory

« Save Discard

Diagnose and solve problems

Manage

Properties

Authentication methods

Registration

Notifications

Customization

On-premises integration

Activity

Audit logs

Usage & insights

Troubleshooting + Support

New support request

Number of methods required to reset

1

2

Methods available to users

Mobile app notification

Mobile app code

Email

Mobile phone

Office phone

Security questions

These settings only apply to end users in your organization. Admins are always enabled for self-service password reset and are required to use two authentication methods to reset their password. Click here to learn more about administrator password policies.

FIGURE 1-13 Configure SSPR Authentication Methods

Skill 1.2: Manage role-based access control (RBAC)

Access control in Microsoft Azure is an important part of an organization's security and compliance requirements. Implementing role-based access control (RBAC) allows Azure to define access rights at a very granular level, based on each user's assigned tasks or the day-to-day activities those users need to perform in their roles. This ensures that each person can perform the task he or she needs to accomplish.

This skill covers:

- [Understand how RBAC works](#)
- [Create a custom role](#)
- Provide access to Azure resources by assigning roles
 - Subscriptions
 - Resource groups
 - Resources (VM, disk, and so on)
- Implement RBAC using Azure portal
- Implement RBAC using Azure PowerShell and the Azure CLI
- [Interpret access assignments](#)

Role-based access control

Role-based access control (RBAC) allows you to manage the entities, also referred to as "security principals," that have access to Azure resources and the actions that those entities can perform. In addition to determining who can do what, Azure RBAC is also applied at a scope that dictates the areas they have access to. In Azure, access can be granted to users, groups, service principals, and managed identities through role assignments, which are then applied at a

scope, such as a subscription, a resource group, or even an individual resource. Azure RBAC is applicable to the management of resources created in the Azure Resource Manager (ARM) deployment model.

A role is the definition of what actions are allowed and/or denied. RBAC is configured by selecting a role and associating the role with a security principal, such as a user, group, or service principal. Then, this combination of role and security principal is applied to a scope of a subscription, a resource group, or a specific resource through a role assignment.

In Azure, there is also role inheritance where child resources inherit the role assignments of any parents. For example, if a user is granted read access to a subscription, that user will have read access to all the resource groups and resources in that subscription. If a managed identity (identity which supports Azure AD authentication) is granted contributor rights for a single resource group, that security principal can only interact with that resource group and its child resources, but it cannot create new resource groups or access resources in other resource groups unless an explicit role assignment is made.

Azure RBAC uses the additive model. As you begin to apply roles to security principals in Azure, it is not uncommon to have overlapping assignments where a security principal is assigned a different role assignment at both a parent and a child scope. For example, if a user is granted Contribute rights at the management group scope and then is granted Reader rights in a subscription, the user will still have Contribute rights across the subscription along with Contribute rights to any other subscriptions under the management group. Another way to think of this is that the most privileged access right takes precedence.

Before a security principal such as a user or group can interact with Azure resources, they must be granted access at a scope through a role assignment. Once a security principal has been granted access, it can perform any action that it has rights to perform. It is always recommended to provide the minimum privileges to an object or user to perform actions as needed. [Figure 1-14](#) shows a suggested access pattern that adheres to the principles of least privilege. In this example, a security group in Azure AD, called `IT Audit`, is granted Read access rights at the subscription scope, granting member of the group Read access to all resource groups and resources in the subscription. A security group called `Application Admins` is granted Contributor access rights to only selected resource groups. Another security group called `Application Owners` is granted Owner access rights to selected resource groups as well. By using multiple

security groups and role assignments at the proper scope, access can be granted in the future just by updating the security group membership in Azure AD.

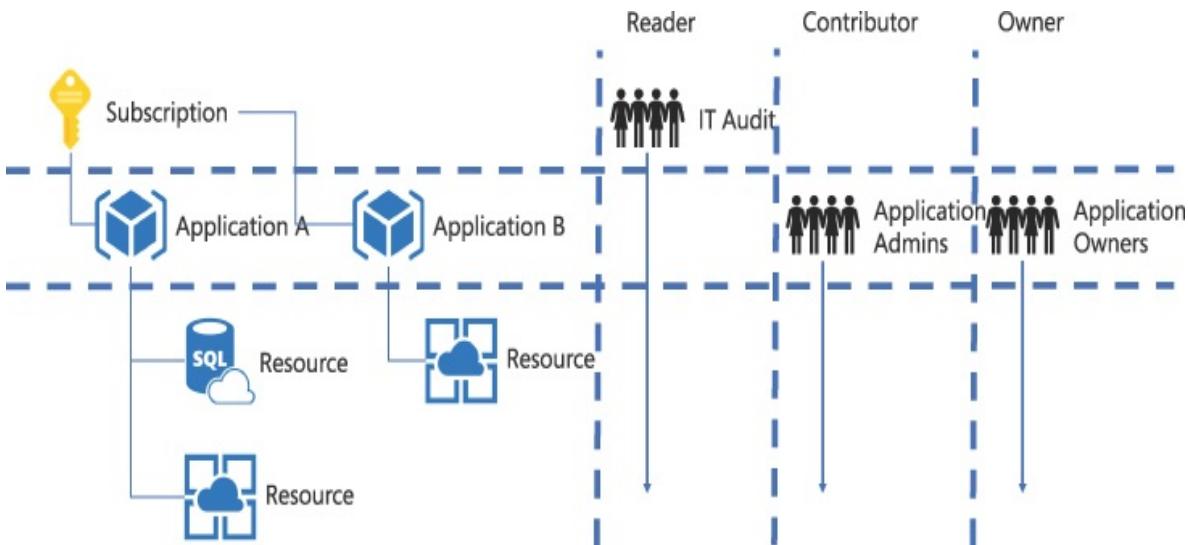


FIGURE 1-14 Azure RBAC role assignments

Important Using Groups with Azure RBAC

RBAC role definitions get “attached” to a user, group, service principal, or managed identity via a role assignment. When assigning roles to a group, all users in the group will inherit the assigned role. You can assign roles to group for easier management and greater flexibility when applying RBAC at scale.

The specific permissions that are applied to a resource with RBAC are defined in a role definition. A role definition contains the list of permissions—or declared permissions—and those permissions define what actions can or cannot be performed against a type of resource, such as read, write, or delete.

Role definitions, or roles, can be either built-in or custom. There are a number of built-in role definitions in Azure. Some of the built-in roles such as the Owner role, which includes permissions that manage resources, security, and the application of role assignments. Also, there are built-in roles with limited permission sets, such as a Storage Blob Data Reader, which allows the assigned security principal to only read and list containers and blobs.

There are many built-in roles in Azure, which can be found at <https://docs.microsoft.com/azure/role-based-access-control/built-in-roles>. Microsoft consistently adds new built-in roles as services evolve or as new services are introduced.

Important Azure Roles and Azure AD Roles

RBAC roles are different from the Azure AD administrative. RBAC roles are used to manage access and allow or restrict users to Azure resources, while Azure AD administrative roles are used to allow or restrict admins to perform identity tasks, such as creating new users, resetting the users' passwords, and so on. For example, a user who is granted Global Administrator rights in Azure AD does not have permissions to create resources in Azure, but he or she can perform all the identity tasks for an Azure AD tenant.

The access rights are controlled with a logical boundary known as scope. For example, to grant a group Contribute rights to all the resources in a resource group, the Contributor role can be assigned to the group at the resource group scope where it is then inherited by all of the resources in the resource group.

There are four scopes at which RBAC can be applied, and scopes are structured in a parent-child relationship where RBAC is inherited by any child scopes. The highest scope, or top-most parent scope, is a management group.



Exam Tip

Management groups are not applicable in all scenarios and in some cases a subscription will be the highest scope you will work with when applying role assignments.

Under the management group are subscriptions; under subscriptions are resource groups; and under resource groups are resources. [Figure 1-15](#) shows a

sample hierarchy with a parent management group and two subscriptions, each with a resource group and child resources. Note that you can also create another management group under a root management group. An Azure AD tenant can support up to 10,000 management groups.

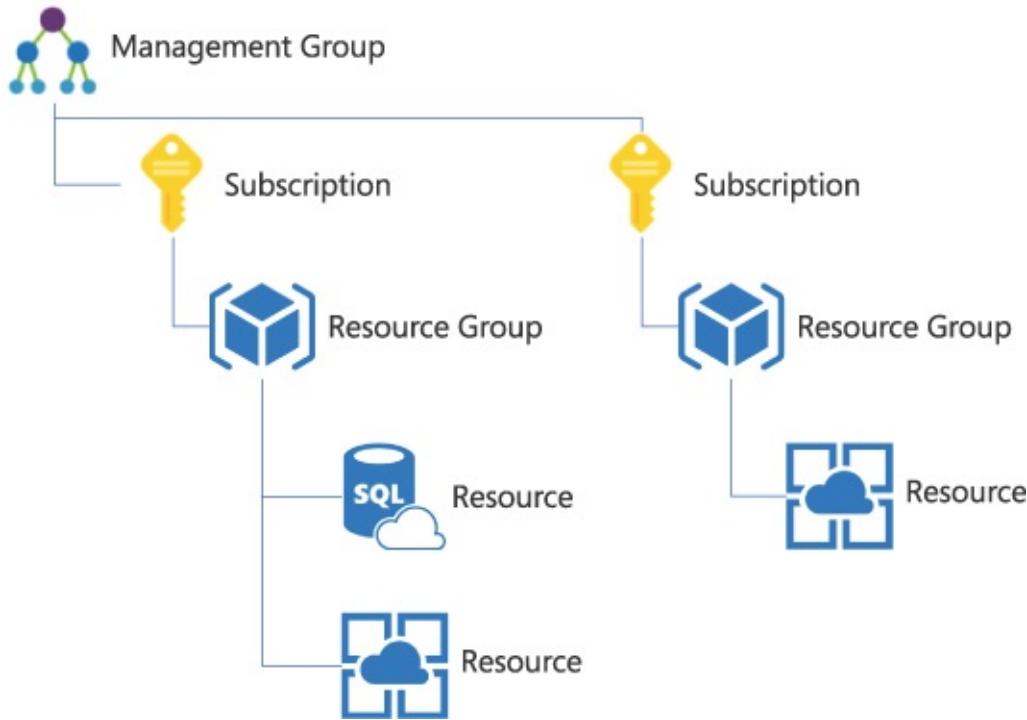


FIGURE 1-15 Scope Hierarchy

Important RBAC Inheritance

The concept of RBAC inheritance is critical. Granting a user access to the Owner role at the management group scope will grant that user Owner rights to all the subscriptions under the management group that is inclusive of all the resource groups and resources within them.

After you have identified the role, security principal, and scope at which the role will be assigned, you can make the assignment. Remember, security principals do not have access to Azure resources until a role assignment is made, and that access can be revoked by removing a role assignment.

Important Role assignment limits

You can have up to 2,000 role assignments in each subscription, and you can have up to 500 role assignments per management group.

To create and remove role assignments, you must have

`Microsoft.Authorization/role Assignments/*` permission at the necessary scope. This permission is granted through the Owner or User Access Administrator built-in roles, or it can be included in custom roles.

Note Azure role assignments

With Azure role assignments, there is no way to revoke access rights at a child scope through the application of a more restrictive role assignment because the role assignment is inherited from the parent. It is, however, possible to apply a deny assignment at a scope when using Azure Blueprints and resource locks. Deny assignments are evaluated before role assignments and can be used to exclude service principals from accessing child scopes. For more information, see <https://docs.microsoft.com/azure/governance/blueprints/tutorials/protect-new-resources>.

Create a custom role

In addition to built-in roles available in Azure, you might face a situation where you need to create the custom role to provide set of permissions that are not available in any of the built-in roles. Custom roles can be created and assigned through Azure portal, Azure PowerShell, Azure CLI, and REST API. In this chapter, we mainly cover how to create the custom role using Azure portal.

Important Custom roles

Custom roles can be shared between subscriptions that trust the same Azure AD directory. There is a limit of 5,000 custom roles per directory, though Azure Germany and Azure China 21Vianet can have up to 2,000 custom roles for each directory.

There are three ways you can create custom roles in Azure portal:

- Clone from the existing built-in roles available
- Start from scratch
- Start from a JSON file to define the custom permissions

To clone a built-in role, open the **Access Control (IAM)** blade by accessing subscription or resource group and then clicking **+ Add > Add Custom Role**, as shown in [Figure 1-16](#).

The screenshot shows the Azure Access control (IAM) blade. On the left sidebar, under 'Cost Management', there are links for 'Cost analysis', 'Budgets', and 'Advisor recommendations'. The main area has a header with 'Search (Ctrl+ /)', 'Add', 'Download role assignments', 'Edit columns', 'Refresh', 'Remove', and 'Got feedback?'. Below the header, there are tabs: 'Overview', 'Activity log', 'Access control (IAM)' (which is selected), 'Tags', 'Diagnose and solve problems', 'Security', and 'Events'. A sub-menu for 'Access control (IAM)' is open, showing 'Add role assignment', 'Add co-administrator', and 'Add custom role'. The 'Add custom role' option is highlighted. The main content area displays a table of built-in roles:

Name	Type	Users
Owner	BuiltinRole	0
Contributor	BuiltinRole	0
Reader	BuiltinRole	0
User Access Administrator	BuiltinRole	1
AcrDelete	BuiltinRole	0
AcrImageSigner	BuiltinRole	0

FIGURE 1-16 Add custom role option in the Access control (IAM) blade

On the **Create A Custom Role** blade, and under **Baseline Permissions**, select **Clone A Role**. Next, from the **Role To Clone** drop-down menu, select the desired role, such as **Virtual Machine Contributor**, as shown in [Figure 1-17](#). You can select the role with the nearest identical permissions from the built-in roles.

On next screen, you have **Add Permissions** and **Exclude Permissions**

options, as shown in [Figure 1-18](#). This screen displays all the permissions associated with the built-in role you selected in [Figure 1-17](#).

When you click **+Add Permissions**, you can search from all the different permissions available from the catalog. For example, search for **virtual machine**, as shown in [Figure 1-19](#). You can select **Microsoft Compute** to access operations available for this resource provider.

Important ARM Resource provider orerations

To explore all the operations available for each Azure Resource Manager resource provider, see <https://docs.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations>.

Create a custom role (Preview)

 Got feedback?

Basics Permissions Assignable scopes JSON Review + create

To create a custom role for Azure resources, fill out some basic information. [Learn more ↗](#)

* Custom role name ⓘ

Virtual Machine Operator



Description

Custom role for Virtual Machine Operators

Baseline permissions ⓘ



Clone a role



Start from scratch



Start from JSON

Role to clone

Virtual Machine Contributor ⓘ



[Review + create](#)

[Previous](#)

[Next](#)

FIGURE 1-17 Creating a custom role

Create a custom role

Got feedback?

Basics Permissions Assignable scopes JSON Review + create

[+ Add permissions](#) [+ Exclude permissions](#)

Click **Add permissions** to select the permissions you want to add to this custom role.

To add a wildcard (*) permission, you must manually add the permission on the **JSON** tab. [Learn more](#)

To exclude specific permissions from a wildcard permission, click **Exclude permissions**. [Learn more](#)

Permission	Description	Permission type	
Microsoft.Authorization/*/read	--	Action	
Microsoft.Compute/availabilitySets/*	--	Action	
Microsoft.Compute/locations/*	--	Action	
Microsoft.Compute/virtualMachines/*	--	Action	
Microsoft.Compute/virtualMachineScaleSets/*	--	Action	
Microsoft.Compute/disks/write	Creates a new Disk or updates an existing one	Action	
Microsoft.Compute/disks/read	Get the properties of a Disk	Action	
Microsoft.Compute/disks/delete	Deletes the Disk	Action	
Microsoft.DevTestLab/schedules/*	--	Action	
Microsoft.Insights/alertRules/*	--	Action	
Microsoft.Network/applicationGateways/backendAddress... Joins an application gateway backend address pool. Not alertable.	Action		
Microsoft.Network/loadBalancers/backendAddressPools... Joins a load balancer backend address pool. Not Alertable.	Action		
Microsoft.Network/loadBalancers/inboundNatPools/join... Joins a load balancer inbound NAT pool. Not alertable.	Action		
Microsoft.Network/loadBalancers/inboundNatRules/join... Joins a load balancer inbound nat rule. Not Alertable.	Action		
Microsoft.Network/loadBalancers/probes/join/action Allows using probes of a load balancer. For example, wit...	Action		
Microsoft.Network/loadBalancers/read Gets a load balancer definition.	Action		

Definitions

Management plane

Actions specify the operations that a role is allowed to perform. NotActions specify the operations that are excluded from the allowed Actions (this is useful if a role has wildcards).

Data plane

DataActions specify the operations that a role is allowed to perform to the data within an object. NotDataActions specify the operations that are excluded from the allowed DataActions (this is useful if a role has wildcards).

Wildcards (*)

A wildcard (*) extends a permission to everything that matches the string you provide. To add a wildcard permission, use the **JSON** tab.

[Review + create](#)

[Previous](#)

[Next](#)

FIGURE 1-18 Add or exclude permissions while creating a custom role

Add permissions

X

- 1 Search for permissions to add to your custom role. For example, search for "virtual machines" to find permissions related to virtual machines.

virtual machine

Microsoft Azure Lab Services

Set up labs for classrooms, trials, development and testing, and other scenarios.

Microsoft ClassicCompute

Microsoft.ClassicCompute

Microsoft ClassicStorage

Microsoft.ClassicStorage

Microsoft Compute

Access cloud compute capacity and scale on demand (such as virtual machines) and only pay for the resources you use.

Microsoft DevTest Labs

Quickly create environments using reusable templates and artifacts.

Microsoft Marketplace Ordering

Microsoft.MarketplaceOrdering

Microsoft Network

Connect cloud and on-premises infrastructure and services to provide your customers and users the best

Microsoft Web Apps

Quickly create and deploy mission critical web apps at scale.

Microsoft.SqlVirtualMachine

Host enterprise SQL Server apps in the cloud.

microsoft.vmware

Microsoft.VMware

Virtual Machine Image Builder

Microsoft.VirtualMachineImages

VMware Solution by CloudSimple

Run your VMware workloads natively on Azure.

Add

Cancel

Download all permissions

FIGURE 1-19 Adding the Microsoft Compute permission

Once you select **Microsoft Compute**, you will have options to select specific permissions from **Actions** and **Data Actions** tabs. The **Actions** tab contains the operations that a role can perform, and the **Data Actions** tab contains the operations that a role can perform on the data within an object. Similarly, if you want to exclude permissions, the **Not Actions** and **Not Data Actions** tabs contain permissions that a role is not allowed to perform based on selection (see [Figure 1-20](#)).

After you select the required permissions, you must select the **Assignable Scopes** for this custom role. The scope can be defined as a **Subscription**, **Resource Group**, or **Resource Level**. The custom role must have at least one valid scope assigned (see [Figure 1-21](#)).

Microsoft.Compute permissions

[All resource providers](#)

i Search for permissions to add to your custom role. For example, search for "virtual machines" to find permissions related to virtual machines.

virtual machine

Actions Data Actions

Permission	Description
Microsoft.Compute/availabilitySets/vmSizes	
<input type="checkbox"/> Read : List Virtual Machine Sizes for Availability Set ⓘ	List available sizes for creating or updating a virtual machine in the availability set
Microsoft.Compute/locations/vmSizes	
<input checked="" type="checkbox"/> Read : List Available Virtual Machine Sizes in Location ⓘ	Lists available virtual machine sizes in a location
Microsoft.Compute/locations/vsmOperations	
<input type="checkbox"/> Read : Get Operation for Virtual Machine Scale Set with the Virtual Machine Runtime Service Extension ⓘ	Gets the status of an asynchronous operation for Virtual Machine Scale Set with the Virtual Machine Runtime Service Extension
Microsoft.Compute/virtualMachineScaleSets	
<input type="checkbox"/> Read : Get Virtual Machine Scale Set ⓘ	Get the properties of a Virtual Machine Scale Set
<input type="checkbox"/> Write : Create or Update Virtual Machine Scale Set ⓘ	Creates a new Virtual Machine Scale Set or updates an existing one
<input type="checkbox"/> Delete : Delete Virtual Machine Scale Set ⓘ	Deletes the Virtual Machine Scale Set
<input type="checkbox"/> Other : Delete Virtual Machines in a Virtual Machine Scale Set ⓘ	Deletes the instances of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Start Virtual Machine Scale Set ⓘ	Starts the instances of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Power Off Virtual Machine Scale Set ⓘ	Powers off the instances of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Restart Virtual Machine Scale Set ⓘ	Restarts the instances of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Deallocate Virtual Machine Scale Set ⓘ	Powers off and releases the compute resources for the instances of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Manual Upgrade Virtual Machine Scale Set ⓘ	Manually updates instances to latest model of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Reimage Virtual Machine Scale Set ⓘ	Reimages the instances of the Virtual Machine Scale Set
<input type="checkbox"/> Other : Reimage all Disks for a Virtual Machine Scale Set ⓘ	Reimages all disks (OS Disk and Data Disks) for the instances of a Virtual Machine Scale Set

Add

Cancel

FIGURE 1-20 Permission list under the Actions tab



FIGURE 1-21 Assignable scopes selection while creating custom role

On the next screen, you will be presented with the JSON code based on the selection made on the prior screens. This code can be downloaded as a .json file, or it can be copied to reuse later. You can proceed to the **Review + Create** screen to create the custom role (see [Figure 1-22](#)).

Create a custom role

Got feedback?

Basics Permissions Assignable scopes JSON Review + create

Here is your custom role in JSON format. [Learn more](#)

Download



Edit

Using JSON

Add permissions

Specify an operation as an Action, NotAction, DataAction, or NotDataAction. Permission strings use the format {Company}.
{ProviderName}/{resourceType}/{action}.

Add wildcards (*)

Add wildcards (*) to a permission string to include all permissions that match the string. For example, if you specify Microsoft.Compute/* as an Action, your role can perform all management operations in Microsoft.Compute.

Add assignable scopes

Subscription-level scopes use the format /subscriptions/{subscriptionId}. Formats vary with other levels of scope (e.g. resource groups).

```
1 {
2   "properties": {
3     "roleName": "Virtual Machine Operator",
4     "description": "Custom role for Virtual Machine Operators",
5     "assignableScopes": [
6       "/subscriptions/00072028-9dce-4729-9b2a-a10e92054447"
7     ],
8     "permissions": [
9       {
10         "actions": [
11           "Microsoft.Authorization/*/read",
12           "Microsoft.Compute/availabilitySets/*",
13           "Microsoft.Compute/locations/*",
14           "Microsoft.Compute/virtualMachines/*",
15           "Microsoft.Compute/virtualMachineScaleSets/*",
16           "Microsoft.Compute/disks/write",
17           "Microsoft.Compute/disks/read",
18           "Microsoft.Compute/disks/delete",
19           "Microsoft.DevTestLab/schedules/*",
20           "Microsoft.Insights/alertRules/*",
21           "Microsoft.Network/",
22           "Microsoft.Network/loadBalancers/backendAddressPools/join/action",
23           "Microsoft.Network/loadBalancers/inboundNatPools/join/action".

```

Review + create

Previous

Next

FIGURE 1-22 Assignable scopes selection while creating custom role

Newly created custom roles can be accessed from the **Roles** tab (see [Figure 1-23](#)). Custom roles appear in the Azure portal with an orange resource icon.

The screenshot shows the Azure portal's Roles tab. At the top, there are buttons for Add, Edit columns, Refresh, Remove, and Got feedback?. Below these are navigation links: Check access, Role assignments, Deny assignments, Classic administrators, and Roles (which is underlined, indicating it is the active tab). A descriptive text follows: "A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles." A link to "Learn more" is provided. The main area is titled "Showing 5 of 168 roles". It has two columns: "Name" and "Type". The "Name" column contains a text input field with "virtual machine" and a dropdown menu. The "Type" column shows a dropdown menu. Below this, a table lists roles:

Name	Type
Classic Virtual Machine Contributor	BuiltInRole
Virtual Machine Administrators Login	BuiltInRole
Virtual Machine Contributor	BuiltInRole
Virtual Machine Operator (highlighted with a blue checkmark and orange border)	CustomRole
Virtual Machine User Login	BuiltInRole

A callout box points to the "Virtual Machine Operator" row, labeled "Custom role for Virtual Machine Operators". A vertical line labeled "Orange resource icon" points from the bottom of the callout box down to the orange square icon next to the checked checkbox in the "Name" column of the table.

FIGURE 1-23 Assignable scopes selection while creating custom role

Alternatively, built-in roles can be cloned by selecting a role from the **Roles** tab. For example, you could select **Virtual Machine Contributor**, click the ellipsis (...), and then click **Clone** (see [Figure 1-24](#)).

[Add](#) [Edit columns](#) [Refresh](#) | [Remove](#) | [Got feedback?](#)

Check access Role assignments Deny assignments Classic administrators Roles

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#) 

Name  Type 

virtual machine	v
-----------------	---

Showing 4 of 167 roles

<input type="checkbox"/>	Name	Type	Users	Groups	Service Principals	...
<input type="checkbox"/>	Classic Virtual Machine Contributor 	BuiltinRole	0	0	0	
<input type="checkbox"/>	Virtual Machine Administrator Login 	BuiltinRole	0	0	0	
<input checked="" type="checkbox"/>	Virtual Machine Contributor 	BuiltinRole	0	0	0	
<input type="checkbox"/>	Virtual Machine User Login 	BuiltinRole	0	0	0	

Permissions

Delete

Edit

Clone

FIGURE 1-24 Cloning a role

Important required Permission to create a custom role

To create a custom role, you must have the `Microsoft.Authorization/roleDefinitions/write` permission on all `AssignableScopes`.

You can also create custom role by choosing **Start From Scratch** from the **Baseline Permissions**. This option could be time consuming because you might need to select all the permissions, one-by-one, to create a custom role from scratch.

Similarly, custom roles can be defined using a JSON (JavaScript Object Notation) file by selecting **Start From JSON** under **Baseline Permissions**. The JSON file contains the role definitions:

- A name represented by the `Name` attribute.
- An identifier represented by the `Id` attribute.
- A description represented by the `Description` attribute.
- A flag that denotes if the role is custom or built-in represented by the `IsCustom` attribute, which is set to `false` for built-in roles; this should be set to `true` when authoring custom roles.
- The actions that can or cannot be performed within the Azure management plane are represented by the `Actions []` and `NotActions []` attributes, respectively.
- Optionally, the scopes at which the role is available through the `AssignableScopes []` attribute.

Interpret access assignments

To manage access (role) assignments, you can use the Azure portal, the Azure CLI, Azure PowerShell, Azure SDKs, or the Resource Manager REST APIs. In the following section, we walk through how to manage role assignments using the Azure portal.

In the Azure portal, the Access Control (IAM) blade is used to manage access to resources, and it is where role assignments are applied or removed. The Access Control (IAM) blade is available at any scope where role assignments can be made (management group, subscription, resource group, and resource). To find the Access Control (IAM) blade, navigate to the resource or service where you want to manage role assignments.

In the following example, the Virtual Machine Contributor built-in role will be assigned to a user at the resource group scope.

In the Azure portal, navigate to a resource group by selecting **Resource Groups** in the left navigation pane, selecting a resource group from the **Resource Groups** pane, and then selecting the **Access Control (IAM)** blade.

From the Access control (IAM) blade, you can

- Check the effective access rights for a security principal at the current scope through the **Check Access** tab, including being able to view access rights inheritance from a parent scope
- Edit role assignments, both granting and revoking access rights through the **Role Assignments** tab
- View deny assignments, which are controlled by Microsoft, through the **Deny Assignments** tab
- View and manage permissions to classic resources through the **Classic Administrators** tab
- View the available roles, both built-in and custom, through the **Roles** tab

Important Deny assignments in the IAM blades

The **Deny Assignments** tab of the **Access Control (IAM)** blade cannot be used to make or alter deny assignments. Deny assignments are set and controlled by applying a resource lock for resources created through Azure Blueprints.

To create a role assignment, navigate to the **Role Assignments** tab and click **Add**, as shown in [Figure 1-25](#).

The screenshot shows the 'Access control (IAM)' blade for a resource group named 'ExamRefRg'. The left sidebar includes options like Overview, Activity log, Access control (IAM), Tags, Events, and Settings. The 'Access control (IAM)' option is selected. The main area has tabs for Check access, Role assignments (which is selected), Roles, Deny assignments, and Classic administrators. A summary bar indicates 12 role assignments out of a limit of 2000. Below this are search and filter fields for Name, Type (set to All), and Role (set to 3 selected).

FIGURE 1-25 Role Assignments tab on the Access Control (IAM) blade

After clicking **Add**, select **Add Role Assignment**, as shown in Figure 1-26.

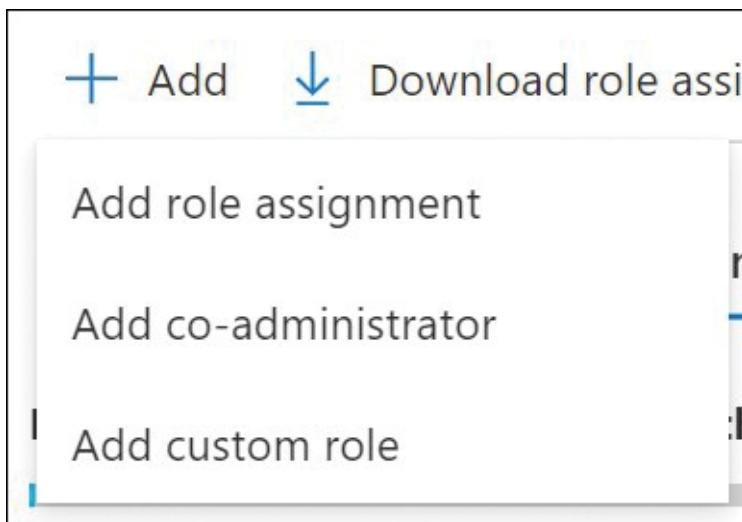


FIGURE 1-26 Add Role Assignment

On the **Add Role Assignment** blade, select the role and from the **Assign Access To** drop-down menu, select the security principal you want to assign the role to. The **Select** drop-down menu can be used to filter users, groups, or service principals found in the Azure AD tenant associated with the Azure subscription. Click **Save** when you are done. Figure 1-27 shows an example where the user, `cloudadmin@opsgility.onmicrosoft.com`, is being granted access to the **Virtual Machine Contributor** role. In the example directory, two security principals were returned from the filtered list using the search term “cloud”—(CloudynAzureCollector and Cloud Administrator). A single principal (Cloud Administrator) was selected (displayed under Selected Members) to apply to the Virtual Machine Contributor role assignment.

Add role assignment

X

Role ⓘ

Virtual Machine Contributor ⓘ



Assign access to ⓘ

Azure AD user, group, or service principal



Select ⓘ

harshul



Harshul Patel

Selected members:



CIE Administrator

harshulp_outlook.com#EXT#@MSP13... [Remove](#)

FIGURE 1-27 Add Role Assignment blade

After clicking **Save**, you will see the role assignment on the **Role Assignments** blade. To remove a role assignment, from the **Role Assignments** tab, select one or more security principals and click **Remove**. An example is shown in Figure 1-28.

Home >

ExamRefRg | Access control (IAM)

Resource group

Search (Ctrl+ /) Add Download role assignments Edit columns Refresh Remove Got feedback?

Overview Check access Role assignments Roles Deny assignments Classic administrators

Activity log Access control (IAM) Tags Events

Number of role assignments for this subscription 12

Settings Name Type Role Scope Group by

Search by name or email All 3 selected All scopes Role

Quickstart 13 items (6 Users, 5 Groups, 2 Service Principals)

Name	Type	Role	Scope
Chris Green	User	Contributor	This resource
Chris White	User	Contributor	This resource
Divya Patel	User	Contributor	This resource
example-app	App	Contributor	Subscription (Inherited)
ExamRef-R-go3-asr-automationaccount_oUOMVRzAP9hSQ6eL	App	Contributor	Subscription (Inherited)
Harshul Patel	User	Contributor	This resource
Robin	User	Contributor	This resource
IT Department	Group	Reader	This resource
sg-Executive	Group	Reader	This resource
sg-IT	Group	Reader	This resource
sg-Operations	Group	Reader	This resource

Properties Locks Export template Cost Management Cost analysis Cost alerts (preview) Budgets Advisor recommendations Monitoring Insights (preview) Alerts Metrics Diagnostic settings Logs Advisor recommendations Workbooks

FIGURE 1-28 Remove a role assignment

Skill 1.3: Manage subscriptions and governance

An Azure subscription, which forms the core of an Azure environment, is a foundational component of every Azure implementation. Every resource that you create in Azure resides in an Azure subscription, which is a billing boundary for Azure resources with per-resource, role-based access controls.

As you build and deploy services in Azure, you will create many types of resources. For instance, when creating your first virtual machine, you will also deploy many other resources including

- A disk for the OS
- A network interface for the VM
- A virtual network and subnet for that network interface to bind to
- A network security group (in a default portal configuration)

It is important to understand that many services in Azure create multiple resources, and how you manage those resources will be driven by organizational Policy and the lifecycle of your infrastructure hosted in Azure.

This skill covers:

- [Configure Azure Policies](#)
- [Configure resource locks](#)
- [Apply and manage tags on resources](#)
- [Create and manage resource groups](#)
- [Manage Azure Subscriptions](#)
- [Configure management groups](#)
- [Configure cost management](#)

A resource in Azure is a single service instance, which can be a virtual machine, a virtual network, a storage account, or any other Azure service (see Figure 1-29).



FIGURE 1-29 Azure resource

Resource groups are logical groupings of resources or those single-service instances ([Figure 1-30](#)).

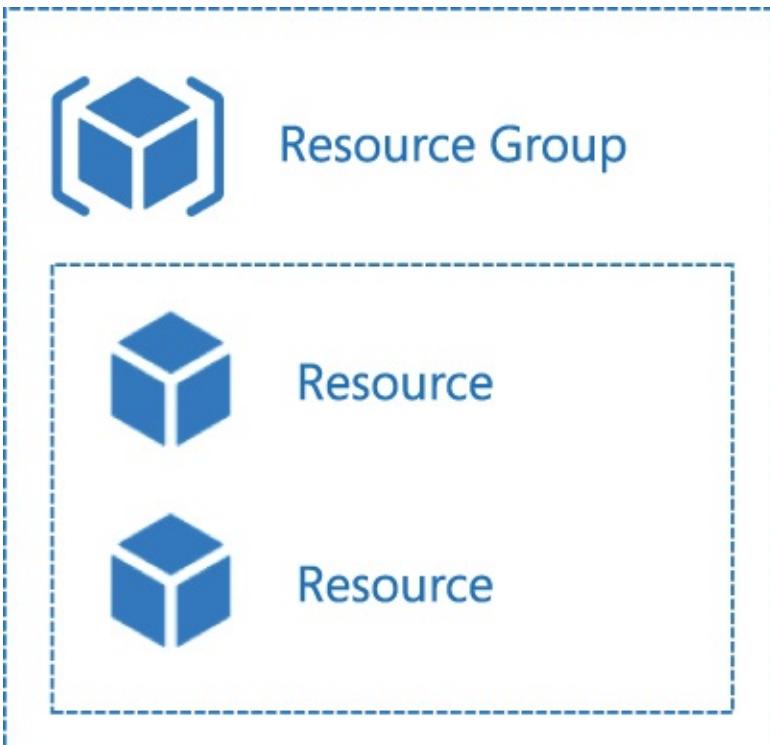


FIGURE 1-30 Azure hierarchy

Using a multiple virtual machine instances, you can group the instances and manage them as one unit. Each resource in Azure can only exist in one resource group, and resource groups cannot be renamed. There are no limitations to the types of resources that can be logically contained within a resource group, and there are no limitations on the regions in which resources must reside when in a resource group.

Figure 1-31 shows this hierarchy with an Azure subscription, multiple resource groups, and the resources that reside within those resource groups.

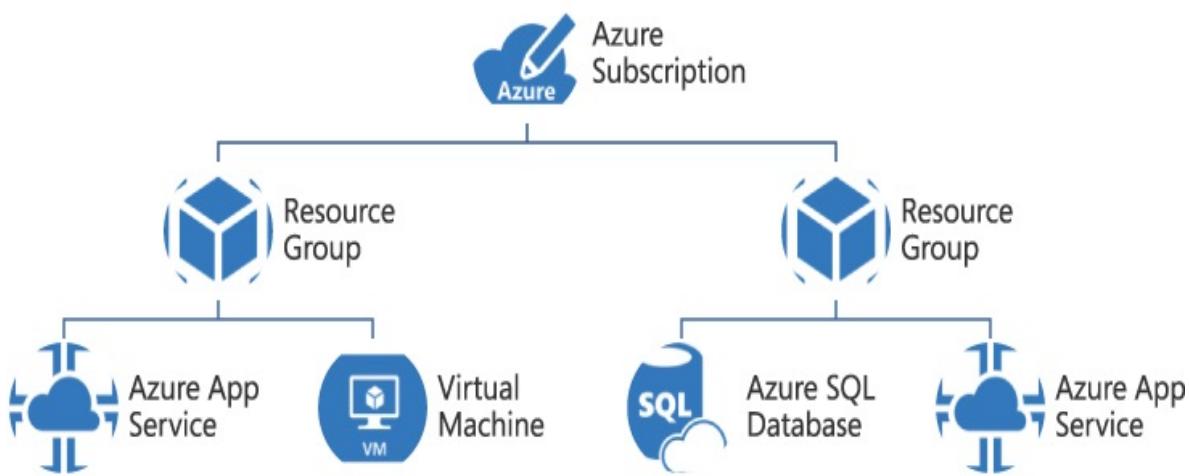


FIGURE 1-31 Azure hierarchy

Configure Azure policies

Azure Policy is an Azure service that can be used to create, assign, and manage policies that enforce governance in your Azure environment. This includes the application of rules that allow or deny a given resource type, apply tags automatically, and even enforce data sovereignty. Azure RBAC and Azure Policy are often used in combination. Where Azure RBAC controls individual user access, group access, and rights to your Azure environments at a specific scope, Azure Policy provides a mechanism to express how the environment is governed for all users at a specified scope regardless of any RBAC assignments. Another way to state this is that Azure RBAC is a default deny mechanism with an explicit allow mechanism, whereas Policy is a default allow mechanism with an explicit deny system.

To implement Policy, a Policy definition must first be authored. That Policy definition is then assigned a specific scope using a Policy assignment. Recall that scope refers to what your policy is assigned to with valid scopes, a management group, a subscription, a resource group, or a resource.

Policy definitions can also be packaged using initiative definitions and applied to a scope using initiative assignments. Policy and initiative definitions both support parameter sets, which help simplify the re-use of a Policy at multiple scopes.

A Policy definition describes your desired behavior for Azure resources at the time resources are created or updated. Through a Policy definition, you declare what resources and resource features are considered compliant within your Azure environment and what should happen when a resource is non-compliant. For example, you can create a Policy that states that resources can only be created in the East US and West US regions for an entire subscription. If a user attempts to create a resource in East US, Azure Policy can deny the creation of the resource because it does not meet the stated compliance goal for allowed regions. In this example, Policy is used to deny the creation of a resource and to enforce organizational standards. As we further explore Policy, you will learn that Policy can be used as not just a deny mechanism, but it also can be used as an auditing and creation mechanism.

Policy definitions are authored in JSON. The schema for Azure Policy can be downloaded from <https://schema.management.azure.com/schemas/2019-06-01/PolicyDefinition.json>. A Policy definition contains these elements:

- Mode
- Parameters
- Display Name
- Description
- Policy Rule
 - Logical Evaluation
 - Effect

Note Policy definition

While you do not need to memorize the schema, it is worthwhile to understand the elements of a Policy definition and how to build your own

policies from a blank template when necessary. Microsoft offers a number of built-in Policy definitions and maintains a repository of samples at <https://docs.microsoft.com/en-us/azure/governance/Policy/samples/> and <https://github.com/Azure/azure-Policy>.

Policy definitions can be created through the Azure portal by browsing to the Policy service at **All Services** and then choosing **Policy > Definitions**. From this blade, you can manage both built-in policies and any custom policies that you create. [Figure 1-32](#) shows a list of the built-in policies for selected subscription.

Scope	Definition type		Type	Category	
Visual Studio Ultimate with M...	Policy		Built-in	All categories	
Name	Definition location	Policies	Type	Definition type	Category
Audit virtual machines without disaster recovery config...			Built-in	Policy	Compute
Azure Backup should be enabled for Virtual Machines			Built-in	Policy	backup
[Preview]: Show audit results from Windows VMs on wh...			Built-in	Policy	Guest Configuration
Azure Cosmos DB allowed locations			Built-in	Policy	Cosmos DB
SQL managed instance TDE protector should be encrypt...			Built-in	Policy	SQL
[Preview]: Network traffic data collection agent should ...			Built-in	Policy	Monitoring
Deploy Diagnostic Settings for Service Bus to Log Analy...			Built-in	Policy	Monitoring
Deploy Log Analytics agent for Linux VMs			Built-in	Policy	Monitoring
Private endpoint should be enabled for PostgreSQL serv...			Built-in	Policy	SQL
Vulnerability Assessment settings for SQL server should ...			Built-in	Policy	SQL
Diagnostic logs in Azure Data Lake Store should be ena...			Built-in	Policy	Data Lake
Audit VMs that do not use managed disks			Built-in	Policy	Compute
CORS should not allow every resource to access your Fu...			Built-in	Policy	App Service
Deploy Log Analytics agent for Windows VMs			Built-in	Policy	Monitoring
Deploy Diagnostic Settings for Search Services to Log A...			Built-in	Policy	Monitoring
Adaptive Network Hardening recommendations should ...			Built-in	Policy	Security Center
There should be more than one owner assigned to your...			Built-in	Policy	Security Center
Disk encryption should be applied on virtual machines			Built-in	Policy	Security Center
Configure backup on VMs of a location to an existing c...			Built-in	Policy	Backup

FIGURE 1-32 Azure built-in policies

Keep in mind that Policy can also be managed and applied at the management group scope. By associating policies with management groups, Policy definitions and Policy assignments can be shared across multiple subscriptions. This includes the ability to monitor multiple subscriptions for compliance. It also allows you to secure the management of organization-wide Policy at a level above a single subscription.

When managing resource groups—and in many cases the multiple Azure services that reside within them—Azure Policy with Policy definitions and Policy assignments can be used to govern those resources. Initiative definitions and initiative assignments can be used to govern those same resources, but instead of applying multiple Policy definitions and making multiple Policy assignments, you can package or group multiple definitions into a single initiative and then assign that initiative to your desired scope.

Controlling resource groups with Azure Policy is done by scoping the assignment of Policy and initiatives. Recall that Azure Policy supports multiple scopes:

- **Management group.** Assignments scoped at the management group (either the Tenant Root Group or a child group) apply to all child resources in the management group including child management groups, all subscriptions, resource groups, and resources.
- **Subscription.** Assignments scoped to a subscription apply to all child resources in the subscription resource groups and resources.
- **Resource group.** Assignments scoped to a resource group apply to all child resources in the resource group.

When creating assignments, it is also possible to configure excluded scopes. You always have the ability to exclude a subscope. For example, when scoping an assignment to a management group, any subscriptions, resource groups, or even resources that are children of the management group, can be excluded. When scoping an assignment to a subscription, child resource groups and resources can be excluded. When scoping an assignment to a resource group, only child resources can be excluded.

The flexibility of Policy scoping is a powerful feature of Azure Policy. This allows you to model your environments with rich declarations in the form of

Policy definitions that are applied exactly as required by your organization's governance needs.

Imagine you have an environment with the following requirements:

- All resources should be tagged with the tag "Environment" and the value "Dev/Test".
- Only A-Series and D-Services virtual machines can be created, specifically Standard A0, A1, and D2 virtual machines that are not promotional.
- Resources in the rgCoreNetwork resource group are exempt from these policies.

To model this environment with Azure Policy, you can create two Policy definitions (or use built-in Policy definitions where applicable), as shown in [Table 1-2](#).

TABLE 1-2 Azure Policy definitions example

Policy Field	Policy Effect	Description
Type	Deny	Do not create virtual machines if they are not in the A-Series or D-Series SKU
tags	Append	Append tag name "Environment" and tag value "Dev/Test" to all resources

In the Azure portal, browse to the **Policy** service and select the **Definitions** blade. To reduce administrative overhead, a new initiative definition will be created. Initiative definitions are a collection of Policy definitions that are focused on the same goal. They allow for a set of policies to be grouped as a single item.

From the **Definitions** blade, select **+Initiative Definition**, as shown in [Figure 1-33](#).



Policy | Definitions

Search (Ctrl+ /)

+ Initiative definition

+ Policy definition

⟳ Refresh

⌚ Overview

⚡ Getting started

📅 Join Preview

📄 Compliance

📌 Remediation

Authoring

📘 Assignments

📁 Definitions

Related Services

(Blueprints (preview))

Resource Graph

User privacy

Scope

Visual Studio Ultimate with M...

...

Definition type

All definition

Name

↑↓

🔒 Audit Windows VMs in which the Administrators group ...

🔒 [Preview]: Audit Windows VMs on which the Log Analyti...

🔒 [Preview]: Audit IRS1075 September 2016 controls and ...

🔒 Audit Windows VMs in which the Administrators group ...

🔒 [Preview]: Audit CIS Microsoft Azure Foundations Bench...

🔒 Enable Monitoring in Azure Security Center

🔒 Audit Windows VMs that do not have the specified appl...

🔒 [Preview]: Audit Australian Government ISM PROTECTE...

🔒 [Preview]: Audit UK OFFICIAL and UK NHS controls and ...

🔒 [Preview]: Audit SWIFT CSP-CSCF v2020 controls and de...

🔒 [Preview]: Audit VMs with insecure password security se...

FIGURE 1-33 Azure Policy Definitions blade

Enter **Dev/Test Compliance** into the **Name** field, select the **Definition Location**, and choose **Create New** from the **Category** options . Type **Custom** in the **Category** field, as shown in [Figure 1-34](#).

Initiative definition

New Initiative definition

BASICS

Definition location *

Visual Studio Ultimate with MSDN

...

Name * ⓘ

Dev/Test Compliance

✓

Description ⓘ

This initiative collects the policies that apply to the Dev/Test environment.

Category ⓘ

Create new Use existing

Custom

Initiative parameters ⓘ

Parameter name

Display name

Type

Allowed values

FIGURE 1-34 Azure Policy Initiative Definition

Add the following built-in policies to the definition and set the values as noted (see [Figure 1-35](#)):

- **Policy Name.** Require a tag and its value on resources, and set the following values:
 - **Tag Name.** Environment
 - **Tag Value.** Dev/Test
- **Policy Name.** Allowed virtual machine SKUs. And set the following values described below:
 - **Allows SKUs.** Standard_A1; Standard_A1_v2; Standard_A2; Standard_A2_v2; Standard_A3; Standard_A4 and Standard_A4_v2

FIGURE 1-35 Azure Policy new initiative definition policies and parameters

Click the **Save** button to save the definition so it can be used in an initiative assignment. Browse to the **Assignments** blade and select **Assign Initiative** (see Figure 1-36).

Policy | Assignments

Search (Ctrl+ /)

Assign initiative

Assign policy

Refresh

Overview

Getting started

Join Preview

Compliance

Remediation

Authoring

Assignments

Definitions

Related Services

Blueprints (preview)

Resource Graph

User privacy

Scope

Visual Studio Ultimate with MSDN

Definition type

All definition types

Search

Filter by name or id...

Category

All categories

Total Assignments ⓘ

1

Initiative Assignments ⓘ

0

Policy Assignments ⓘ

1

name

↑↓ Scope

↑↓ Type

↑↓ Policies

Allowed locations

Visual Studio Ultimate with MSDN

Policy

1

FIGURE 1-36 Azure Policy Assignments blade

To meet the environmental requirements, set the **Scope** of the assignment to the target subscription and configure the **Exclusions** to exclude the rgCoreNetwork resource group. Also, set the **Initiative Definition** to Dev/Test Compliance and set the **Assignment Name** to Dev/Test Compliance. Lastly, make sure **Policy Enforcement** is enabled (see [Figure 1-37](#)). Then click the **Review + Create** and click the **Create** button.

Assign initiative

Basics Parameters Remediation Review + create

Scope

Scope [Learn more about setting the scope *](#)

Visual Studio Ultimate with MSDN



Exclusions

Visual Studio Ultimate with MSDN/rgCoreNetwork



Basics

Initiative definition *

Dev/Test Compliance



Assignment name * ⓘ

Dev/Test Compliance



Description

(This field is required)

Policy enforcement ⓘ

Enabled Disabled

Assigned by

Harshul Patel

[Review + create](#)

[Cancel](#)

[Previous](#)

[Next](#)

FIGURE 1-37 Azure Policy Assign initiative blade

After Policy definitions have been assigned, either through Policy assignments or initiative assignments, the effects of the Policy will be immediately applicable. Policy evaluation for compliance happens about once an hour, which means you might not be able to view the compliance state of a new assignment immediately.

Compliance state can be viewed on the Compliance blade of the Azure Policy service. You can delete, edit, and duplicate the Policy assignment by right-clicking it on the Compliance blade, as shown in [Figure 1-38](#).

Home > Policy

Policy | Compliance

Search (Ctrl+I)

Assign policy Assign initiative Refresh

Overview

Getting started

Join Preview

Compliance

Remediation

Authoring

Assignments

Definitions

Related Services

Blueprints (preview)

Resource Graph

User privacy

Scope

Type

Compliance state

Search

Visual Studio Ultimate with MSDN

All definition types

All compliance states

Filter by name or ID...

Overall resource compliance

Non-compliant initiatives

Non-compliant policies

Non-compliant resources

0%

0

0

3

0 out of 3

out of 0

out of 1

out of 3

Name

Scope

Compliance state

Resource compliance

Non-Compliant Resources

Non-compliant policies

Allowed locations

Visual Studio Ultimate with MSDN

Not started

100% (0 out of 0)

0

View definition

Delete assignment

Edit assignment

Duplicate assignment

FIGURE 1-38 Azure Policy Compliance blade

Configure resource locks

Azure resource locks (sometimes called management locks) are used to prevent the accidental deletion or modification of resources. There are two types of locks:

- **CanNotDelete.** Locks prevent the deletion of a resource. A CanNotDelete lock only prevents deletion of a resource and does not impede the modification of a resource.
- **ReadOnly.** Locks prevent users from modifying a resource, which includes updating or deleting a resource.

Note that both types of resource locks allow for authorized users to read resources; resource locks apply across all users and roles, even custom and privileged roles.

Resource locks, regardless of type, can be applied to the subscription, resource group, and resource scopes. When you apply a lock to a scope, the resources within that scope inherit the lock. This means that a lock applied to the resource group scope applies to all the resources in the resource group. Resource locks apply to all service instances and resources within a scope.

Lock inheritance varies based on the type of lock that is applied. ReadOnly locks are inherited by child resources, while CanNotDelete locks are also inherited by child resources, but it has a side effect. If the CanNotDelete lock is applied to one of the resources in the resource group and you attempt to delete that resource group, it will fail. When you try to delete the resource group, the operation tries to delete all the underlying resources first and won't be able to delete the resource with CanNotDelete lock, hence the resource group deletion would also fail.

Note that resource locks get applied to the management plane of Azure. This means resource locks don't affect the resource's own functionality; instead, they restrict the interactions with other Azure resources. For example, a ReadOnly lock applied to a Storage Account would prevent users from reading the access keys. If you attempt to read or modify the access keys, the operation will fail with a "Cannot access the data plane because of a read lock on the resource or its parent" error, as shown in [Figure 1-39](#).

test342345325 | Access keys

Storage account

Search (Ctrl+ /)

«

 Access control (IAM)

 Tags

 Diagnose and solve problems

 Data transfer

 Events

 Storage Explorer (preview)

Settings

 Access keys

 Geo-replication

 CORS

 Configuration

 Encryption

 Shared access signature

 Firewalls and virtual networks

 Private endpoint connections

 Advanced security

 Static website

 Properties

 Locks

 Export template



Access blocked

The resource is locked

Cannot access the data plane because of a read lock on the resource or its parent.

Summary

Session ID	Resource ID
43ecee74b13b427d9ca3e3c014bad122	/subscriptions/00b72028-9dce-4729-9b2a-a10e920544...
Extension	Content
Microsoft_Azure_Storage	KeyManagementBladeV2
Error code	
403	

FIGURE 1-39 Read-only lock applied to Storage Account

When creating locks, you should exercise caution because they can have unexpected results. Many operations appearing to be read operations require write access within the Azure management plane. For example, the same **ReadOnly** lock on a storage account would prevent users from creating new containers because the action requires write access.

Once you have determined the type of lock you will apply based on your requirements, you can apply the lock through the Azure portal, Azure PowerShell, the Azure CLI, Resource Manager templates, or the REST API.

To create a lock through the Azure portal, browse to the desired scope and select the **Locks** blade. From the blade, select **+Add** to create a new lock. Give the lock a **Lock Name**, select the **Lock Type**, and describe the lock in the **Notes** field, as shown in [Figure 1-40](#).

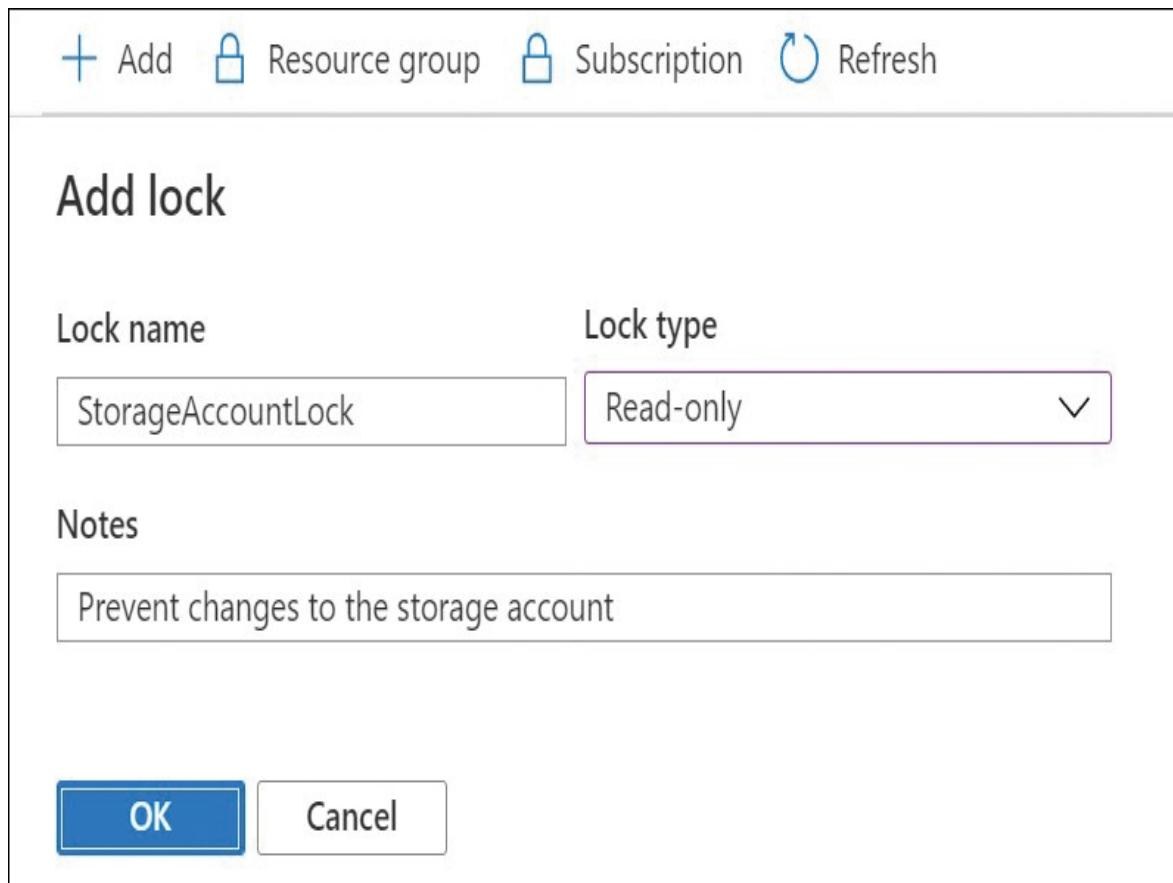


FIGURE 1-40 Creating a lock

Apply and manage tags on resources

Resource tags allow you to apply custom metadata to your Azure resources to logically organize them and to build out custom taxonomies. A tag is a name and a value pair. For example, as you deploy resources in Azure, you want to track the environment the resource is associated with. To do this, you can create a tag called `Environment` and the value `Production` for all resources in production. For downstream environments such as development or test environments, you can use the same `Environment` tag with the `Dev/Test` value. Common tags include the environment with which a resource is associated, a cost center or billing code, and resource owner.

As tags are applied, you can query the resources in your subscription using your tags, and you can even do this across resource groups. This allows you to understand related resources across resource groups for both billing and management. Tags are also included in the billing data for Azure EA subscriptions through the EA Portal and for non-EA subscriptions through the Account Portal at <https://account.azure.com/subscriptions>. Billing exports give clear line of sight for chargeback to understand resource usage and cost. [Figure 1-41](#) shows an example of an export with resource tags from an Azure EA subscription.

PublisherTy	ChargeTy	ServiceName	ServiceTier	Meter	PartNumb	CostUSD
azure	usage	storage	premium ssd managed	p10 disks		10.740285
azure	usage	storage	standard page blob	disk read operations		97.188975
azure	usage	storage	standard page blob	disk write operations		889.79809
azure	usage	storage	tables	batch write operations		9.9925364

FIGURE 1-41 Azure detailed usage export

Note Tags and usage reports

Tags must be applied at the resource scope to be visible in detailed usage exports. Tags applied at the resource group scope are not inherited by child resources. This means that as you are applying tags to your resources in

Azure, you should think about applying tags to each resource to have the clearest line of sight into your usage based on your organizational tags.

When planning for resource tags, any taxonomy should include a strategy for both on-demand (or self-service) tagging and automatic tagging through Azure Policy. In the “[Configure Azure policies](#)” section, you learned how to automatically apply tags using Azure Policy. In this section, you will learn how to create tags and manually apply them to resources.

As you plan your tagging taxonomy, be mindful of the limitations of tags in Azure, as detailed in [Table 1-3](#).

TABLE 1-3 Azure Tag Limitations

Tag Limit	Notes
Resource support	<ul style="list-style-type: none">Not all resource types support tags. This means that you will not be able to apply tags to everything in Azure. For example, management groups and generalized VMs don't support tags. Refer to this link: https://docs.microsoft.com/azure/azure-resource-manager/management/tag-support.
Number of tags	<ul style="list-style-type: none">Resources, resource groups, and subscriptions are limited to 50 tags. Each resource can have different tags.
Tag name	<ul style="list-style-type: none">Tag names cannot exceed 512 characters. For storage accounts, tag names are limited to 128 characters.

Tag value	<ul style="list-style-type: none"> Tag values cannot exceed 256 characters.
Virtual machine tags	<ul style="list-style-type: none"> VMs cannot exceed 2048 characters for all tag names and values combined.
Tag inheritance	<ul style="list-style-type: none"> Tags are not inherited by child resources. Tags applied to a resource group are not applied to resources in that resource group.
Classic resources	<ul style="list-style-type: none"> Tags cannot be applied to classic resources and are only available for resources created in the Azure Resource Manager model.
Illegal characters	<ul style="list-style-type: none"> Tag names cannot contain the following characters: <, >, %, &, \, ?, /

To apply tags to a subscription, resource group, or resource, the user applying the tag must have `write` access to the resource (Contributor role or higher access).

Tags can be created and applied to Azure resources through

- The Azure portal
- Azure PowerShell

- The Azure CLI
- Resource Manager templates
- Resource Manager REST API

This means tags can be applied both in an imperative manner and declaratively through Resource Manager templates. While this can be done through the Azure portal, PowerShell, the CLI, or Resource Manager, templates are better suited when this is being done as resources are created because you don't want to perform this manually for each resource after deployment.

Tags can be applied at the subscription, resource group, and/or the resource level. Note again that there is no inheritance for tags. If you need a tag to be applied to all resources in a resource group, each resource must be tagged individually.

You can apply new tags to resources by using these PowerShell commands:

[Click here to view code image](#)

```
$tags = @{"Environment"="Production"; "Application"="ABC123"}
$resource = Get-AzResource -Name prodServer -ResourceGroup prodRG
New-AzTag -ResourceId $resource.id -Tag $tags
```

Alternatively, you can use the `az tag create` Azure CLI command to create new tags. Also, you can update tags by using the following PowerShell commands:

[Click here to view code image](#)

```
$tags = @{"Environment"="Non-Production"; "Application"="ABC123"}
$resource = Get-AzResource -Name prodServer -ResourceGroup prodRG
Update-AzTag -ResourceId $resource.id -Tag $tags -Operation Replace
```

If you use the `-Operation` parameter with `Update-AzTag` command, it supports three values:

- `Replace` Replaces the specified tags in the listed resources
- `Merge` Merges the newly specified tags with the existing ones and overrides the conflicts for the listed resources
- `Delete` Deletes the specified tags from the listed resources

Alternatively, you can use `az tag update` Azure CLI command to create new tags.

Create and manage resource groups

When creating resource groups, it is important that you consider factors such as a single resource being associated with only one resource group at a time and the following:

- A resource group cannot be nested in another resource group.
- You can add or remove a resource from a resource group at any time.
- You can move a resource from one resource group to another.
- A resource group can be used to scope access control.
- A resource group can be used to scope Policy.
- A resource in a resource group can interact with resources in another resource group.
- A resource group is created in a location. The location of a resource group specifies where the metadata for the resource group is stored. If you have compliance constraints, this is an important consideration.
- Microsoft recommends that all resources in a resource group share the same lifecycle.
- It is not mandatory to have all Azure resources belong to a resource group. Resources deployed to a subscription, tenant, or management group exist outside of resource groups.

Creating a resource group through Azure portal can be an easier task. You just need region or location details along with a valid resource group name (See [Figure 1-42](#)). Optionally, you can also apply tags to the resource group explicitly.

Create a resource group

Basics Tags Review + create

Resource group - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. [Learn more](#)

Project details

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Resource group * ⓘ

ExamRefRG



Resource details

Region * ⓘ

(US) East US



FIGURE 1-42 Creating a resource group

Move resources across resource groups

Some resources in Azure can be moved between resource groups and even across subscriptions, but support for move operations varies based on the service. A reference of services that can be moved can be found at <https://docs.microsoft.com/azure/azure-resource-manager/move-support-resources>. In Figure 1-43, the VM in Resource Group 2 can be moved into Resource Group 1, and it can also be moved across subscriptions into the resource group in Subscription 2.

Important Move operations

Even if a resource states that it supports move operations, there can be other factors that prevent the resource from moving. To find out move operation support for Azure resources, see <https://docs.microsoft.com/azure/azure-resource-manager/management/move-support-resources>.



FIGURE 1-43 Moving resources diagram

During a move operation, your resources will be locked. Both write and delete operations to the Azure resource will be blocked, but the underlying service will continue to function. For example, if you move an Azure App Service, the service will continue to serve web requests to visitors. It can take up to four hours for a move operation to complete. If the move operation fails within the four-hour window, resource manager will reattempt the move operation.

To move resources between subscriptions, both subscriptions must be

associated with the same Azure AD tenant. If the subscriptions do not belong to the same tenant, you can update the target subscription to use the source Azure AD tenant by transferring ownership of the subscription to another account. Note that this operation can have unexpected effects because the Azure AD tenant associated with a subscription is used for RBAC to any currently deployed Azure services.

More Info Transfer subscription or point to new Azure ad tenant

To transfer ownership of an Azure subscription to another account, see <https://docs.microsoft.com/azure/cost-management-billing/manage/billing-subscription-transfer>. Also, to add an Azure subscription to a new Azure AD tenant, see <https://docs.microsoft.com/azure/active-directory/fundamentals/active-directory-how-subscriptions-associated-directory>.

When moving resources between subscriptions, the resource provider of the source resource must also be registered in the target subscription. This is not a concern when moving resources within the same subscription because the resource provider will already be registered.

If you are moving resources between subscriptions, you must also be mindful of resource quotas. For example, if you are moving many virtual machines, you will need to make sure that the target subscription has enough vCPUs available or the move operation will fail. Make sure you validate any quotas prior to moving a resource.

Finally, there are limitations in Azure Resource Manager that affect the number of resources you can move in a single operation. A single move operation in Resource Manager cannot move more than 800 resources. With this constraint, it is recommended that you break large operations into smaller batches. Note that even if you are moving less than 800 resources in a single move request, the operation may still fail by timing out.

If the resource you are moving has any dependent resources, the resources must all be located within the same resource group, and they must all be moved together.

Once you have met the stated prerequisites to a move operation, you are

ready to perform the move operation. You can move the resources with the Azure portal, Azure PowerShell, the Azure CLI, or the REST API. Note that Azure performs basic validation before performing the actual move operation, irrespective of the method being used. Additionally, you can validate the move operation through the REST API with the `validateMoveResources` method without actually performing the move operation. This API validates whether resources can be moved from one resource group to another resource group. If validation succeeds, an HTTP 204 will be returned, and if it fails, an HTTP 409 with an error message will be returned in the response. This method can be called with a POST request to:

[Click here to view code image](#)

```
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{sourceResourceGroupName}/validateMoveResources?api-version=2018-05-01
```

In a POST request, include a request body with "resources" and "targetResourceGroup" properties:

[Click here to view code image](#)

```
{  
  "resources": ["<resource-id-1>", "<resource-id-2>"],  
  "targetResourceGroup": "/subscriptions/<subscription-id>/resourceGroups/<target-group>"  
}
```

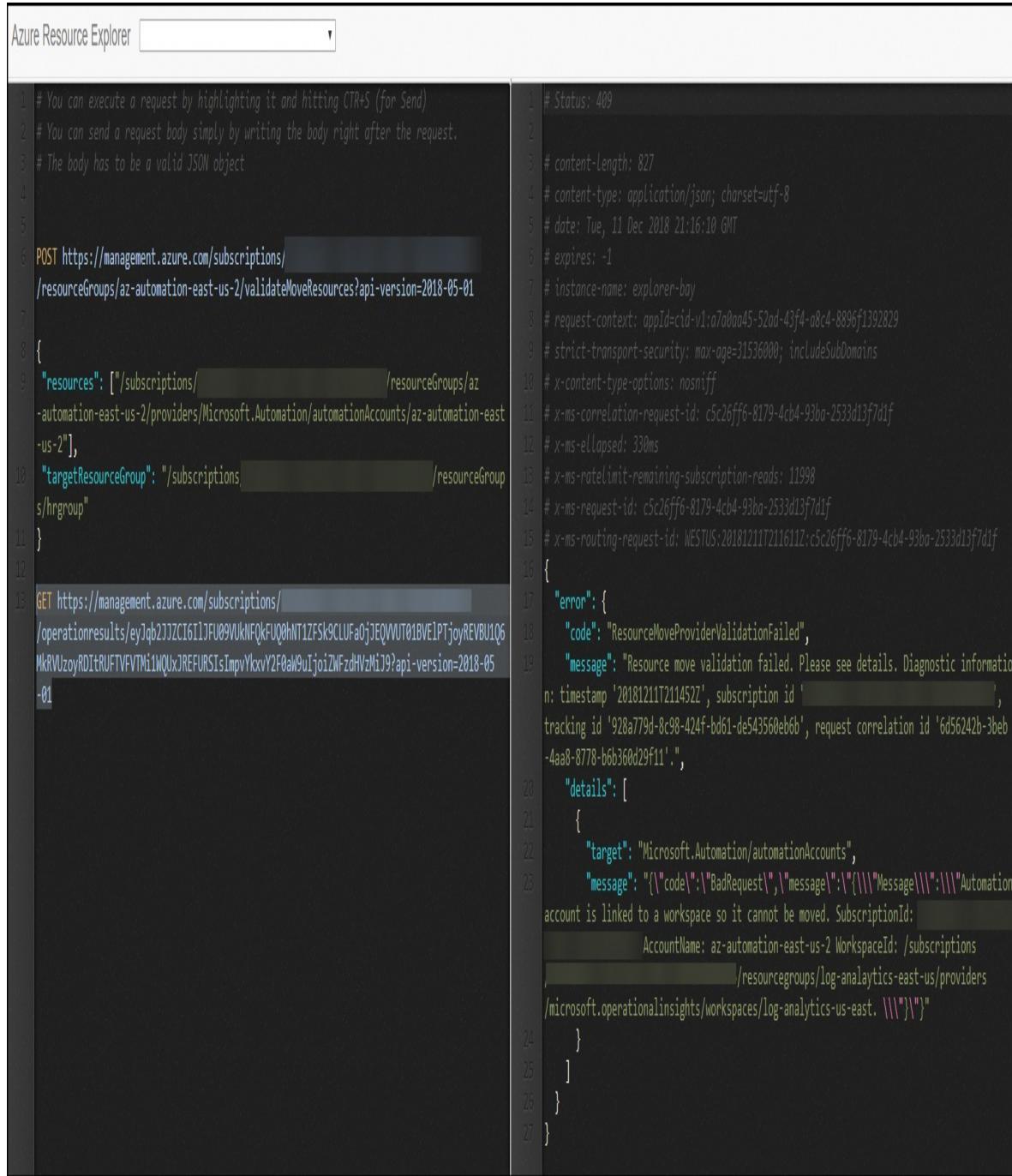
If the request is properly formatted, the operation will return output like the following:

[Click here to view code image](#)

```
Response Code: 202  
cache-control: no-cache  
pragma: no-cache  
expires: -1  
location: https://management.azure.com/subscriptions/<subscription-id>/operationresults/<operation-id>?api-version=2018-02-01  
retry-after: 15  
...
```

The HTTP 202 response code shows the request was accepted. The location

URI can be used in an `HTTP GET` that you can use to check the status of the long-running operation for the final `HTTP 204` or `HTTP 409` status code. [Figure 1-44](#) shows the output of an operation to validate a move request for an Azure Automation account associated with a Log Analytics workspace. As expected, the validation operation returned an `HTTP 409` because this move request cannot be executed.



```

Azure Resource Explorer [ ] ▾

# You can execute a request by highlighting it and hitting CTR+S (for Send)
# You can send a request body simply by writing the body right after the request.
# The body has to be a valid JSON object

POST https://management.azure.com/subscriptions/
/resourceGroups/az-automation-east-us-2/validateMoveResources?api-version=2018-05-01

{
  "resources": [
    "/subscriptions/[REDACTED]/resourceGroups/az-automation-east-us-2/providers/Microsoft.Automation/automationAccounts/az-automation-east-us-2"
  ],
  "targetResourceGroup": "/subscriptions/[REDACTED]/resourceGroups/[REDACTED]"
}

GET https://management.azure.com/subscriptions/
/operationResults/eyJqb2JZCIGI1JFU09VUkNFOQFU00hNT1ZFSk9CLUFa0jJEQWU01BVElPTjoyREVBUI06
MkRVUzoyR0ItRUFTVFVTM1WQUxJREFURSiImpvYkvY2F0aW9uIjoizZWfdhVzMj9?api-version=2018-05-01

1 # Status: 409
2
3 # content-length: 827
4 # content-type: application/json; charset=utf-8
5 # date: Tue, 11 Dec 2018 21:16:10 GMT
6 # expires: -1
7 # instance-name: explorer-bay
8 # request-context: appId=cid-v1:a7a0aa45-52ad-43f4-a8c4-8896f1392829
9 # strict-transport-security: max-age=31536000; includeSubDomains
10 # x-content-type-options: nosniff
11 # x-ms-correlation-request-id: c5c26fff-8179-4cb4-93ba-2533d13f7d1f
12 # x-ms-elapsed: 330ms
13 # x-ms-ratelimit-remaining-subscription-reads: 11998
14 # x-ms-request-id: c5c26fff-8179-4cb4-93ba-2533d13f7d1f
15 # x-ms-routing-request-id: WESTUS:20181211T211611Z:c5c26fff-8179-4cb4-93ba-2533d13f7d1f
16 {
17   "error": {
18     "code": "ResourceMoveProviderValidationFailed",
19     "message": "Resource move validation failed. Please see details. Diagnostic information: timestamp '20181211T211452Z', subscription id '[REDACTED]', tracking id '928a779d-8c98-424f-bd61-de543560eb6b', request correlation id '6d56242b-3beb-4aa8-8778-b6b360d29f11'.",
20     "details": [
21       {
22         "target": "Microsoft.Automation/automationAccounts",
23         "message": "{\"code\":\"BadRequest\", \"message\":\"\\\"Message\\\"\\\"\\\"Automation account is linked to a workspace so it cannot be moved. SubscriptionId: [REDACTED] AccountName: az-automation-east-us-2 WorkspaceId: /subscriptions/[REDACTED]/resourcegroups/log-analytics-east-us/providers/microsoft.operationalinsights/workspaces/log-analytics-us-east. \\\"\\\"\\\"\"}"
24       }
25     ]
26   }
27 }

```

FIGURE 1-44 ValidateMoveResources API Response

To use the Azure portal, browse to the resource group containing the resources, select the **Move** button, and choose **Move To Another Resource Group** or **Move To Another Subscription**, as shown in [Figure 1-45](#).

Home >

ExamRefRg

Resource group

Search (Ctrl+ /) Add Edit columns Delete resource group Refresh Move v Export to CSV Open query Assign tags Delete Export template Feedback

Overview Activity log Access control (IAM) Tags Events Settings Quickstart Deployments Policies Properties Locks Export template

Essentials

Subscription (change) : Visual Studio Ultimate with MSDN
Subscription ID : 00b72028-9dce-4729-9b2a-a10e92054447
Tags (change) : Click here to add tags

Move to another reso...
Move to another subs...

Deployments : 1 Succeeded

Filter by name... Type == all X Location == all X Add filter

Showing 1 to 1 of 1 records. Show hidden types ⓘ No grouping

Name ↑	Type ↑	Location ↑
test342345325	Storage account	East US

FIGURE 1-45 Move button in the Azure portal

You can now select the resources to move and select the destination resource group. Note that you must acknowledge that you might need to update existing tools or scripts to account for the changes in resource IDs (see [Figure 1-46](#)).

Home > ExamRefRg >

Move resources

Resources to move

Select all

 test342345325

Type

Storage account

Move these resources to

Resource group *

I understand that tools and scripts associated with moved resources will not work until I update them to use new resource IDs ⓘ

OK

FIGURE 1-46 Move Resources blade

Remove resource groups

In Azure, you can delete individual resources in a resource group, or you can delete a resource group and all its resources. Deleting a resource group removes all the resources contained within it in one operation. When deleting resource groups, exercise caution because the resource group might contain resources that other resources you have deployed depend on. For example, if you attempt to delete a storage account that is used by an application to store application data, the Azure platform will not recognize that dependency and will allow the storage account to be deleted.

For resources that has dependent resources, you will not be able to delete the target resource until the dependencies have been cleared. For example, to delete a resource group that contains an App Service plan, you must first remove or disassociate any App Services that depend on that plan. An example of attempting to delete an App Service plan with existing App Service associations is shown in [Figure 1-47](#).

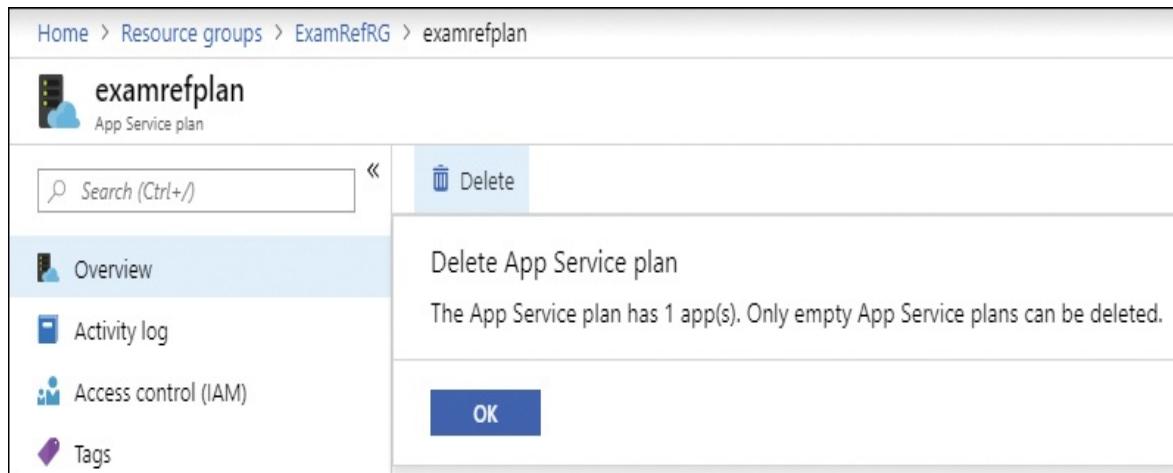


FIGURE 1-47 Delete an Azure resource with dependencies

To delete a resource group, you can use the Azure portal, Azure PowerShell, the Azure CLI, or the REST API.

To delete a resource group in the Azure portal, browse to the resource group and click the **Delete Resource Group** button (see [Figure 1-48](#)).

Home >

ExamRefRg

Resource group

Search (Ctrl+ /) Add Edit columns Delete resource group Refresh Move Export to CSV Open query Assign tags Delete Export template

Overview Activity log Access control (IAM) Tags Events Settings Quickstart Deployments Policies Properties

^ Essentials

Subscription (change) : Visual Studio Ultimate with MSDN Deployments : 1 Succeeded

Subscription ID : 00b72028-9dce-4729-9b2a-a10e92054447

Tags (change) : Click here to add tags

Filter by name... Type == all X Location == all X Add filter

Showing 1 to 1 of 1 records. Show hidden types ⓘ

Name ↑	Type ↑↓
test342345325	Storage account

FIGURE 1-48 Delete Resource Group

In the **Are You Sure You Want To Delete [“resource group name”]?** dialog box that opens, you will need to type the resource group name to confirm that you want to delete it. As shown in [Figure 1-49](#), the blade will also show the affected resources and warn you that the operation is irreversible.

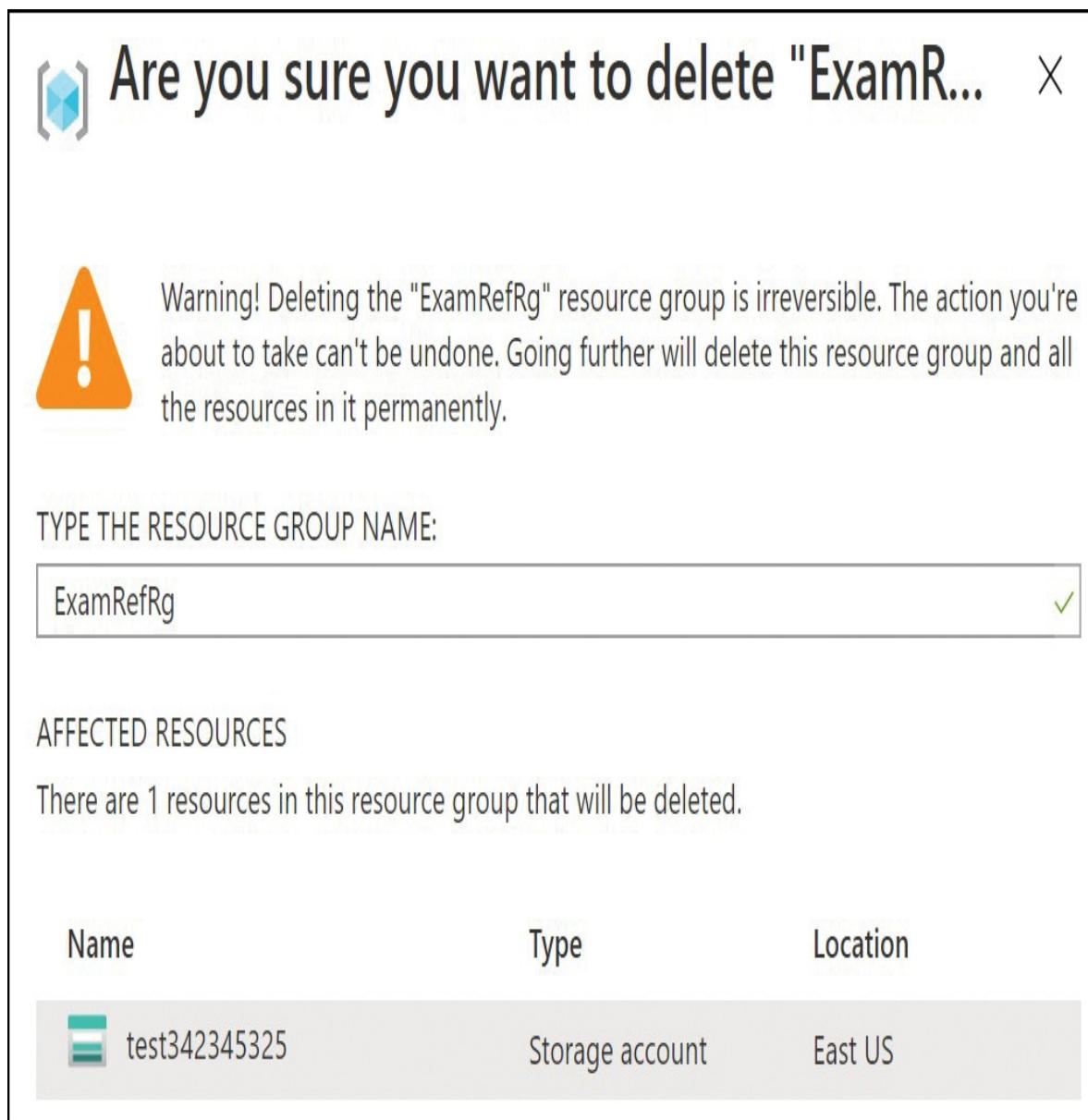


FIGURE 1-49 Azure resource group deletion confirmation

Selecting **Delete** will begin deleting resources immediately. Note that it can take several minutes for a resource group to be deleted because each resource is

deleted individually.

Manage Azure Subscriptions

Azure subscriptions have controls available that govern access to the resources within a subscription, govern cost through quotas and tagging, and govern the resources that are allowed in an environment with Azure Policy.

As discussed earlier, a subscription is a logical unit of Azure services linked to an Azure account, which is an identity in Azure Active Directory (Azure AD). Azure AD is an identity provider for Azure and provides authentication to resources in an Azure subscription. The resources themselves then have role-based access controls applied to them that provide authorization to the resources (see [Figure 1-50](#)).

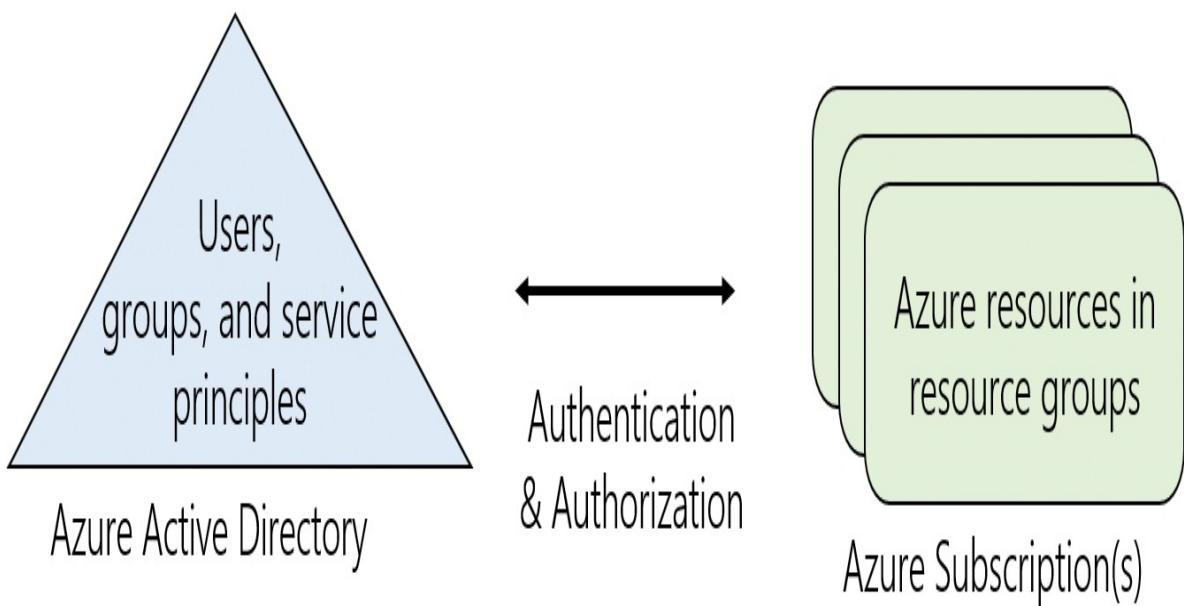


FIGURE 1-50 Azure AD and Azure Subscription relationship

There are multiple ways to obtain an Azure subscription, and a wide range of subscription types (or offers). Some common types include the following:

- Free trial
- Pay-As-You-Go/Web Direct

- Visual Studio/MSDN subscriptions
- Microsoft Resellers
- Cloud Solution Provider
- Microsoft Open Licensing
- Enterprise Agreements

The capabilities of each subscription are similar in that each subscription type allows you to create and manage resources. Some subscription types have restrictions on supported resource types and locations. For example, Visual Studio subscriptions typically do not have a credit card associated with them, which prevents you from purchasing services from the Azure Marketplace, such as network virtual appliances. Visual Studio subscriptions for Azure only have access to a limited number of Azure regions. The regional restrictions for each offer can be viewed at <https://azure.microsoft.com/regions/offers/>.

Assigning administrator permissions

Azure has many different roles for managing access to Azure resources. These include classic subscription administrative roles like Account Administrator, Service Administrator, or Co-Administrator, as well as Azure role-based access controls (RBAC) that are available in Azure Resource Manager (ARM). When managing access to Azure subscriptions and resources, it is recommended to use Azure RBAC roles whenever possible.

More Info roles and relationships

To learn more about the correlation between classic subscription administrator roles, Azure RBAC roles, and Azure AD roles, see <https://docs.microsoft.com/azure/role-based-access-control/rbac-and-directory-admin-roles>.

Classic subscription administrators have full access to an Azure subscription. They can manage resources through the Azure portal, Resource Manager APIs (including through PowerShell and the CLI), and the classic deployment model APIs.

By default, the account that is used to sign up for an Azure subscription is automatically set as both the Account Administrator and the Service

Administrator. They both are authorized to perform subscription management activities, but access to Account Center and creation of new Azure subscriptions and billing changes can be performed only by Account Administrator. There can be only one Account Administrator per account and one Service Administrator per subscription.

Once the subscription has been created, more Co-Administrators can be added. The Co-Administrator has the same level of access as the Service Administrator but cannot change the association of subscriptions to Azure directories. There can be up to 200 Co-Administrators per subscription.

Users assigned with the Service Administrator and Co-Administrator roles have the same access as a user who is assigned the Azure RBAC Owner role at the subscription scope.

In the Azure portal, you can view the current assignments for the Account Administrator and Service Administrator roles by browsing to a subscription in the Azure portal and selecting the **Properties** blade, as seen in [Figure 1-51](#).

Visual Studio Ultimate with MSDN | Properties

Subscription

Search (Ctrl+ /) < Service Admin X

Security

Events

Cost Management

Cost analysis

Cost alerts

Budgets

Advisor recommendations

Billing

Invoices

External services

Payment methods

Partner information

Settings

Programmatic deployment

Resource groups

Resources

Usage + quotas

Policies

Management certificates

My permissions

Resource providers

Deployments

Properties

Resource locks

SUBSCRIPTION ID
00b72028-9dce-4729-9b2a-a10e92054447

STATUS
Active

CURRENT BILLING PERIOD
8/12/2020-9/11/2020

PURCHASE DATE
8/12/2014

CURRENCY
INR

OFFER
MSDN

OFFER ID
MS-AZR-0063P

ACCOUNT ADMIN
[REDACTED]

SERVICE ADMIN
harshulp@outlook.com

FIGURE 1-51 Azure subscription properties

Azure RBAC roles are more flexible than classic administrator roles and allow for more fine-grained access management. Azure RBAC has more than 70 built-in roles, but there are four foundational roles, as shown in [Table 1-4](#).

TABLE 1-4 Azure RBAC roles

Azure RBAC role	Permissions	Notes
Owner	<ul style="list-style-type: none">■ Full access to all resources■ Delegate access to others	<ul style="list-style-type: none">■ The Service Administrator and Co-Administrators are assigned the Owner role at the subscription scope.■ Applies to all resource types.
Contributor	<ul style="list-style-type: none">■ Create and manage all of types of Azure resources■ Cannot grant access to others	<ul style="list-style-type: none">■ Applies to all resource types.
Reader	<ul style="list-style-type: none">■ View Azure resources	<ul style="list-style-type: none">■ Applies to all resource types.

User Access Administrator	<ul style="list-style-type: none"> ▪ Manage user access to Azure resources
---------------------------	---

Configure management groups

Management groups can also be used to apply Azure RBAC to a subscription. Management groups allow you to apply governance consistently across subscriptions, including the application of common RBAC controls and the application of Azure Policy, as discussed later in this chapter.

Management groups allow subscriptions to be organized in a multi-level hierarchy, providing a number of tangible benefits:

- **Reduced overhead.** There is no need to apply governance on every subscription.
- **Enforcement.** Company admins can apply governance at the management group level, outside the control of the subscription admin and the controls implemented at the management group can be applied to both existing and new subscriptions. This eliminates inconsistencies in the application of governance as the same controls are applied the same way to the desired subscriptions.
- **Reporting.** The Azure Policy provides reports of compliance; with management groups the reporting can span across multiple/all subscriptions in an organization.

Management groups form a hierarchy that is up to six levels deep, excluding the root and subscription levels. Each group has exactly one parent group and can have multiple child groups. An example hierarchy is shown in [Figure 1-52](#). In such a hierarchy, one common set of Policy could be applied at the root management group, which all child management groups and subscriptions would inherit. Then, as needed, those children can have additional controls applied.

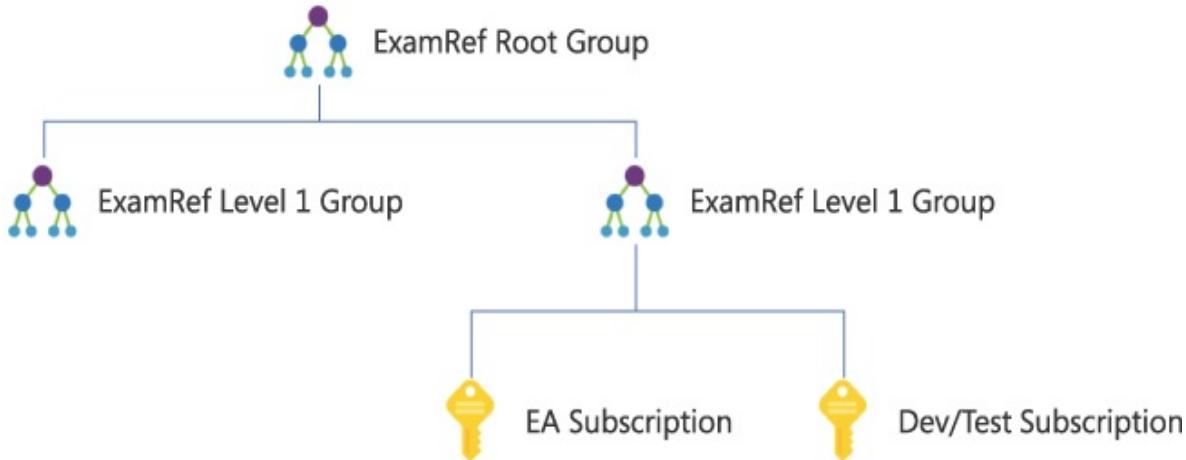


FIGURE 1-52 Example Management group hierarchy

There is a single root management group at the root of the hierarchy. This management group is associated with the Azure AD tenant that is then associated with an Azure subscription. It cannot be moved or deleted. Individual subscriptions, including new subscriptions, are added to a management group.

Like RBAC, Azure Policy is also applied at a specific scope. The scope can be a subscription, a resource group, or an individual resource. For example, when Policy is applied at the subscription scope, it gets inherited to all the resource groups and resources in the subscription, as shown in [Figure 1-53](#).

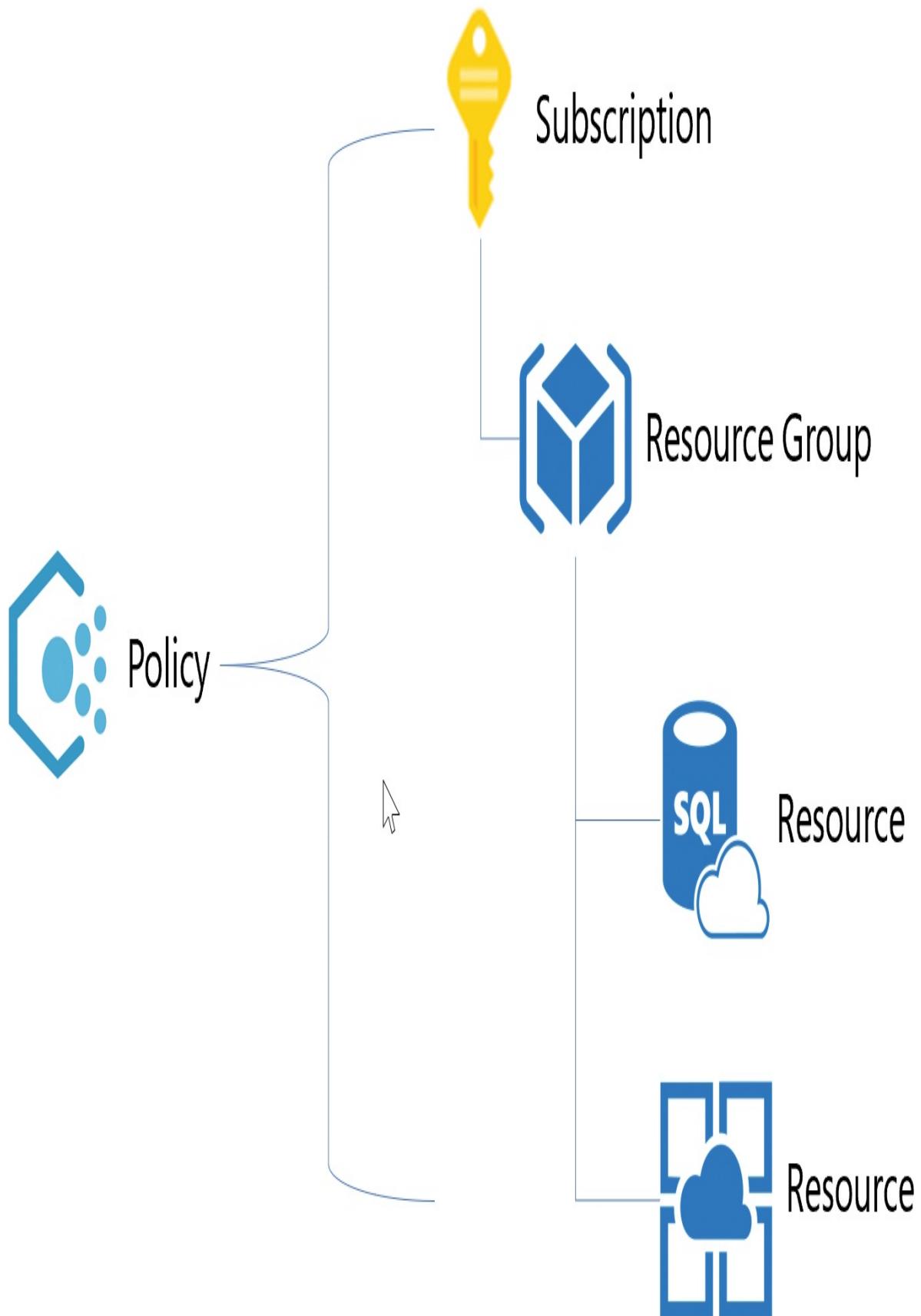


FIGURE 1-53 Example Policy applied at the subscription scope

Management groups introduce an additional scope above a subscription. When applied at the management group scope, each subscription under the management group inherits the RBAC and Policy assignments of the management group as shown in [Figure 1-54](#).

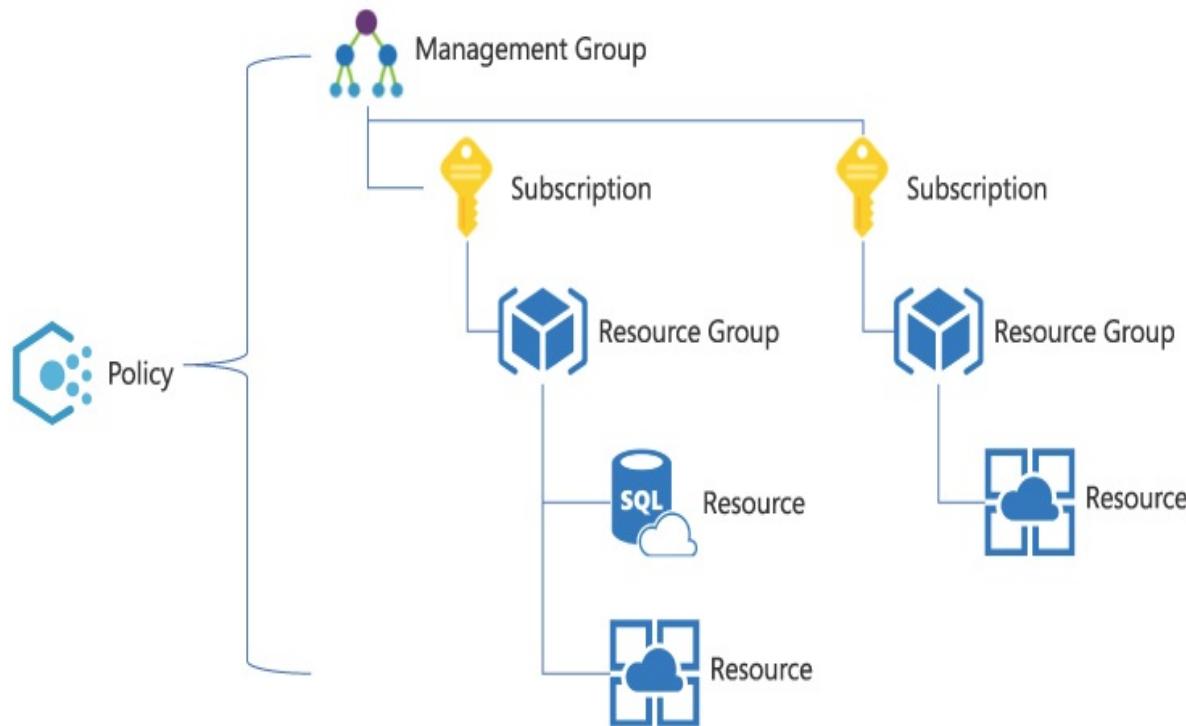


FIGURE 1-54 Example Policy applied at the management group scope

To add a role assignment to a management group, browse to the management groups service in the Azure portal. Select a management group and then click the **Details** button next to that group's name. Select the **Access Control (IAM)** blade and click **Add Role Assignment**, just as you would to an Azure subscription, as shown in [Figure 1-55](#).

ExamRefRG | Access control (IAM)

Management group

 Search (Ctrl+)[Add](#)[Download role assignments](#)[Edit columns](#)[Refresh](#)[Remove](#)[Got feedback?](#)[Overview](#)[Add role assignment](#)[Assignments](#)[Roles](#)[Deny assignments](#)[Classic administrators](#)[Activity Log](#)[Add co-administrator](#)[Access control \(IAM\)](#)[Check access](#)

Review the level of access a user, group, service principal, or managed identity has to this resource. [Learn more](#)

[Find](#) Azure AD user, group, or service principal Search by name or email address[Cost Management](#)[Cost analysis](#)[Budgets](#)[Settings](#)[Hierarchy settings](#)[Deployments](#)[Add a role assignment](#)

Grant access to resources at this scope by assigning a role to a user, group, service principal, or managed identity.

[Add](#)[Learn more](#)[View role assignments](#)

View the users, groups, service principals and managed identities that have role assignments granting them access at this scope.

[View](#)[Learn more](#)[View deny assignments](#)

View the users, groups, service principals and managed identities that have been denied access to specific actions at this scope.

[View](#)[Learn more](#)

FIGURE 1-55 Access control (IAM) blade for an Azure management group

Important RBAC and management groups

RBAC applied at the management group level is inherited by all the child resources within the scope of the management group (subscriptions, resource groups, and resources). For instance, if you add a user as an Owner at the management group scope, that user will become an Owner in all the subscriptions associated with the management group.

Configure cost management

In Azure, there are several types of quotas that are applicable to subscriptions, including resource quotas and spending quotas. With Azure resource quotas (or limits), Azure administrators can view the current consumption and usage of resources within an Azure subscription and understand how that consumption can be affected by Azure resource limits. Administrators can also request quota increases for certain resource types. For instance, the number of cores available for virtual machines is limited to 20 per region by default. This limit can be increased by submitting a request to Microsoft support.

There are also spending quotas in Azure. Spending quotas allow administrators to set alerts within an Azure subscription by configuring budgets to inform the business when their Azure spending has hit a certain threshold. While a resource limit can stop resources from being created (for example, there are not enough cores available to the subscription in the desired region), a spending quota acts as an alerting mechanism and does not stop resources from being created or consumed. While an alert can be generated from a spending quota, resources can still be created and consumed which could cause the spending quota to be exceeded.

Tags in Azure Resource Manager allow consumers of Azure to logically categorize Azure resource groups and Azure resources. As resources are tagged, they can then be queried and tracked based on the associated tags. Tags are a crucial component to implement chargeback within an Azure subscription. For example, in organizations where an Azure subscription is shared by multiple business units or departments, there might be a need to understand how

resources are used for individual departments and show the cost associated with each department, either to bill that department for their Azure consumption (chargeback) or to help that department understand their spend in Azure (showback).

Configure resource quotas

To view the existing resource quotas (or service limits) for your Azure subscription, browse to the Azure subscription in the Azure portal and select the **Usage + Quotas** blade. From this blade, you can view existing quotas by service, resource provider, and location. You also filter the list by resource types you have deployed.

To increase a quota, click the **Request Increase** button, as shown in [Figure 1-56](#).

Home > Subscriptions > Usage + quotas

Usage + quotas

Subscription

Search (Ctrl+ /) Refresh

You can use each Microsoft Azure resource up to its quota. Each subscription has separate quotas and usage is tracked per subscription. If you reach a quota cap, you can request an increase via Help + Support. [Learn more](#)

Request Increase

All service quotas Microsoft.Compute All locations Show only items with usage

Virtual Machines

Quota	Provider	Location	Usage
Virtual Machines	Microsoft.Compute	Central India	0 % 1 of 25000

Cost Management

- Cost analysis
- Budgets
- Advisor recommendations

Billing

- Invoices
- External services
- Payment methods
- Partner information

Settings

- Programmatic deployment
- Resource groups
- Resources
- Usage + quotas
- Policies

FIGURE 1-56 Azure subscription resource quotas

Clicking the **Request Increase** button will begin the process to open a new support request. As a part of the request, you must select the quota type (for example, Compute/VM cores or Machine Learning service) and provide a description of your request.

Important Quota increase

Submitting a request to increase a quota is only submitting a support request to Microsoft. Microsoft Support must respond to the request, and while most requests are granted, it is not guaranteed that a quota increase will be granted.

The consumption of resources within a subscription against a resource quota can also be viewed with PowerShell. There are multiple cmdlets available in the `Az` (formerly `AzureRm`) PowerShell modules for querying per-service quota usage. For example, to view the current usage of vCPU quotas, use the `Get-AzVMUsage`, and to view the current resource usage for the storage service use `Get-AzStorageUsage`.



Exam Tip

In this chapter and throughout the remaining reference, PowerShell cmdlets are referenced using the new `Az` module. You might see examples on the web and in other reference materials that refer to the `AzureRm` cmdlets. The `Az` module can use `AzureRm` aliases with the command `Enable-AzureRmAlias` for compatibility with existing scripts. See <https://docs.microsoft.com/powershell/azure/overview?#about-the-new-az-module> for more detail.

Configure cost center quotas

One of the key factors in managing an Azure subscription is being able to plan for and drive organizational accountability for Azure spend. One of the best ways to drive accountability is to make sure that the consumers of Azure resources understand their cost, including current usage and forecasting future spend based on current resource consumption.

Budgets in Azure Cost Management provide Azure customers subscriptions under many offer types with the ability to proactively manage cost and monitor Azure spend over time at a subscription level.



Exam Tip

The full list of supported accounts and offers for Azure Cost Management can be found at <https://docs.microsoft.com/azure/cost-management/understand-cost-mgt-data>

Budgets are a monitoring mechanism only with set thresholds and notification rules. When a budget threshold is exceeded a notification is triggered but resources continue to run.

To use Budgets with an Azure subscription, that subscription must be a supported offer type as previously stated. Users must have at least read access (Reader rights) to a subscription to view budgets and must have Contributor (or higher) rights to create and manage budgets. There are also specialized roles that can be used to grant principals access to Cost Management data including Cost Management Contributor and Cost Management Reader.

To create a budget in the Azure portal, navigate to **Cost Management + Billing** and then click **Cost Management**. Select a subscription and then click **Budgets**.

Note Subscription budgets

By default, you will be creating a budget at the subscription scope, but budgets can also be created at the management group as well as resource group scope if necessary.

Click **+Add** and in the **Create Budget** blade, enter a budget **Name** and budget **Amount**. You can also change your desired scope by clicking Change scope option. Choose the **Reset Period** (monthly, quarterly, or annual) and an **Expiration Date**. Budgets require at least one **Cost Threshold** (percent of budget) and an email address for the alert recipient. [Figure 1-57](#) shows an example for a monthly budget for \$10,000.

✓ Create a budget 2 Set alerts

Create a budget and set alerts to help you monitor your costs.

BUDGET SCOPING

The budget you create will be assigned to the selected scope. Use additional filters like resource groups to have your budget monitor with more granularity as needed.

Scope



[Change scope](#)

BillingPeriod : 0 selected x

+/- Add filter

BUDGET DETAILS

Give your budget a unique name. Select the time window it analyzes during each evaluation period, its expiration date and the amount.

* Name

FY20



* Reset period ⓘ

Billing month



* Start date ⓘ

2020



April



12

* Expiration date ⓘ

2022



April



11



BUDGET AMOUNT

Give your budget amount threshold

Amount (₹) *

10000



i Suggested budget: ₹1,350 based on forecast.

[Previous](#)

[Next >](#)

FIGURE 1-57 Azure budgets

Figure 1-58 shows a threshold set at 90 percent of the budget (\$9,000).

✓ Create a budget ✓ Set alerts

Configure alert conditions and send email notifications based on your spend.

* Alert conditions

% Of budget	Amount	Action group	Action group type
90 	9000	None 	
Enter %	-	None 	

[Manage action group](#) ⓘ

* Alert recipients (email)

Alert recipients (email)

admin@examref.com  

example@email.com

It is recommended to add azure-noreply@microsoft.com to your email white list to ensure alert mails do not go to your spam folder.

FIGURE 1-58 Azure budget alerts

Note Budget alerts

Budget alerts can also leverage the same Action Groups that Azure Monitor supports. Action groups are a collection of notification preferences and are discussed in detail in [Chapter 5](#).

After your budgets have been created, they can be viewed through the **Budgets** blade. When viewing the subscription scope, you will see the budgets for both the subscription and any resource group scoped budgets in a single view, as shown in [Figure 1-59](#).

The screenshot shows the Azure Budgets interface. At the top, there are buttons for '+ Add', 'Refresh', and 'Help'. Below the header, there are search fields for 'Scope' (set to 'Visual Studio Ultimate with MSDN') and 'Search by name', and a dropdown for 'All periods'. A message box at the bottom left informs users that budget evaluations now include reserved instance and purchase charges, with a link to documentation. The main table lists one budget entry:

Name	Scope	Reset period	Start date	End date	Budget	Evaluated spend	Progress
FY20	00b72028-9dce-472...	BillingMonth	4/12/2020	4/11/2022	10000	N/A	0.00%

FIGURE 1-59 Azure budgets

Monitor and report spend

While Azure Advisor and its cost recommendations provide one method for monitoring spend and unused resources, Azure has many other tools that can help you monitor the cost of your resources and report on that cost.

There are several considerations that you must account for when reporting on the cost associated with your Azure resources:

- Azure services are available to customers in 140 countries worldwide.
- Billing is supported across 24 major currencies.

- Azure subscriptions are billed monthly. If you are paying by credit card, note that pre-paid cards and virtual credit cards are not accepted.
- You can also pay for Azure by monthly invoice. To apply for invoice payment, raise an appropriate billing support ticket from the Azure management portal. Processing the request takes 5-7 days, depending on the time required for the necessary credit checks. Invoice payment is only available to business customers, and once a subscription has been moved to invoice payment, it cannot be moved back to credit card payment. If you choose invoice payment, you will get an invoice, and you will pay with a wire transfer or check.
- Customers on an Enterprise Agreement (EA) can add up-front commitments to Azure and then create multiple subscriptions under the agreement, which draw from the monetary commitment.
 - EA commitments are billed immediately, and then consumed throughout the year against the Azure resources consumed.
 - If the committed spend is exceeded, the extra spend, or “overage,” is billed at the same discounted EA rate. Billing for overage is annual if the overspend is under 50 percent of the commitment, or quarterly if over 50 percent.
- Azure Marketplace third party services are billed separately with a potentially different billing period, separate invoice, and separate credit card charge. Each service has its own billing model, which will be described in the Azure portal at the time of purchase. These range from pay-as-you-go per-minute billing to fixed monthly charges. Some services also offer a “bring your own license” model, which must provide a license purchased separately prior to using the service.

There are three portals that are used to manage Azure subscriptions that are relevant for billing and cost management. They are:

- The EA Portal available at <https://ea.azure.com>. This is available only to customers with an Enterprise Agreement and is used for managing spend across one or more subscriptions.
- The Account Portal at <https://account.azure.com/subscriptions>. This is available for all subscriptions and accessible by Account owners. It is used to manage subscriptions, payment methods, and spending limits.
- The Azure portal at <https://portal.azure.com>. This is available for all

subscriptions and includes Azure Cost Management.

The EA Portal can be used to monitor spend across multiple subscriptions with the ability to view costs by the entire organization or by the business unit. Organizations can view historical spending, broken out by commitment, and overage or third-party Azure Marketplace consumption (see [Figure 1-60](#)). They can also download their current price sheet to see their EA discount rates, which often differ from the public pricing shown in the Azure portal and in the pricing calculator.

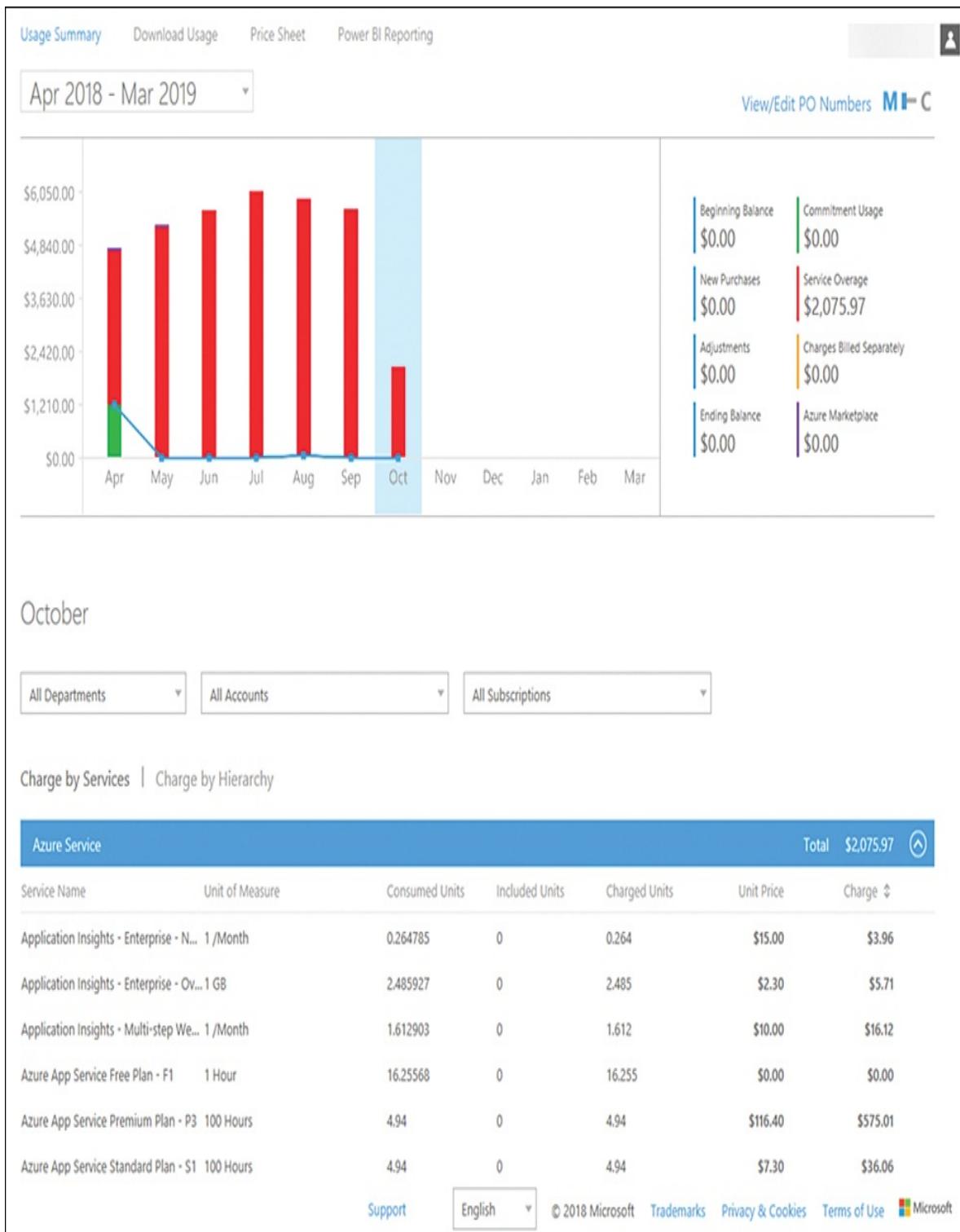


FIGURE 1-60 Azure EA Portal

EA customers can create spending quotas and set notification thresholds

through the EA Portal. This is in addition to the budget alerts available through Azure Cost Management and Billing Alerts found in the Account Portal. An advantage of using the EA portal to configure spending notifications is that a quota alert can be triggered based on aggregate spending across all the subscriptions within a department. Cost centers can be assigned to the departments that accounts and subscriptions roll up to for EA customers, making it easier to track cost by business unit and operate a showback or chargeback model.

Within the Azure portal, EA customers can also use Azure Cost Management for tracking cost for individual subscriptions. Cost Management includes features for performing cost analysis, setting per-subscription budgets and alerts, setting recommendations for optimization, and exporting cost management data to perform deeper analysis.

Access to the Cost Management service is dictated by scopes. A user must have at least read access to one of the following scopes shown in [Table 1-5](#) to view data in Cost Management.

TABLE 1-5 Cost Management access scopes

Scope	Defined at	Required access to view data	Prerequisite EA setting	Consolidates data to
Billing account	https://ea.azure.com	Enterprise Admin	None	All subscriptions from the enterprise agreement
Department	https://ea.azure.com	Department Admin	DA view charges enabled	All subscriptions belonging to an enrollment account that is linked to the department
Enrollment account	https://ea.azure.com	Account Owner	AO view charges enabled	All subscriptions from the enrollment account
Management	https://portal.azure.com	Cost Management	AO view charges	All subscriptions below the management group

group	https://portal.azure.com	Reader (or Reader)	enabled	
Subscription	https://portal.azure.com	Cost Management Reader (or Reader)	AO view charges enabled	All resources/resource groups in the subscription
Resource group	https://portal.azure.com	Cost Management Reader (or Reader)	AO view charges enabled	All resources in the resource group

To access Cost Management, in the Azure portal browse to **Cost Management + Billing** and choose **Cost Management**. Finally, select **Cost Analysis**, as shown in [Figure 1-61](#).

Home > Cost Management + Billing > Cost Management: Visual Studio Ultimate with MSDN | Cost analysis

Cost Management: [REDACTED] | Cost analysis

Subscription

Search (Ctrl+ /) Save Save as Delete view Share Refresh Export Cost by resource Settings Try preview Help

Scope: [REDACTED] VIEW Accumulated costs Invoice (Azure) : Apr 12-May 11 Add filter

ACTUAL COST (INR ONLY) FORECAST: CHART VIEW ON BUDGET: FY20
₹722.02 ₹1,306.92 ₹10,000 Group by: None Granularity: Accumulated Area

Accumulated cost: ₹217.15 Forecast cost: -

Accumulated cost Forecast cost

Service name: storage Location: in central Resource group name: poc-rq

Storage cost: ₹722.02 Location cost: ₹722.02 Resource group cost: ₹722.01

Cloud shell storage cost: ₹0.01

FIGURE 1-61 Azure Cost Management Cost Analysis

If you have access to more than one scope, you can filter by scope and begin interacting with the data. From cost analysis, you can view the total costs for the current month, view the budget (if available), set the granularity (Accumulated, Daily, or Monthly), and apply the filters. You can filter by Department Name, Enrollment Account Name, Location, Meter, Meter Category, Meter Subcategory, Resource, Resource Group Name, Resource Type, Service Name, Service Tier, Subscription ID, Subscription Name, and Tag.

The data in a view can be downloaded from Cost Analysis as a CSV. Any filtering that you have applied, including groupings, are applied to the file.

Thought experiment

In this thought experiment, apply what you have learned. You can find answers to these questions in the next section.

You are responsible for creating and tracking resources in Azure for two business units within your organization: HR and Marketing. Your organization has an Enterprise Agreement (EA). Each business unit needs to deploy its own resources. Your Finance department needs to be able to understand the consumption of resources for each business unit for chargeback purposes. Finance would also like to be able to receive a notification when a defined monetary threshold is reached for each business unit.

The resources that each business unit will deploy are from a known set of resources and users should be prevented from creating unapproved resources. There will be resources within a subscription that are not billed back directly to the business units, but will be billed to IT. These resources must be differentiated for Finance.

1. How will you ensure that users can only create approved resources in Azure?
2. How will you grant access to create resources and restrict each business unit's users from impacting the other business units?
3. How will Finance access billing data for Azure and how will they be able to tell where each cost is coming from?
4. How will Finance be notified when each business unit is nearing their

spending threshold?

Thought experiment answers

This section contains the solution to the thought experiment for the chapter.

For each business unit, HR and Marketing, a separate subscription can be created. This will allow for the separation of resources by business unit and allow for segregated and aggregated cost reporting and monitoring for Finance through the EA Portal.

1. To ensure users can only create approved resources, policies should be defined that can be assigned to each subscription. The policies will deny the creation of any unapproved resources and compliance can be monitored through Azure Policy as well.
2. Each business unit will be placed into its own subscription. Within a subscription, resource groups will be created, and users will be granted appropriate rights at the resource group level. As RBAC is inherited by child resources, with the appropriate rights granted, users will be able to create and manage resources as needed without impact others in the subscription. This will be layered with Azure Policy to ensure that only allowed resources can be created. This can be extended further by creating Azure Resource Manager templates, which can be used by business unit users to deploy their resources with well-known configurations.

Alternatively, you can also use management groups to segregate the business units. You can still use RBAC to inherit the access subscription and child resources from a management group.

3. Users in the Finance department can be granted access to the EA Portal and/or Cost Azure Cost Management by configuring access through the required scopes. To make sure that they can tell where each resource cost is coming from, tags should be applied to all resources using a taxonomy defined by Finance. For example, “BusinessUnit” can be a tag with the allowed values “HR,” “Marketing,” and “IT.” That taxonomy should be governed through Azure Policy to ensure that all resources are tagged with required and valid tags.
4. To manage thresholds, Department quotas can be configured in the EA Portal. In addition, Budgets can be created in Cost Management. Budgets

in Cost Management can provide more flexibility has multiple notification thresholds can be set and each notification can have a different receiver. This would allow a single budget to send notifications to both business unit owners and Finance.

Chapter summary

Here are some of the key takeaways from this chapter:

- Windows 10 can be added to Azure AD as a device to be managed, enabling BYOD or corporate cloud only deployments with Azure AD Join.
- Azure AD Join enables administrators to manage device identity independently of users. For example, dynamic security groups can be created based on device attributes and then conditional access policies could be applied to those groups.
- Downstream Windows clients can be managed through Azure AD using Azure AD hybrid join.
- Conditional access is a feature of Azure AD which allows administrators to control access to cloud applications through additional checks such as user location, the device the user is accessing the cloud app from, and more.
- Multiple Azure AD tenants can be created and managed through Azure. This includes creating new directories and deleting existing directories.
- Users and groups can be created through the Azure portal, Azure PowerShell, the Azure CLI, and the Graph API.
- Users and groups can be managed in bulk with tools like PowerShell.
- Self-service password reset can be combined with the password writeback features of Azure AD Connect to allow users to reset their passwords from the cloud while adhering to on-premises password standards.
- Many advanced features of Azure AD require Azure AD Premium P1 or Azure AD Premium P2 licenses. When considering Azure AD features, administrators need to be aware of the licensing boundaries.
- Azure offers a rich ecosystem of governance controls with user-level and platform-level controls in the form of role-based access control (RBAC)

and Azure Policy.

- Azure management groups can be used to control Policy and RBAC for multiple subscriptions. Management groups enable organizational alignment for your Azure subscriptions through custom hierarchies and groupings.
- Tags in Azure can be used to logically organize resources by categories. Each tag is a name and a value pair. Tags can be shared across multiple resources and enforced with Azure Policy.
- Azure Policy is a service that lets you create, manage, and apply Policy to Azure resources at a subscription, resource group, or resource level. Policies enforce different rules over your Azure resources, so those resources remain compliant with your organization's standards.
- Role-based access control allows you to grant users, groups, and service principals access to Azure resources at the subscription, resource group, or resource scopes with RBAC inheritance. The three core roles are Owner, Contributor, and Reader.
- You can create resources from the portal, PowerShell, the CLI tools, and Azure Resource Manager templates. You should understand when to use which tool and how to configure the resource during provisioning and after provisioning.
- A resource is simply a single service instance in Azure. Most services in Azure can be represented as a resource. For example, a Web App instance is a resource. An App Service Plan is also a resource. Even a SQL Database instance is a resource.
- A resource group is a logical grouping of resources. For example, a Resource Group where you deploy a VM compute instance may be composed of a Network Interface Card (NIC), a Virtual Machine, a Virtual Network, and a Public IP Address.
- A resource group template is a JSON file that allows you to declaratively describe a set of resources. These resources can then be added to a new or existing resource group. For example, a template can contain the configuration necessary to create two API App instances, a Mobile App instance, and a Document DB instance.
- A template can simplify orchestration because you only need to deploy the template to deploy all your resources.

- A template allows you to configure multiple resources simultaneously and use variables/parameters/functions to create dependencies between resources.

Chapter 2

Implement and manage storage

Implementing and managing storage is one of the most important aspects of building or deploying a new solution using Azure. There are several services and features available for use, and each has its own place. Azure Storage is the underlying storage for most of the services in Azure. It provides service for the storage and retrieval of files, and it has services that are available for storing large volumes of data through tables. Also, Azure Storage includes a fast and reliable messaging service for application developers with queues. In this chapter, we review how to implement and manage storage with an emphasis on Azure Storage.

Also, we discuss related services such as Import/Export, Azure Files, and many of the tools that simplify the management of these services.

Skills covered in this chapter:

- [Skill 2.1: Secure Storage](#)
- [Skill 2.2: Manage Storage](#)
- [Skill 2.3: Configure Azure Files and Azure Blob Storage](#)

Skill 2.1: Secure Storage

An Azure Storage account is an entity you create that is used to store Azure Storage data objects such as blobs, files, queues, tables, and disks. Data in an Azure Storage account is durable and highly available, secure, massively scalable, and accessible from anywhere in the world over HTTP or HTTPS.

This section covers how to:

- [Configure network access to storage accounts](#)
- [Create and configure storage accounts](#)

- Generate shared access signatures
- Manage access keys
- Configure Azure AD Authentication for a storage account

Configure network access to the storage accounts

Storage accounts are managed through Azure Resource Manager. Management operations are authenticated and authorized using Azure Active Directory and RBAC. Each storage account service exposes its own endpoint used to manage the data in that storage service (blobs in Blob Storage, entities in tables, and so on). These service-specific endpoints are not exposed through Azure Resource Manager; instead, they are (by default) Internet-facing endpoints.

Access to these Internet-facing storage endpoints must be secured, and Azure Storage provides several ways to do so. In this section, we will review the network-level access controls: the storage firewall and service endpoints. We also discuss Blob Storage access levels. The following sections then describe the application-level controls: shared access signatures and access keys. In later sections, we also discuss Azure Storage replication and how to leverage Azure AD authentication for a storage account.

Storage firewall

The storage firewall allows you to limit access to specific IP addresses or an IP address range. It applies to all storage account services (blobs, tables, queues, and files). For example, by limiting access to the IP address range of your company, access from other locations will be blocked. Service endpoints are used to restrict access to specific subnets within an Azure VNet.

To configure the storage firewall using the Azure portal, open the storage account blade and click **Firewalls And Virtual Networks**. Under **All Access From**, click **Selected Networks** to reveal the **Firewall** and **Virtual Network** settings, as shown in [Figure 2-1](#).

Save Discard Refresh

Allow access from

All networks Selected networks

i Configure network security for your storage accounts. [Learn more](#)

Virtual networks

Secure your storage account with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

Virtual Network	Subnet	Address range	Endpoint Status	Resource Group	Subscription
virtualNetwork1	1			rgCoreNetwork	Visual Studio ...
	subnet1	10.2.0.0/24	✓ Enabled	rgCoreNetwork	Visual Studio ...

Firewall

Add IP ranges to allow access from the internet or your on-premises networks. [Learn more](#).

Add your client IP address (99.245.212.142)

Address range

32.54.231.0/24



IP address or CIDR

Exceptions

Allow trusted Microsoft services to access this storage account

Allow read access to storage logging from any network

Allow read access to storage metrics from any network

FIGURE 2-1 Configuring a storage account firewall and virtual network service endpoint access

When accessing the storage account via the Internet, use the storage firewall to specify the Internet-facing source IP addresses (for example, 32.54.231.0/24, as shown in [Figure 2-1](#)) that will make the storage requests. All Internet traffic is denied, except the defined IP addresses in the storage firewall. You can specify a list of either individual IPv4 addresses or IPv4 CIDR address ranges. (CIDR notation is explained in the chapter on Azure Networking.)

The storage firewall includes an option to allow access from trusted Microsoft services. These services include Azure Backup, Azure Site Recovery, and Azure Networking. For example, it will allow access to storage for NSG flow logs if the **Allow Trusted Microsoft Services To Access This Account** exceptions checkbox is selected (see [Figure 2-1](#)). It will also allow read-only access to storage metrics and logs.

Note Address space for Storage firewall

When creating a storage firewall, you must use public Internet IP address space. You cannot use IPs in the private IP address space.

Virtual network service endpoints

In some scenarios, a storage account is only accessed from within an Azure virtual network. In this case, it is desirable from a security standpoint to block all Internet access. Configuring virtual network service endpoints for your Azure Storage accounts allows you to remove access from the public Internet and only allow traffic from a virtual network for improved security.

Another benefit of using service endpoints is optimized routing. Service endpoints create a direct network route from the virtual network to the storage service. If forced tunneling is being used to force Internet traffic to your on-premises network or to another network appliance, requests to Azure Storage will follow that same route. By using service endpoints, you can use direct route to the storage account instead of the on-premises route, so no additional latency is incurred.

Configuring service endpoints requires two steps. First, from the virtual

network subnet, choose **Microsoft.Storage** from the **Service Endpoints** drop-down menu. This creates the route from the subnet to the storage service but does not restrict which storage account the virtual network can use. To update the subnet settings, you should choose **virtualNetwork1** from the **Virtual Networks** blade. Then go to **Subnets** in the left pane under **Settings**. Click **Subnet1** to access the subnet settings. [Figure 2-2](#) shows the subnet settings, including the service endpoint configuration.

The second step is to configure which virtual networks can access a particular storage account. From the storage account blade, click **Firewalls And Virtual Networks**. Under **All Access From**, click **Selected Networks** to reveal the **Firewall** and **Virtual Network** settings, as shown previously in [Figure 2-1](#). Under **Virtual Networks**, select the virtual networks and subnets that should have access to this storage account.

 **subnet1**
virtualNetwork1

 Save  Discard  Delete  Refresh

Address range (CIDR block) * ⓘ

10.2.0.0/24

10.2.0.0 - 10.2.0.255 (256 addresses)

Available addresses ⓘ

251

NAT gateway ⓘ

None

Add IPv6 address space

Network security group

None

Route table

None

Users



Manage users

Service endpoints

Services ⓘ

Microsoft.Storage

Service

Status

Microsoft.Storage

Succeeded

...

Subnet delegation

Delegate subnet to a service ⓘ

None

FIGURE 2-2 Configuring a subnet with a service endpoint for Azure Storage

Blob Storage access levels

Storage accounts support an additional access control mechanism that is limited only to Blob Storage. By default, no public read access is enabled for anonymous users, and only users with rights granted through RBAC or with the storage account name and key will have access to the stored blobs. To enable anonymous user access, you must change the container access level (see [Figure 2-3](#)). The supported levels are as follows:

- **Private.** With this option, only the storage account owner can access the container and its blobs. No one else would have access to them.
- **Blob.** With this option, only blobs within the container can be accessed anonymously.
- **Container.** With this option, blobs and their containers can be accessed anonymously.

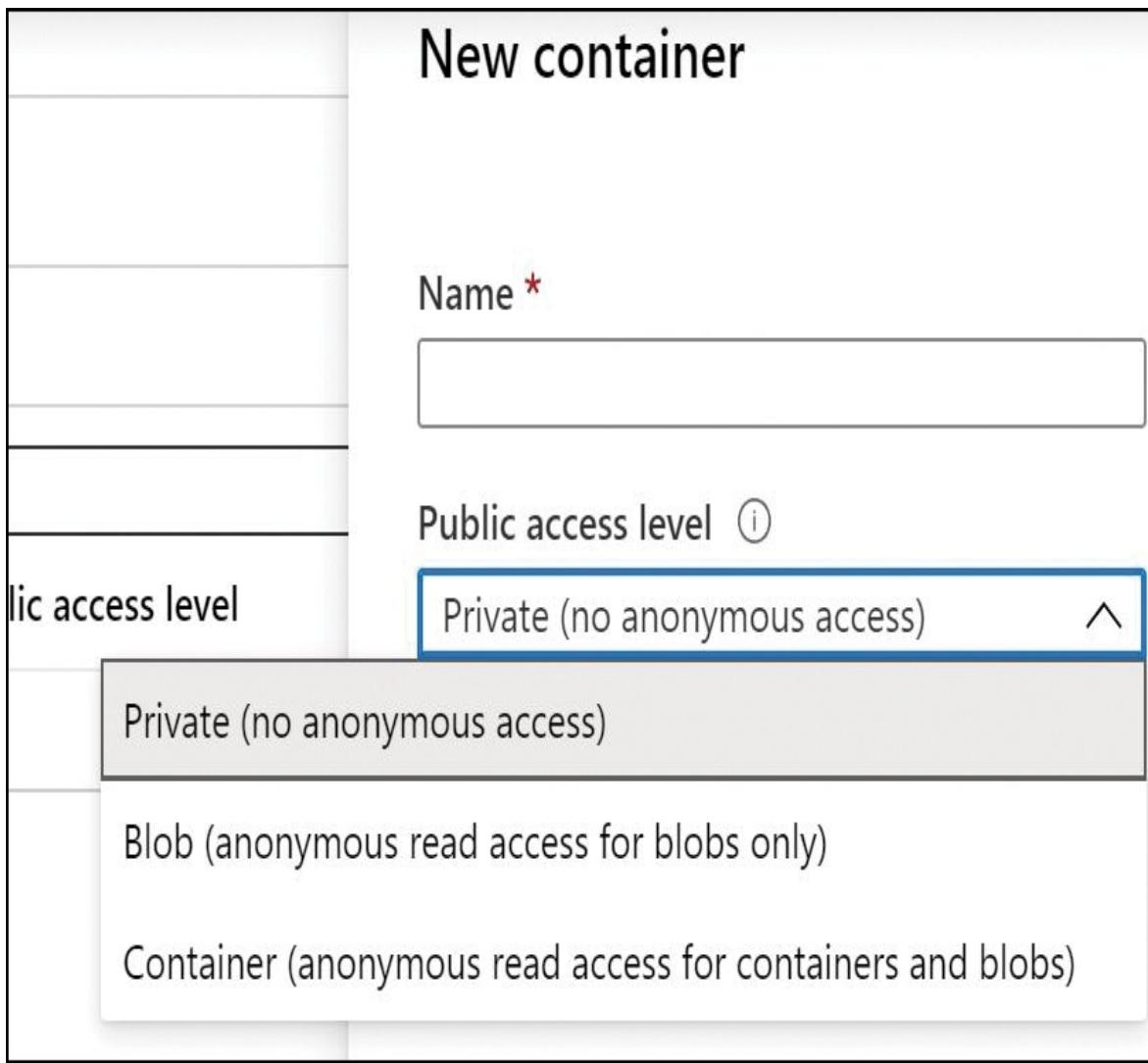


FIGURE 2-3 Blob Storage access levels

You can change the access level through the Azure portal, Azure PowerShell, Azure CLI, programmatically using the REST API, or by using Azure Storage Explorer. The access level is configured separately on each blob container.

A shared access signature token (SAS token) is a URI query string parameter that grants access to specific containers, blobs, queues, and tables. Use an SAS token to grant access to a client that should not have access to the entire contents of the storage account (and therefore, should not have access to the storage account keys) but still requires secure authentication. By distributing an SAS URI to these clients, you can grant them access to a specific resource, for a specified period of time, and with a specified set of permissions. Frequently,

SAS tokens are used to read and write the data to users' storage accounts. Also, SAS tokens are widely used to copy blobs or files to another storage account.

Note SAS tokens using HTTPS

When dealing with SAS tokens, you must use only the HTTPS protocol. Because active SAS tokens provide direct authentication to your storage account, you must use a secure connection, such as HTTPS, to distribute SAS token URIs.

Create and configure storage accounts

Azure Storage accounts provide a cloud-based storage service that is highly scalable, available, performant, and durable. Within each storage account, a number of separate storage services are provided:

- **Blobs.** Provides a highly scalable service for storing arbitrary data objects such as text or binary data.
- **Tables.** Provides a NoSQL-style store for storing structured data. Unlike a relational database, tables in Azure storage do not require a fixed schema, so different entries in the same table can have different fields.
- **Queues.** Provides reliable message queueing between application components.
- **Files.** Provides managed file shares that can be used by Azure VMs or on-premises servers.
- **Disk.** Provides a persistent storage volume for Azure VM which can be attached as a virtual hard disk.

There are three types of storage blobs: Block Blobs, Append Blobs, and Page Blobs. Page Blobs are generally used to store VHD files when deploying unmanaged disks. (Unmanaged disks are an older disk storage technology for Azure virtual machines. Managed disks are recommended for new deployments.)

When creating a storage account, there are several options that must be set: Performance Tier, Account Kind, Replication Option, and Access Tier. There are some interactions between these settings. For example, only the Standard

performance tier allows you to choose the access tier. The following sections describe each of these settings. We then describe how to create storage accounts using the Azure portal, PowerShell, and Azure CLI.

Naming storage accounts

While naming an Azure Storage Account, you need to remember these points:

- The storage account name must be unique across all existing storage account names in Azure.
- The name must be between 3 to 24 characters and can contain only lowercase letters and numbers.

Performance tiers

When creating a storage account, you must choose between the Standard and Premium performance tiers. This setting cannot be changed later.

- **Standard.** This tier supports all storage services: blobs, tables, files, queues, and unmanaged Azure virtual machine disks. It uses magnetic disks to provide cost-efficient and reliable storage.
- **Premium.** This tier is designed to support workloads with greater demands on I/O and is backed by high-performance SSD disks. It only supports General-Purpose accounts with Disk Blobs and Page Blobs. It also supports Block Blobs or Append Blobs with BlockBlobStorage accounts and files with FileStorage accounts.

Note Replication options with premium tier

Premium tier only supports LRS as a replication option for general-purpose storage accounts. It supports LRS and ZRS, both for BlockBlobStorage and FileStorage accounts.

Account kind

There are three possible values for the Standard tier: StorageV2 (General-Purpose V2), Storage (General-Purpose V1), and BlobStorage. There are four possible values for the Premium tier: StorageV2 (General-Purpose V2), Storage (General-Purpose V1), BlockBlobStorage, and FileStorage. [Table 2-1](#) shows the

features for each kind of account. Key points to remember are as follows:

- The Blob Storage account is a specialized storage account used to store Block Blobs and Append Blobs. You can't store Page Blobs in these accounts; therefore, you can't use them for unmanaged disks.
- Only General-Purpose V2 and Blob Storage accounts support the Hot, Cool, and Archive access tiers.

General-Purpose V1 and Blob Storage accounts can both be upgraded to a General-Purpose V2 account. This operation is irreversible. No other changes to the account kind are supported.

TABLE 2-1 Storage account types and their supported features

	General-Purpose V2	General-Purpose V1	Blob Storage	Block Blob Storage	File Storage
Services supported	Blob, File, Queue, Table	Blob, File, Queue, Table	Blob (Block Blobs and Append Blobs only)	Blob (Block Blobs and Append Blobs only)	File only
Unmanaged Disk (Page Blob) support	Yes	Yes	No	No	No
Supported Performance Tiers	Standard, Premium	Standard, Premium	Standard	Premium	Premium
Supported Access Tiers	Hot, Cool, Archive	N/A	Hot, Cool, Archive	N/A	N/A
Replication Options	LRS, ZRS, GRS, RA-GRS, GZRS, RA-	LRS, GRS, RA-	LRS, GRS, RA-GRS	LRS, ZRS	LRS, ZRS

GZRS	GRS			
------	-----	--	--	--

Replication options

When you create a storage account, you can also specify how your data will be replicated for redundancy and resistance to failure. There are four options, as described in [Table 2-2](#).

TABLE 2-2 Storage account replication options

Replication Type	Description
Locally redundant storage (LRS)	Makes three synchronous copies of your data within a single datacenter. Available for General-Purpose or Blob Storage accounts at both the Standard and Premium Performance tiers.
Zone redundant storage (ZRS)	Makes three synchronous copies to three separate availability zones within a single region. Available for General-Purpose V2 storage accounts only, at the Standard Performance tier only. Also available for BlockBlobStorage and FileStorage.
Geographically redundant storage (GRS)	This is the same as LRS (three local copies), plus three additional asynchronous copies to a second datacenter hundreds of miles away from the primary region. Data replication typically occurs within 15 minutes, although no SLA is provided. Available for General-Purpose or Blob Storage accounts, at the Standard Performance tier only.
Read access geographically redundant storage (GZRS)	This has the same capabilities as GRS, plus you have read-only access to the data in the secondary datacenter.

Locally redundant storage (RA-GRS)	Available for General-Purpose or Blob Storage accounts, at the Standard Performance tier only.
Geographically redundant storage (GZRS)	This is the same as ZRS (three synchronous copies across multiple availability zones), plus three additional asynchronous copies to a second datacenter hundreds of miles away from the primary region. Data replication typically occurs within 15 minutes, although no SLA is provided.
	Available for General-Purpose v2 storage accounts only, at the Standard Performance tier only.
Read access geographically redundant storage (RA-GZRS)	This has the same capabilities as GZRS, plus you have read-only access to the data in the secondary datacenter.
	Available for General-Purpose V2 storage accounts only at the Standard Performance tier only.

Note Replication options

These replication options control the level of durability and availability of the storage account. When the entire datacenter is unavailable, LRS would incur an outage. If the primary region is unavailable, both the LRS and ZRS options would incur an outage, but the GRS and GZRS options would still provide the secondary region that takes care of the requests during the outage. However, not all the replication options are available in all regions. You can find supported regions with these replication options at <https://docs.microsoft.com/azure/storage/common/storage-redundancy>.

Note Specifying replication and performance tier settings

When creating a storage account via the Azure portal, the replication and performance tier options are specified using separate settings. When creating an account using Azure PowerShell, the Azure CLI, or via a template, these settings are combined within the SKU setting.

For example, to specify a Standard storage account using locally redundant storage using the Azure CLI, use `--sku Standard_LRS`.

Access tiers

Azure Blob Storage supports three access tiers: Hot, Cool, and Archive. Each represents a trade-off of performance, availability, and cost. There is no trade-off on the durability (probability of data loss), which is extremely high across all tiers.

Note Blob Storage only

Access tiers apply to Blob Storage only. They do not apply to other storage services, including Block Blob Storage.

The tiers are as follows:

- **Hot.** This access tier is used to store frequently accessed objects. Relative to other tiers, data access costs are low while storage costs are higher.
- **Cool.** This access tier is used to store large amounts of data that is not accessed frequently and that is stored for at least 30 days. The availability SLA is lower than for the Hot tier. Relative to the Hot tier, data access costs are higher and storage costs are lower.
- **Archive.** This access tier is used to archive data for long-term storage, that is accessed rarely, can tolerate several hours of retrieval latency, and will remain in the Archive tier for at least 180 days. This tier is the most cost-effective option for storing data, but accessing that data is more expensive than accessing data in the Hot or Cool tiers.

New blobs will default to the access tier that is set at the storage account level, though you can override that at the blob level by setting a different access tier, including the archive tier.

Note ARCHIVE TIER supportability

Currently, the archive tier is not supported for ZRS, GZRS, or RA-GZRS accounts.

Creating an Azure Storage account

To create a storage account by using the Azure portal, first click **Create A Resource** and then select **Storage**. Next, click **Storage Account**, which will open the **Create Storage Account** blade (see [Figure 2-4](#)). You must choose a unique name for the storage account name. Storage account names must be globally unique and may only contain lowercase characters and digits. Select the Azure region (Location), the performance tier, the kind of storage account, the replication mode, and the access tier. The blade adjusts based on the settings you choose so that you cannot select an unsupported feature combination.

Create storage account

Basics Networking Advanced Tags Review + create

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.

[Learn more about Azure storage accounts ↗](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

Create new

Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

Storage account name * ⓘ

examref104

Location *

(Canada) Canada East

Performance ⓘ

Standard Premium

Account kind ⓘ

StorageV2 (general purpose v2)

Replication ⓘ

Read-access geo-redundant storage (RA-GRS)

Access tier (default) ⓘ

Cool Hot

[Review + create](#)

< Previous

Next : Networking >

FIGURE 2-4 Creating an Azure storage account using the Azure portal

The **Networking** tab of the **Create Storage Account** blade is shown in [Figure 2-5](#). This tab allows us to maintain storage account access either publicly by choosing **Public Endpoint (Selected Networks)** or privately by choosing **Private Endpoint**.

The **Advanced** tab of the **Create Storage Account** blade is shown in [Figure 2-6](#). This tab allows you to specify whether SSL is required for accessing objects in storage; disabling or enabling Azure Files support; choosing data protection options such as blob Soft Delete or Versioning; and for enabling Data Lake Storage integration. Additionally, clicking the **Tags** tab allows you to specify tags on the storage account resource.

Create storage account

Basics Networking Advanced Tags Review + create

Network connectivity

You can connect to your storage account either publicly, via public IP addresses or service endpoints, or privately, using a private endpoint.

Connectivity method *

- Public endpoint (all networks)
- Public endpoint (selected networks)
- Private endpoint

 All networks will be able to access this storage account.

[Learn more about connectivity methods](#) 

FIGURE 2-5 The networking properties that can be set when creating an Azure Storage account using the portal

More Info Creating a Storage Account with PowerShell

You can learn more about the additional parameters at
<https://docs.microsoft.com/en-us/powershell/module/az.storage/new-azstorageaccount>.

More Info Creating a storage account with the Azure CLI

You can learn more about the additional parameters at
<https://docs.microsoft.com/cli/azure/storage/account#az-storage-account-create>.

Generate shared access signatures

There are few different ways you can create an SAS token. An SAS token is a way to granularly control how a client can access data in Azure storage account. You can also use an account-level SAS to access the account itself. You can control many things, such as what services and resources the client can access, what permission the client has, how long the token is valid for, and more.

Create storage account

Basics Networking Advanced Tags Review + create

Security

Secure transfer required ⓘ Disabled Enabled

Azure Files

Large file shares ⓘ Disabled Enabled

Data protection

Blob soft delete ⓘ Disabled Enabled

Versioning ⓘ Disabled Enabled

i The current combination of subscription, storage account kind, performance, replication and location does not support versioning.

Data Lake Storage Gen2

Hierarchical namespace ⓘ Disabled Enabled

NFS v3 ⓘ Disabled Enabled

i Sign up is currently required to utilize the NFS v3 feature on a per-subscription basis. [Sign up for NFS v3 ↗](#)

[Review + create](#)

< Previous

Next : Tags >

FIGURE 2-6 The advanced properties that can be set when creating an Azure Storage account using the Azure portal

In this section, we examine how to create SAS tokens using various methods. The simplest way to create one is by using the Azure portal. Browse to an Azure storage account and open the **Shared Access Signature** blade (see [Figure 2-7](#)). You can check the services, resource types, and permissions based on specific requirements, along with the duration for the SAS token validity and the IP addresses that are providing access. Lastly, you have an option to choose which key you want to use as the signing key for this token.

Home > examref | Shared access signature

examref | Shared access signature

Storage account

Search (Ctrl+ /)

A shared access signature (SAS) is a URI that grants restricted access rights to Azure Storage resources. You can provide a shared access signature to clients who should not be trusted with your storage account key but whom you wish to delegate access to certain storage account resources. By distributing a shared access signature URI to these clients, you grant them access to a resource for a specified period of time.

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Private endpoint connections

Advanced security

Static website

Properties

Locks

Export template

An account-level SAS can delegate access to multiple storage services (i.e. blob, file, queue, table). Note that stored access policies are currently not supported for an account-level SAS.

[Learn more](#)

Allowed services ⓘ

Blob File Queue Table

Allowed resource types ⓘ

Service Container Object

Allowed permissions ⓘ

Read Write Delete List Add Create Update Process

Blob versioning permissions ⓘ

Start and expiry date/time ⓘ

Start: 05/06/2020 10:32:41 PM

End: 05/07/2020 6:32:41 AM

(UTC-05:00) Eastern Time (US & Canada)

Allowed IP addresses ⓘ

10.0.0.3

Allowed protocols ⓘ

HTTPS only HTTPS and HTTP

Signing key ⓘ

key1

Generate SAS and connection string

FIGURE 2-7 Creating a shared access signature using the Azure portal

Once the token is generated, it will be listed along with connection string and SAS URLs, as shown in [Figure 2-8](#).

Connection string

BlobEndpoint=https://examref.blob.core.windows.net/QueueEndpoint=https://examref.queue.core.windows.net/FileEndpoint=https://examref.file.core.windows.net/TableEndpoint=https://examref.table.core.windows.... [🔗](#)

SAS token [ⓘ](#)

?sv=2019-10-10&ss=b&st=sco&sp=rw&dur=17&se=2020-05-07T10:32:41Z&st=2020-05-07T02:32:41Z&sig=10.0.0.3&spr=https&sig=b5MVhDwA1GmCKH1dXJK4AYZ330JLSMFN89jcEzeM%3D [🔗](#)

Blob service SAS URL

<https://examref.blob.core.windows.net/?sv=2019-10-10&ss=b&st=sco&sp=rw&dur=17&se=2020-05-07T10:32:41Z&st=2020-05-07T02:32:41Z&sig=10.0.0.3&spr=https&sig=b5MVhDwA1GmCKH1dXJK4AYZ330JL...> [🔗](#)

File service SAS URL

<https://examref.file.core.windows.net/?sv=2019-10-10&ss=b&st=sco&sp=rw&dur=17&se=2020-05-07T10:32:41Z&st=2020-05-07T02:32:41Z&sig=10.0.0.3&spr=https&sig=b5MVhDwA1GmCKH1dXJK4AYZ330JL...> [🔗](#)

Queue service SAS URL

<https://examref.queue.core.windows.net/?sv=2019-10-10&ss=b&st=sco&sp=rw&dur=17&se=2020-05-07T10:32:41Z&st=2020-05-07T02:32:41Z&sig=10.0.0.3&spr=https&sig=b5MVhDwA1GmCKH1dXJK4AYZ330JL...> [🔗](#)

Table service SAS URL

<https://examref.table.core.windows.net/?sv=2019-10-10&ss=b&st=sco&sp=rw&dur=17&se=2020-05-07T10:32:41Z&st=2020-05-07T02:32:41Z&sig=10.0.0.3&spr=https&sig=b5MVhDwA1GmCKH1dXJK4AYZ330JL...> [🔗](#)

FIGURE 2-8 Generated SAS token with connection string and SAS URLs

Also, you can create SAS tokens using Storage Explorer or the command-line tools (or programmatically using the REST APIs/SDK). To create an SAS token using Storage Explorer, you need to first select the resource (storage account, container, blob, and so on) for which the SAS token needs to be created. Then right-click the resource and select **Get Shared Access Signature**. [Figure 2-9](#) demonstrates how to create an SAS token using Azure Storage Explorer.



Shared Access Signature

X

Shared Access Signature

Start time: 2020-05-07 08:38 PM

Expiry time: 2020-05-08 08:38 PM

Time zone:

- Local
 UTC

Permissions:

- Read
 Write
 Delete
 List
 Add
 Create
 Update
 Process

Services:

- Blobs
 Files
 Queues
 Tables

Resource types:

- Service
 Container
 Object

[Learn more about permissions](#)

Create

Cancel

FIGURE 2-9 Creating a shared access signature using Azure Storage Explorer

Using shared access signatures

Each SAS token is a query string parameter that can be appended to the full URI of the blob or other storage resource for which the SAS token was created.

Create the SAS URI by appending the SAS token to the full URI of the blob or other storage resource.

The following example shows the combination in more detail. Suppose the storage account name is `examref`, the blob container name is `examrefcontainer`, and the blob path is `sample-file.png`. The full URI to the blob in storage is

[Click here to view code image](#)

`https://examrefstorage.blob.core.windows.net/examrefcontainer/sample-file.png`

The combined URI with the generated SAS token is

[Click here to view code image](#)

`https://examrefstorage.blob.core.windows.net/examrefcontainer/sample-file.png?sv=2019-10-10&ss=bfqt&srt=sco&sp=rw&lacupx&se=2020-05-08T08:50:14Z&st=2020-05-08T00:50:14Z&spr=https&sig=65tNhZtj2lu0tih8HQtK7aEL9YCIpGGprZocXjiQ%2Fko%3D`

Using account-level SAS

You can create the SAS at the storage account-level, too. With this SAS, you can manage all the resources belonging to the storage account. You can also perform write and delete operations for all the resources (blobs, tables, and so on) of the storage account.

Currently, stored access policy is not supported for account-level SAS.

More Info Account level SAS

You can learn more about the account level SAS here:

<https://docs.microsoft.com/rest/api/storageservices/create-account-sas>.

Using user delegation SAS

You can also create user delegation SAS using Azure AD credentials. The user delegation SAS is only supported by the Blob Storage, and it can grant access to containers and blobs. Currently, SAS is not supported for user delegation SAS.

More Info USer Delegation SAS

You can learn more about the user delegation SAS at

<https://docs.microsoft.com/rest/api/storageservices/create-user-delegation-sas>.

Using a stored access policy

An SAS token incorporates the access parameters (start and end time, permissions, and so on) as part of the token. The parameters cannot be changed without generating a new token, and the only way to revoke an existing token before its expiry time is to roll over the storage account key used to generate the token or delete the blob. In practice, these limitations can make standard SAS tokens difficult to manage.

Stored access policies allow the parameters for an SAS token to be decoupled from the token itself. The access policy specifies the start time, end time, and access permissions, and the access policy is created independently of the SAS tokens. SAS tokens are generated that reference the stored access policy instead of embedding the access parameters explicitly.

With this arrangement, the parameters of existing tokens can be modified by simply editing the stored access policy. Existing SAS tokens remain valid and use the updated parameters. You can revoke the SAS token by deleting the access policy, renaming it (changing the identifier), or changing the expiry time.

More Info Stored access policy effect

It can take up to 30 seconds for a stored access policy to take effect, and users might see an HTTP 403 when attempting access during that time.

Figure 2-10 shows the creation of stored access policies in the Azure portal.

Home > examref | Containers > examrefcontainer | Access policy

examrefcontainer | Access policy

Container

Search (Ctrl+ /) « Save

Add policy

Identifier * Permissions

examrefcontainer-171F1D44E2F ✓ 2 selected ▾

Start time Expiry time

05/07/2020 12:00:00 AM 05/08/2020 12:00:00 AM

(UTC-05:00) Eastern Time (U... ▾ (UTC-05:00) Eastern Time (US... ▾

OK Cancel

The screenshot shows the Azure portal interface for managing access policies. On the left, a sidebar lists navigation items: Overview, Access Control (IAM), Settings, Access policy (which is highlighted in grey), Properties, and Metadata. The main content area is titled 'examrefcontainer | Access policy'. It includes a search bar, a save button, and a large 'Add policy' button. Below these are sections for 'Identifier *' (containing the value 'examrefcontainer-171F1D44E2F' with a green checkmark) and 'Permissions' (showing '2 selected'). Further down are fields for 'Start time' (set to '05/07/2020 12:00:00 AM') and 'Expiry time' (set to '05/08/2020 12:00:00 AM'), each with a calendar icon. Below these time fields are dropdown menus for time zones, showing '(UTC-05:00) Eastern Time (U...' and '(UTC-05:00) Eastern Time (US...'. At the bottom of the form are 'OK' and 'Cancel' buttons.

FIGURE 2-10 Creating stored access policies using Azure portal

[Figure 2-11](#) shows stored access policies being created in Azure Storage Explorer.



X

Access Policies

Container:

Access policies:

Id	Start time:	Expiry time:	Read	Add	Create	Write	Delete	List	Remove
examrefcontainer-171F1D44E2F	2020-05-07 08:54 PM	2020-05-14 08:54 PM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="button" value="Remove"/>				

Time zone:

- Local
- UTC

FIGURE 2-11 Creating stored access policies using Azure Storage Explorer

To use the created policies, reference them by name when creating an SAS token using Storage Explorer or when creating an SAS token using PowerShell or the CLI tools.

More Info Max Access policies

You can only have a max of five access policies on a container, table, queue, or file share.

Manage access keys

The simplest way to manage access to a storage account is to use access keys. With the storage account name and an access key of the Azure storage account, you have full access to all data in all services within the storage account. You can create, read, update, and delete containers, blobs, tables, queues, and file shares. In addition, you have full administrative access to everything other than the storage account itself. (You cannot delete the storage account or change settings on the storage account, such as its type.)

Applications will use the storage account name and key for access to Azure Storage. Sometimes, this is to grant access by generating an SAS token, and sometimes, it is for direct access with the name and key.

To access the storage account name and key, open the storage account from within the Azure portal and click **Access Keys**. [Figure 2-12](#) shows the primary and secondary access keys for the `examref` storage account.



Storage account



Search (Ctrl+ /)



Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines.

[Learn more about regenerating storage access keys](#)

Storage account name

examref

key1

Key

qCOjsLGB4BalwgjVEDC/sFRXU9LLsw0/gBHQa7XcCf0xc7R0x4QoHy+bSqmW2SrCTKbG/Cfj3EdMo8URZTQ==

Connection string

DefaultEndpointsProtocol=https;AccountName=examref;AccountKey=qCOjsLGB4BalwgjVEDC/sFRXU9LLsw0/gBHQa7XcCf0xc7R0x4QoHy+bSqmW2SrCTKbG/Cfj3EdMo8URZTQ==;EndpointSuffix=core.windows.net

key2

Key

V0slq25grf8apAftf6iQz29F9e6lZFgDYwlsilUwNp8zLsztm7hcUs2SPwb6xbRY+7JMo9yuGh2D4ZTpCkQ==

Connection string

DefaultEndpointsProtocol=https;AccountName=examref;AccountKey=V0slq25grf8apAftf6iQz29F9e6lZFgDYwlsilUwNp8zLsztm7hcUs2SPwb6xbRY+7JMo9yuGh2D4ZTpCkQ==;EndpointSuffix=core.windows.net

FIGURE 2-12 Access keys for an Azure storage account

Each storage account has two access keys. This allows you to modify applications to use the second key instead of the first and then regenerate the first key. This technique is known as “key rolling,” and it allows you to reset the primary key with no downtime for applications that directly access storage using an access key.

Storage account access keys can be regenerated using the Azure portal or the command-line tools. In PowerShell, this is accomplished with the `New-AzStorageAccountKey` cmdlet; with Azure CLI, you will use the `az storage account keys renew` command.

Note Access keys and SAS tokens

Rolling a storage account access key will invalidate any SAS tokens that were generated using that key.

Managing access keys in Azure Key Vault

It is important to protect the storage account access keys because they provide full access to the storage account. Azure Key Vault helps safeguard cryptographic keys and secrets used by cloud applications and services, such as authentication keys, storage account keys, data encryption keys, and certificate private keys.

Keys in Azure Key Vault can be protected in software or by using hardware security modules (HSMs). HSM keys can be generated in place or imported. Importing keys is often referred to as bring your own key, or BYOK.

More Info Using HSM-protected keys for Azure Key Vault

You can learn more about the bring your own key (BYOK) scenario here:
<https://docs.microsoft.com/azure/key-vault/key-vault-hsm-protected-keys>.

You can manage storage account keys with key vault using Azure PowerShell or CLI. You can learn more using the following links:

- PowerShell: <https://docs.microsoft.com/azure/key-vault/>

vault/secrets/overview-storage-keys-powershell

- CLI: <https://docs.microsoft.com/azure/key-vault/secrets/overview-storage-keys>

Accessing and unencrypting the stored keys is typically done by a developer, although keys from Key Vault can also be accessed from ARM templates during deployment.

More Info Accessing encrypted keys from Azure Key Vault

You can learn more about how developers securely retrieve and use secrets from Azure Key Vault here:

<https://docs.microsoft.com/azure/storage/blobs/storage-encrypt-decrypt-blobs-key-vault>.

Configure Azure AD Authentication for a storage account

Azure AD authentication is beneficial for large customers who want to control the data access at an enterprise level based on their security and compliance standards. AAD authentication was recently added to the list in addition to existing shared-key and SAS token authorization mechanisms for Azure Storage (Blob and Queue). Azure blobs and queues are supported by Azure AD authentication. Azure Table storage is not supported with Azure AD authorization as of now.

Note Azure AD authorization support for storage accounts

Storage accounts that are created with the Azure Resource Manager deployment model only support Azure AD authorization.

AAD authentication enables customers to leverage Azure's RBAC for granting the required permissions to a security principal (users, groups, and applications) down to the scope of an individual blob container or queue. While

authenticating a request, Azure AD returns an OAuth 2.0 token to security principal, which can be used for authorization against Azure Storage (blob or queue).

Azure AD authorization can be implemented in many ways, such as assigning a RBAC roles to a security principal (users, groups, and applications), using a managed service identity (MSI), or creating shared access signatures signed by Azure AD credentials and so on.

If an application is running from within an Azure entity such as an Azure VM, a virtual machine scale set, or an Azure Functions app, it can use a managed service identity (MSI) to access blobs or queues.

Need More Review? Authorizing Access

More information about authorizing access to blob and queue data with managed identities for Azure resources can be found at

<https://docs.microsoft.com/en-us/azure/storage/common/storage-auth-aad-msi>

RBAC roles for blobs and queues

There are few built-in RBAC roles available in Azure for authorizing access to Blob and Queue Storage.

- **Storage Blob Data Owner:** Sets ownership and manages POSIX access control for Azure Data Lake Storage Gen2.
- **Storage Blob Data Contributor:** Grants read/write/delete permissions for Blob Storage.
- **Storage Blob Data Reader:** Grants read-only permissions for Blob Storage.
- **Storage Queue Data Contributor:** Grants read/write/delete permissions for Queue Storage.
- **Storage Queue Data Reader:** Grants read-only permissions for Queue Storage.
- **Storage Queue Data Message Processor:** Grants peek, retrieve, and delete permissions to messages in queues.

- **Storage Queue Data Message Sender:** Grants add permissions to messages in queues.

Need More Review? Built-in role details

For more information about built-in roles, see

<https://docs.microsoft.com/azure/role-based-access-control/built-in-roles#storage>.

Resource scope for blobs and queues

It is also important to determine the scope of the access for security principal before you assign an RBAC role. You can narrow down the scope to the container or queue level. Below are the valid scopes:

- **Container.** Under this scope, the role assignment will be applicable at the container level. All the blobs inside the container, the container properties, and the metadata will inherit the role assignment when this scope is selected.
- **Queue.** Under this scope, the role assignment will be applicable at the queue level. All the messages inside the queue, as well as queue properties and metadata will inherit the role assignment when this scope is selected.
- **Storage account.** Under this scope, the role assignment will be applicable at the storage account level. All the containers, blobs, queues, and messages within the storage account will inherit the role assignment when this scope is selected.
- **Resource group.** Under this scope, the role assignment will be applicable at the resource group level. All the containers or queues in all the storage accounts in the resource group will inherit the role assignment when this scope is selected.
- **Subscription.** Under this scope, the role assignment will be applicable at the subscription level. All the containers or queues in all the storage accounts in all the resource groups in the subscription will inherit the role assignment when this scope is selected.

AAD authentication and authorization in Azure portal

In the following example, you will learn how to configure the AAD authentication method in order to allow users to access the blob data.

In [Figure 2-13](#), you can see the `examrefcontainer` container has one blob named `UserCreateTemplate.csv`. Also, notice that the authentication method is currently set as **Access Key**.

The screenshot shows the Azure Storage Explorer interface. The left sidebar has a tree view with 'Home' at the top, followed by 'examrefcontainer' under 'Containers'. Below the tree are three navigation tabs: 'Overview' (selected), 'Access Control (IAM)', and 'Settings'. The 'Settings' tab has three sub-options: 'Access policy', 'Properties', and 'Metadata'. The main content area is titled 'examrefcontainer' and 'Container'. It features a search bar ('Search (Ctrl+ /)') and several action buttons: 'Upload', 'Change access level', 'Refresh', 'Delete', 'Change tier', 'Acquire lease', 'Break lease', 'View snapshots', and 'Create snapshot'. Below these buttons, the 'Authentication method' is listed as 'Access key (Switch to Azure AD User Account)' and the 'Location' is 'examrefcontainer'. There is also a search bar for blobs ('Search blobs by prefix (case-sensitive)'). On the right side, there is a checkbox labeled 'Show deleted blobs'. A table lists the blobs in the container:

Name	Modified	Access tier	Blob type	Size	Lease state
UserCreateTemplate.csv	5/8/2020, 10:52:13 PM	Hot (Inferred)	Block blob	682 B	Available

FIGURE 2-13 The overview blade of `examrefcontainer`

Switch the authentication method to **Azure AD User Account** by clicking **Switch To Azure AD Account**. You will see a warning message indicating that you do not have permission to list the data (see [Figure 2-14](#)).

The screenshot shows the Azure Storage Blob Container overview blade for a container named 'examrefcontainer'. The left sidebar includes links for Home, examrefcontainer, Overview, Access Control (IAM), Settings, Access policy, Properties, and Metadata. The main area features a search bar and various management actions: Upload, Change access level, Refresh, Delete, Change tier, Acquire lease, Break lease, View snapshots, and Create snapshot. A prominent red warning message states: 'You do not have permissions to list the data using your user account with Azure AD. Click to learn more about authenticating with Azure AD. This request is not authorized to perform this operation using this permission.' Below this, it says 'Authentication method: Azure AD User Account' and 'Location: examrefcontainer'. A search bar at the bottom allows for blob prefix search, and a checkbox for 'Show deleted blobs' is present. A table at the bottom lists blob details with columns: Name, Modified, Access tier, Blob type, Size, and Lease state. The message 'No blobs found.' is displayed.

FIGURE 2-14 The overview blade of examrefcontainer

Now let's assign **Storage Blob Data Reader** role to the logged in user at

container level. Go to the **Access Control (IAM)** blade on the container and select **Role** from the **Storage Blob Data Reader** drop-down menu. Then search for and select **CIE Administrator**. Click **Save** to apply the role assignment (see [Figure 2-15](#)).

Add role assignment

Role ⓘ

Storage Blob Data Reader ⓘ



Assign access to ⓘ

Azure AD user, group, or service principal



Select ⓘ

CIE

CA

CIE Administrator

harshulp_outlook.com#EXT#@MSP131499.onmicro...

Selected members:

CA

CIE Administrator

Remove

Save

Discard

FIGURE 2-15 Storage Blob Data Reader Role assignment

You should now see the current user with the role **Storage Blob Data Reader**, which appears under **Role Assignments** (see [Figure 2-16](#)).

examrefcontainer | Access Control (IAM)

Container

Search (Ctrl+ /)

Add Edit columns Refresh | Remove | Got feedback?

 Overview

 Access Control (IAM)

Settings

 Access policy

 Properties

 Metadata

Check access Role assignments Deny assignments Classic administrators Roles

Manage access to Azure resources for users, groups, service principals and managed identities at this scope by creating role assignments. [Learn more](#)

Number of role assignments for this subscription

3 2000

Name Type Role Scope Group by

Search by name or email All 3 selected All scopes Role

3 items (2 Users, 1 Service Principals)

<input type="checkbox"/>	Name	Type	Role	Scope
--------------------------	------	------	------	-------

Contributor

<input type="checkbox"/>	 example-app	App	Contributor <input type="button"/>	Subscription (Inherited)
--------------------------	---	-----	------------------------------------	--------------------------

Storage Blob Data Reader

<input type="checkbox"/>	 CIEAdministrator harshulp.outlook.com#EXT#@MSP131499....	User	Storage Blob Data Reader <input type="button"/>	This resource
--------------------------	---	------	---	---------------

FIGURE 2-16 Role assignments for examrefcontainer

If you navigate to **Overview** blade of `examrefcontainer` now, you will see the `UserCreateTemplate.csv` blob with authentication method shown as **Azure AD User Account** (see [Figure 2-17](#)).

Note RBAC roles effect

Sometimes, RBAC roles take up to 5 minutes to propagate the role assignments.

The screenshot shows the Azure Storage Explorer interface. At the top, the navigation bar displays "Home > examref | Containers > examrefcontainer". Below the navigation is a header with the container name "examrefcontainer" and its type "Container". A search bar labeled "Search (Ctrl+Shift+F)" is followed by a set of actions: Upload, Change access level, Refresh, Delete, Change tier, Acquire lease, Break lease, View snapshots, and Create snapshot. On the left, a sidebar menu includes "Overview" (selected), "Access Control (IAM)", "Settings", "Access policy", "Properties", and "Metadata". The main content area shows the container's authentication method as "Azure AD User Account" and its location as "examrefcontainer". A search bar for blobs is present. The table below lists the blobs in the container:

Name	Modified	Access tier	Blob type	Size	Lease state
UserCreateTemplate.csv	5/8/2020, 10:52:13 PM	Hot (Inferred)	Block blob	682 B	Available

FIGURE 2-17 The overview blade of examrefcontainer

Configure access to Azure Files

Azure Files provides managed file shares that are accessible over the SMB

protocol. SMB is a network file-sharing protocol, and Azure Files provides flexibility to use the following two types of identity-based authentication to access the shares.

- On-premises Active Directory Domain Services (AD DS)
- Azure Active Directory Domain Services (Azure AD DS)

In this section, you will learn how to use either of these domain services to access file shares over SMB. Azure file shares leverage Kerberos tokens to authenticate a user or application to access the file shares. You can configure authorization either at the share or directory/file levels. Share-level permission can be assigned using Azure built-in roles such as Storage File Data SMB Share Reader, which allows Azure AD users or groups to grant read access to an Azure file share.

On-premises Active Directory Domain Services (AD DS) authentication and authorization

You can enable AD DS authentication for your Azure file shares to authenticate using your on-premises AD DS credentials. You can also manage granular access control by syncing identities from on-premises AD DS to Azure AD with AD Connect. The share-level access can be availed with identities that are synced to Azure AD, and file/share-level access can be availed using on-premises AD DS credentials.

To configure identity-based authentication using AD DS, there is a five-step process you need to follow for your Azure file shares. You can get the documentation link using the Azure portal as shown in [Figure 2-18](#). Click the **How To Domain Join The Storage Account** hyperlink to access the latest official documentation.



Search (Ctrl+ /)



Save



Discard



Refresh

 Standard Premium

i This setting cannot be changed after the storage account is created.

Secure transfer required [?](#) Disabled EnabledAllow Blob public access [?](#) Disabled EnabledAllow shared key access [?](#) Disabled EnabledMinimum TLS version [?](#)

Version 1.2

Blob access tier (default) [?](#) Cool HotReplication [?](#)

Read-access geo-redundant storage (RA-GRS)

Large file shares [?](#) Disabled Enabled

i The current combination of subscription, storage account kind, performance,

Identity-based access for file shares

Azure Active Directory Domain Services (Azure AD DS) [?](#) Disabled Enabled

Active Directory Domain Services (AD DS)

[How to domain join this storage account](#) Overview Activity log Tags Diagnose and solve problems Access Control (IAM) Data migration Events Storage Explorer (preview)

Settings

 Access keys Geo-replication CORS Configuration Encryption Shared access signature Networking Security Static website Properties Locks

FIGURE 2-18 Configuring identity-based access for file shares using AD DS

Follow these steps for AD DS authentication:

1. Enable AD DS authentication on your storage account.
2. Assign share-level access permissions to an Azure AD identity.
3. Assign directory/file-level permissions using Windows ACLs.
4. Mount the Azure file share.
5. Update the password of your storage account identity in AD DS.

Azure Active Directory Domain Services (Azure AD DS) authentication and authorization

You can enable Azure AD DS authentication for your Azure file shares to authenticate with Azure AD credentials. Azure AD DS–joined Windows machines can access Azure file shares with Azure AD credentials over SMB.

To configure identity-based authentication using Azure AD DS, there is a set of steps you need to follow for your Azure file shares.

1. First, you must enable Azure AD DS for your storage account. You can enable AD DS using the Azure portal by accessing the storage account **Configuration** option, and then setting **Azure AD DS Identity-Based Access For File Shares** setting to **Enabled**, as shown in [Figure 2-19](#). Once **Enabled**, click the **Save** option at the top.

 examref | Configuration

Storage account

 Search (Ctrl+ /)

<

 Save  Discard  Refresh Overview Standard Premium

 This setting cannot be changed after the storage account is created.

 Activity logSecure transfer required  Disabled Enabled TagsAllow Blob public access  Disabled Enabled Diagnose and solve problemsAllow shared key access  Disabled Enabled Access Control (IAM)Minimum TLS version 

Version 1.2

 Data migration Events Storage Explorer (preview)

Settings

 Access keysBlob access tier (default)  Cool Hot Geo-replication

CORS

Replication 

Read-access geo-redundant storage (RA-GRS)

 EncryptionLarge file shares  Disabled Enabled Shared access signature

 The current combination of subscription, storage account kind, performance, location, and replication tier does not support large file shares.

 Networking

Identity-based access for file shares

Azure Active Directory Domain Services (Azure AD DS)  Disabled Enabled Security

Active Directory Domain Services (AD DS)

[How to domain join this storage account](#) Static website Properties Locks

FIGURE 2-19 Configuring Identity-Based Access For File Shares using Azure AD DS

2. You also need to register your storage account with AD DS and enable AD DS authentication for your Azure file shares. There are two ways to accomplish this:
 - **Use the AzFilesHybrid PowerShell module** This PowerShell module makes the required modifications and enables the feature for you. Note the following:
 - You can download and extract the AzFilesHybrid module here: <https://github.com/Azure-Samples/azure-files-samples/releases>.
 - Note that v0.2.0 and above are GA versions.
 - Install the module on a device that is domain joined to an on-premises AD DS with AD DS credentials that have permissions to create a Service Logon Account or a computer account in the target AD.
 - You must execute the commands using an on-premises AD DS credential that is synced to your Azure AD. The on-premises AD DS credential must have either the storage account owner or the contributor Azure role permissions.
 - Join-AzStorageAccountForAuth is a module command that registers the target storage account with your active directory environment under the target OU. You can also choose to create the identity that represents the storage account as either a Service Logon Account or Computer Account depends on the AD permission you have, as well as your preference. Moreover, you can run Get-Help Join-AzStorageAccountForAuth for more details on this cmdlet.
 - Manually perform the enablement actions To enable the feature manually, you will need have the Active Directory PowerShell and Az.Storage 2.0 modules installed. You also need to check your AD DS to see if either a computer account or Service Logon Account has already been created. If not, then you must create one. Now you can use the following command to enable the feature on your storage account. You can provide the target storage account and the required

AD domain information.

[Click here to view code image](#)

```
Set-AzStorageAccount
    -ResourceGroupName "<your-resource-group-name-here>"` 
        -Name "<your-storage-account-name-here>"` 
        -EnableActiveDirectoryDomainServicesForFile $true` 

        -ActiveDirectoryDomainName "<your-domain-name-here>"` 
            -ActiveDirectoryNetBiosDomainName "<your-netbios-domain-name-here>"` 
                -ActiveDirectoryForestName "<your-forest-name-here>"` 
                    -ActiveDirectoryDomainGuid "<your-guid-here>"` 
                    -ActiveDirectoryDomainsid "<your-domain-sid-here>"` 
                        -ActiveDirectoryAzureStorageSid "<your-storage-account-sid>"`
```

- Next, you need to configure share-level permissions in order to get access to your file shares. First, you need to set up a hybrid identity that is in AD DS and that is synced to your Azure AD. Authentication and authorization against identities that only exist in Azure AD, such as Azure Managed Identities (MSIs), are not supported with AD DS authentication. You can assign share-level permissions to the identity using Azure portal by accessing the **Access Control (IAM)** blade on an Azure file share. Select **Add A Role Assignment** and select one of the following roles and select the **identity**. Save the changes by clicking **Save** on the top of the blade.
Storage File Data SMB Share Reader—Allows read access in Azure Storage file shares over SMB.
 - **Storage File Data SMB Share Contributor** Allows read, write, and delete access in Azure Storage file shares over SMB.
 - **Storage File Data SMB Share Elevated Contributor** Allows read, write, delete, and modify NTFS permissions in Azure Storage file shares over SMB.
- 3. Once you assign share-level permissions, you must assign granular-level permissions at the root, directory, or file level using basic and advanced Windows ACLs. The following permissions are supported on the root

directory of a file share:

- BUILTIN\Administrators:(OI)(CI)(F)
 - NT AUTHORITY\SYSTEM:(OI)(CI)(F)
 - BUILTIN\Users:(RX)
 - BUILTIN\Users:(OI)(CI)(IO)(GR,GE)
 - NT AUTHORITY\Authenticated Users:(OI)(CI)(M)
 - NT AUTHORITY\SYSTEM:(F)
 - CREATOR OWNER:(OI)(CI)(IO)(F)
4. You should mount an Azure file share from a domain-joined VM Log in to the domain-joined VM using an Azure AD identity such as user with the granted permissions mentioned in previous steps. Note that if the client machine is not in the AD DS network, you must use VPN in order to successfully authenticate. You can use the net use command to mount the file share. Following is a sample command:

[Click here to view code image](#)

```
net use <drive-letter>: \\<storage-account>.file.core.windows.net\<fileshare>
```

5. If you want to grant permission to additional users, you can follow the steps again with the target Azure AD identity to provide access to Azure file shares.
6. To configure ACLs with superuser permissions, you must mount the share by using your storage account key from your domain-joined VM:

[Click here to view code image](#)

```
net use <drive-letter>: \\<storage-account>.file.core.windows.net\<fileshare> /  
user:Azure\<storage-account-name> <storage-account-key>
```

7. You can configure the Windows ACLs using either Windows File Explorer or icacls. You can use following command to grant full permissions to all directories and files under the file share, including the root directory:

[Click here to view code image](#)

```
icacls <mounted-drive-letter>: /grant <user-email>:(f)
```

Need More Review? More about icacls

More information on icacls can be found at
<https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/icacls>.

8. Alternatively, you can also use Windows File Explorer to grant the necessary permissions. Please note, both share-level and file/directory level permissions are enforced when a user attempts to access a file/directory. If there is a conflict in any of these permissions, the most restrictive permission will be applied. For example, if the user has read/write access at the share level but has only read access at the file level, then the user can only read the file. The same is true in reverse.
9. The final step is to update the password of AD DS identity/account that represents your storage account in an organizational unit or domain that enforces password expiration time. You can use the `AzStorageAccountADObjectPassword` command from the `AzFilesHybrid` module to update the password. This command performs actions similar to storage account key rotation and must be performed by a hybrid user with owner permission to the storage account and AD DS permissions to change the password of the identity representing the storage account. You must run the following command in an on-premises, AD DS-joined environment:

[Click here to view code image](#)

```
Update-AzStorageAccountADObjectPassword -RotateToKerbKey kerb2
-ResourceGroupName
"<resource-group-name>" -StorageAccountName "<storage-account-name>"
```

Skill 2.2: Manage storage

If your dataset is large enough, or if you have limited or no connectivity from your data to the Internet, you might want to physically ship the data and import it into Microsoft Azure instead of uploading it. The solution is the Azure Import/Export service, which allows you to ship data into or out of an Azure Storage account by physically shipping disks to an Azure datacenter. This

service is ideal when uploading or downloading the data directly is either not possible or prohibitively expensive. Azure Import/Export is only used with Blob Storage and Azure Files. You will learn how to import and export the data in this section. In a later part of this skill, you will also learn how to use tools like Azure Storage Explorer and AzCopy.

This section covers how to:

- [Create an export from an Azure job](#)
- [Create an import into an Azure job](#)
- [Install and use Azure Storage Explorer](#)
- [Copy data by using AZCopy](#)
- [Implement Azure Storage replication](#)
- [Configure blob object replication](#)

Create an export from an Azure job

An export job allows you to export large volumes of data from Azure Storage to your on-premises environment by shipping you the data on disk. This service only supports the export of blobs.

To export blob data, create an export job on the storage account using the management portal. To create an export job, do the following:

1. Log in to the Azure portal, click **All Services**, and search for and select **Import/Export Jobs**.
2. Click **Create Import/Export Job**.
3. On the **Basics** tab (as shown in [Figure 2-20](#)), choose **Export From Azure** and specify the job name and the resource group to contain the created job. Click **OK**.
4. On the **Job Details** tab, choose which storage account to export from and choose the blobs to export. You have the following options.
 - **Export All**
 - **Selected Containers And Blobs**

- **Export From Blob List File (XML Format)**

5. Click **OK**.
6. On the **Return Shipping Info** tab, specify your carrier information and the address for the disks to be shipped to. Click **OK**.
7. On the **Summary** tab, click the **OK** button after confirming the export job.

Home > New > Import export job > Create import/export job > Basics

Create import/export job

X Basics □ X

1 Basics >
Configure basic settings

2 Job details >
Configure data and destination

3 Return shipping info >
Configure return shipping details

4 Summary >
Configure return shipping details

Type *

Import into Azure
 Export from Azure

Name *

ExamRefExportJob ✓

Subscription *

Visual Studio Ultimate with MSDN ✓

Resource group *

(New) ExamRefRG ✓

Create new

FIGURE 2-20 The Create Import/Export Job blade in the Azure portal

More Info Walkthrough creating a data export job

To learn more about creating an import job see

<https://docs.microsoft.com/azure/storage/common/storage-import-export-data-from-blobs>.

After you receive the disks from Microsoft, you will need to retrieve the BitLocker keys from the Azure portal to unlock the disks.

Create an import into an Azure job

An import job allows you to import large volumes of data to Azure by shipping the data on disk to Microsoft.

The first step to import data using the Azure Import/Export service is to install the Microsoft Azure Import/Export tool known as the WAImportExport tool.

Note WAImportExport Tool

There are two versions of the WAImportExport tool. Version 1 is recommended for Azure Blob Storage, and version 2 is recommended for Azure Files.

Download links:

- Version 1: <https://www.microsoft.com/download/details.aspx?id=42659>
- Version 2: <https://www.microsoft.com/download/details.aspx?id=55280>

Additional requirements and limitations of the Azure Import/Export Jobs tool include:

- Windows 7, Windows Server 2008 R2, or a later OS version is required.
- The tool also requires .NET Framework 4.5.1 or later and BitLocker.
- All storage account types are supported (General-Purpose V1, General-Purpose V2, and Blob Storage).
- Block, Page, and Append Blobs are supported for both import and export.

- The Azure Files service is only supported for import jobs but not export jobs.

Note WAImportExport Tool support

This tool only works with 64-bit operating systems and might not work with 32-bit operating systems.

[Table 2-3](#) lists the disks requirements for sending data to the Import/Export service.

TABLE 2-3 Supported disks for the Import/Export service

Disk Type	Size	Supported	Not Supported
SSD	2.5"	SATA III	—
HDD	3.5"	SATA II, SATA III	External HDD with built-in USB adaptor Disk inside the casing of an external HDD USB drives



Exam Tip

A single import/export job can have a maximum of 10 HDDs and SSDs and a mix of HDDs and SSDs of any size.

The second step to import data is to prepare your drives using the

WAImportExport tool and copy the data to transfer to the drives.

When preparing the drive, the first session, requires several parameters, such as the destination storage account key, the BitLocker key, and the log directory. The following example (for the v1 tool) shows the syntax for using the Azure Import/Export tool with the `PrepImport` parameter to prepare the disk for an import job for the first session.

[Click here to view code image](#)

```
WAImportExport.exe PrepImport /j:<JournalFile> /id:<SessionId>
[/logdir:<LogDirectory>]
[/sk:<StorageAccountKey>] /InitialDriveSet: <driveset> /DataSet:
<driveset>
```

The Azure Import/Export tool creates a journal file that contains the information necessary to restore the files on the drive to the Azure Storage account, such as mapping a folder or file to a container, blob, or files. Each drive used in the import job will have a unique journal file that is created by the tool.

Note Using the WAImportExport tool

To add a single file to the drive and journal file, use the `/srcfile` parameter, instead of the `/srcdir` parameter.

The Azure Import/Export tool supports a number of other parameters. For a full list, see:

- **Version 1:** <https://docs.microsoft.com/azure/storage/common/storage-import-export-tool-preparing-hard-drives-import-v1>
- **Version 2:** <https://docs.microsoft.com/azure/storage/common/storage-import-export-tool-preparing-hard-drives-import>

Once the drive preparation is complete, the third step in the import process is to create an import job through the Azure portal. To create an import job, do the following:

1. Log in to the Azure portal and click **All Services > Storage > Import/Export Jobs**.
2. Click **Create Import/Export Job**.
3. On the **Basics** tab, choose **Import Into Azure** and specify the job name

and the resource group to contain the created job. Click **OK**.

4. On the **Job Details** tab, choose the journal file created with the `WAImportExport.exe` tool and select the destination storage account. Click **OK**.
5. On the **Return Shipping Info** tab, specify your carrier information and return address for the return disks. Click **OK**.
6. On the **Summary** tab, click the **OK** button after confirming the import job.

Having created the import job, the fourth step in the import process is to physically ship the disks to Microsoft using a supported courier service with a tracking number for your package. Once you have the tracking number, then you update the job properties of the existing import job. The drives will be returned using the courier information provided in the import job.

Check the job status regularly using get job operation until it is completed. You can then verify that the data has been uploaded to Azure.

Need More Review? Walkthrough creating a data import job

To learn more about creating an import job see the following:

- Version 1 (blobs).
<https://docs.microsoft.com/azure/storage/common/storage-import-export-data-to-blobs>
- Version 2 (files).
<https://docs.microsoft.com/azure/storage/common/storage-import-export-data-to-files>

Install and use Azure Storage Explorer

Azure Storage Explorer is a cross-platform application designed to help you quickly manage one or more Azure Storage accounts. It can be used with all storage services: Blob Storage, Azure Tables, Queue Storage, and Azure Files. In addition, Azure Storage Explorer also supports the CosmosDB and Azure Data Lake Storage services.

You can install Azure Storage Explorer by navigating to its landing page at <https://azure.microsoft.com/features/storage-explorer/> and selecting your

operating system choice (Windows, macOS, or Linux).

In addition, a version of Storage Explorer with similar functionality is integrated into the Azure portal. To access, simply click **Storage Explorer (Preview)** from the storage account blade.

Connecting Storage Explorer to Storage Accounts

After Storage Explorer is installed, you can connect to Azure Storage in one of five different ways (shown in [Figure 2-21](#)):

- **Add An Azure Account.** This option allows you to sign in using a work or Microsoft account and access all your storage accounts via role-based access control.
- **Using A Connection String.** This option requires you to have access to the connection string of the storage account. The connection string is retrievable by opening the storage account blade in the Azure portal and clicking **Access Keys**.
- **Use A Shared Access Signature URI.** A shared access signature provides access to a storage account without requiring an account key to be shared. Access can be restricted, for example, to read-only access for Blob Storage for one week only.
- **Using A Storage Account Name And Key.** This option requires you to have access to the storage account name and key. These values can also be accessed from the Azure portal under Access keys.
- **Attach To A Local Emulator.** Allows you to connect to the local Azure Storage emulator as part of the Microsoft Azure SDK.

After connecting, you then filter on which subscriptions to use. Once you select a subscription, all the supported services within the subscriptions will be made available. [Figure 2-22](#) shows an expanded Azure Storage account named examref.



Connect to Azure Storage

X

Connect to Azure Storage

How do you want to connect to your storage account or service?

- Add an Azure Account

Azure environment:

Azure



- Add a resource via Azure Active Directory (Azure AD)
- Use a connection string
- Use a shared access signature (SAS) URI
- Use a storage account name and key
- Attach to a local emulator

Next

Cancel

FIGURE 2-21 Connecting to an Azure Storage Account using Azure Storage Explorer

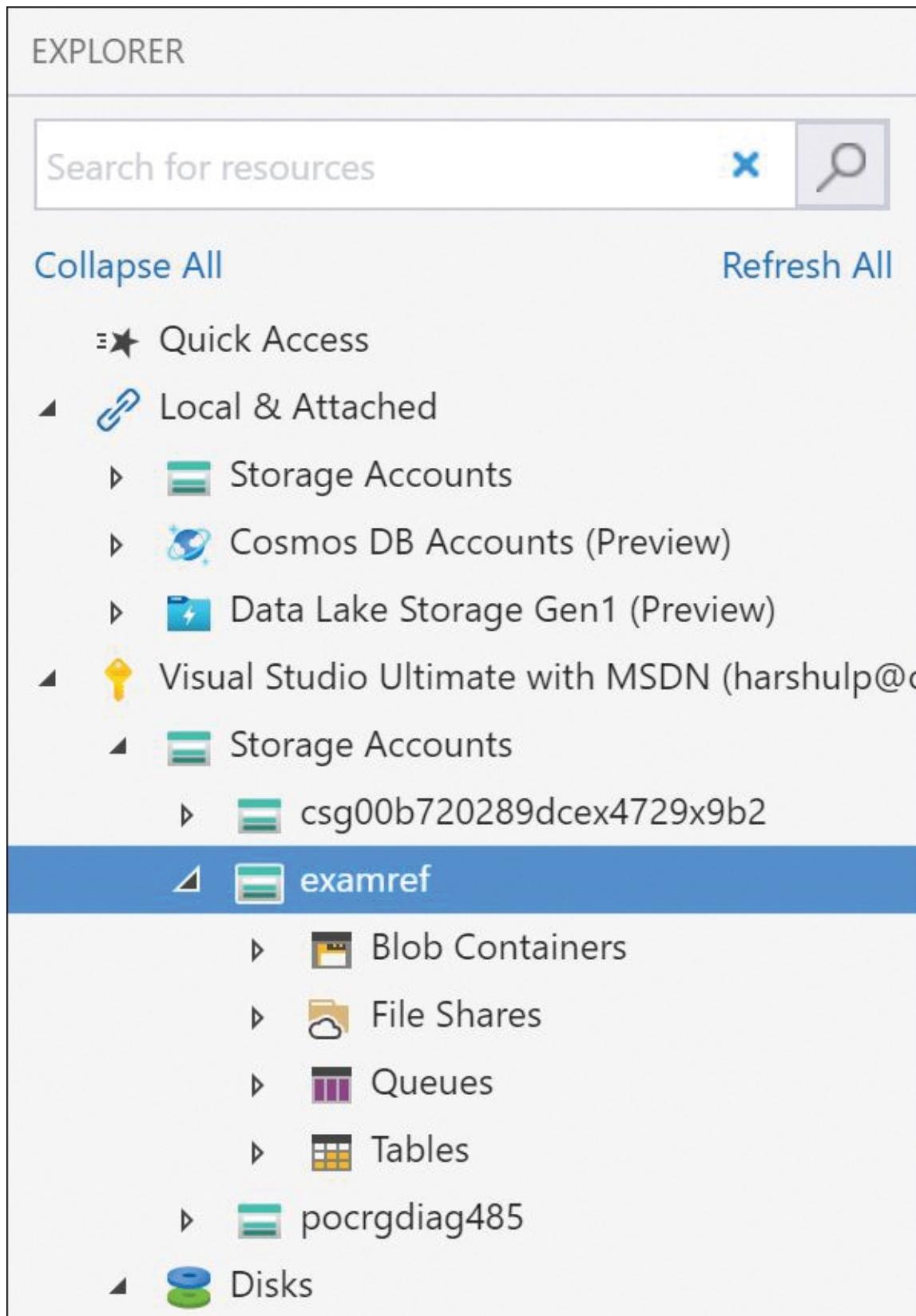


FIGURE 2-22 Azure Storage Explorer showing an Azure Storage Account

beneath the subscription

Note Connect to Cosmos DB using Storage Explorer

You can connect to Cosmos DB using Storage Explorer. You can find more details at <https://docs.microsoft.com/azure/cosmos-db/storage-explorer>.

Using Storage Explorer

Using Storage Explorer, you can manage each of the storage services: Blob Storage, Azure Tables, Queue Storage, and Azure Files. **Table 2-4** summarizes the supported operations for each service.

TABLE 2-4 Storage Explorer Operations

Storage Service	Supported Operations
Blob	Blob containers. Create; rename; copy; delete; control public access level; manage leases; and create and manage shared access signatures and access policies Blobs. Upload; download; manage folders; rename and delete blobs; copy blobs; create and manage blob snapshots; change blob access tier; and create and manage shared access signatures and access policies
Table	Tables. Create; rename; copy; delete; and create and manage shared access signatures and access policies
	Table entities. Import, export, view, add, edit, delete, and query
Queue	Queues. Create, delete, create, and manage shared access signatures and access policies

Messages. Add, view, dequeue, and clear all messages

Files **File shares.** Create; rename; copy; delete; create and manage snapshots; connect a VM to a file share; and create and manage shared access signatures and access policies

Files. Upload folders or files; download folders or files; manage folders; copy, rename; and delete

In each case, Azure Storage Explorer provides an intuitive GUI interface for each operation.

Storage blob copy

The Azure Storage Explorer can be used to perform a storage blob copy. To copy between storage accounts, navigate to the source storage account, select one or more files, and click the **Copy** button on the toolbar. Next, navigate to the destination storage account, expand the container that you want to copy to, and click **Paste** from the toolbar. In [Figure 2-23](#), the `CreateUserTemplate.csv` blob was copied from `examref\srccontainer` to `examref\destcontainer` using this technique.

Microsoft Azure Storage Explorer

File Edit View Help

EXPLORER srccontainer destcontainer

Search for resources X

Collapse All Refresh All

Quick Access

Local & Attached

- Storage Accounts
- Cosmos DB Accounts (Preview)
- Data Lake Storage Gen1 (Preview)

Visual Studio Ultimate with MSDN (harshulp@...)

- Storage Accounts
 - csg00b720289dce4729x9b2
 - examref
 - Blob Containers
 - destcontainer
 - examrefcontainer
 - srccontainer
 - File Shares
 - Queues
 - Tables
 - pocrgdiag485
 - Blob Containers
 - bootdiagnostics-scovm-0cb&
 - File Shares
 - Queues
 - Tables
- Disks
 - cloud-shell-storage-centralindia
 - NetworkWatcherRG
 - poc-rg
 - rgCoreNetwork
 - VS-harshulp-Group

← → ↑ ↓ Active blobs (default) destcontainer

Name	Access Tier	Access Tier Last Modified	Last Modified	Block Type	Content Type	Size	Status	Remaining
UserCreateTemplate.csv	Hot (inferred)		5/9/2020, 12:17:20 AM	Block Blob	application/vnd.ms-excel	682 B	Active	

Showing 1 to 1 of 1 cached items

Actions Properties Activities

URL	https://examref.blob.core.windows.net	Clear completed Clear successful
Type	Blob Container	Transfer of 'srccontainer/UserCreateTemplate.csv' to 'destcontainer/' complete (used SAS)
HNS Enabled	false	
Lease State	available	
Lease Status	unlocked	
Public Read Access	off	
Last Modified	Sat, 09 May 2020 04:10:23 GMT	

FIGURE 2-23 Using the async blob copy service with Storage Explorer

Copy data by using AzCopy

AzCopy is a command-line utility that you can use to perform large-scale bulk transfer of data to and from Azure Storage. AzCopy performs all the operations asynchronously and can run simultaneously. Moreover, it is also fault-tolerant, so if the operation is interrupted for some reason, it can resume from where it left off once the issue is resolved.

The latest version of AzCopy lets you take incremental backups of blobs and keep it synchronized in order to contain the same version of data. AzCopy can be added to the system path, so that you can run AzCopy from any of the folder from your system while using it in Windows PowerShell. Otherwise, you have to change the directory to where the AzCopy executable is stored every time. You can see list of commands using `azcopy -h`.

Note AzCopy with Storage Explorer

Storage Explorer is a graphical user interface which uses AzCopy to perform all its data transfer operations in the backend.

AzCopy needs an authentication to Azure Storage first before it runs any operations within the session. It can be achieved by running the `azcopy login` command and signing in. AzCopy also supports other authorizations, such as service principal, SAS token, access key, managed identity, and so on. For example, run this command to authenticate using service principal:

[Click here to view code image](#)

```
azcopy login --service-principal --application-id <application-id>
--tenant-id=<tenant-id>
```

More Info Create an AAD app and service principal

You can find step by step instructions for how to connect an Azure AD app and service principal at <https://docs.microsoft.com/azure/active-directory/develop/howto-create-service-principal-portal>.

Upload/download the data using AzCopy

You can upload the data to Azure Blob Storage using AzCopy. The only condition is that the storage account and destination container should already exist. In the example below, the `CreateUserTemplate.csv` file will be copied to the `destcontainer`.

[Click here to view code image](#)

```
azcopy copy "CreateUserTemplate.csv"  
"https://examref.blob.core.windows.net/  
destcontainer/"
```

If you are using SAS token, the syntax would be as follows:

[Click here to view code image](#)

```
azcopy copy "CreateUserTemplate.csv"  
"https://examref.blob.core.windows.net/  
destcontainer/?<sas token>"
```

You can upload multiple files with folder structures using the `--recursive=true` option with AzCopy.

[Click here to view code image](#)

```
azcopy copy "CreateUserTemplate.csv"  
"https://examref.blob.core.windows.net/  
destcontainer/?<sas token>"
```

You can also download the data from Azure Blob Storage using AzCopy. In the example below, the `CreateUserTemplate.csv` file will be downloaded from the `srccontainer`.

[Click here to view code image](#)

```
azcopy copy "https://examref.blob.core.windows.net/srccontainer/"  
"CreateUserTemplate.csv"
```

Async blob copy

The AzCopy application can also be used to copy between storage accounts. The following example shows how to copy the blob from source storage account's container to destination storage account's container using SAS token.

[Click here to view code image](#)

```
AzCopy copy "https://examref.blob.core.windows.net/  
srccontainer/[blob-path]?<sas token>"  
"https://examrefdest.blob.core.windows.net/destcontainer/[blob-  
path]?<sas token>"
```

More Info AzCopy

AzCopy version 10 is multi-platform, and works with Windows, Linux, and macOS. For more information on AzCopy, see <https://docs.microsoft.com/azure/storage/common/storage-use-azcopy>.

Sync blob copy

You can use the `azcopy sync` command to do synchronized copy between two blob containers. This command synchronizes the contents of a destination container with a source container by copying blobs if the last modified time of a blob in the destination is earlier than that of the corresponding blob in the source. By default, the recursive flag is `true` for the `sync` command and copies all subdirectories:

[Click here to view code image](#)

```
azcopy sync "https://examref.blob.core.windows.net/srccontainer/?  
<sas token>"  
"https://examref.blob.core.windows.net/destcontainer/"
```

Note Delete Destination Flag

You can use `--delete-destination` flag with the `azcopy sync` command if you want to delete blobs in the destination that don't exist in the source. It can be set to `true`, `false`, or `prompt`. Using `prompt` will prompt you for deletions to make it safer.

Implement Azure Storage replication

The data in your Azure Storage accounts is always replicated for durability and high availability. The built-in storage replication options were discussed at a high level in [Table 2-5](#). It's important to understand when each replication option

should be used and the level of availability you require for your scenario. [Table 2-3](#) describes the scenarios and expected availability for each of the replication options.

TABLE 2-5 Durability and availability for various replication options

Scenario	LRS	ZRS	GRS	RA-GRS	GZRS	RA-GZRS
Supported storage account types	GPv21, GPv12, blob	GPv2	GPv1, GPv2, blob	GPv1, GPv2, blob	GPv2	GPv2
Server or other failure within a datacenter	Available	Available	Available	Available	Available	Available
Failure affecting an entire data center (such as a fire)	Not available	Available	Available	Available	Available	Available
Failure affecting all datacenters in a region (such as a hurricane)	Not available	Not available	Microsoft controlled failover	Read access only until failed over	Microsoft controlled failover	Read access only until failed over
Designed durability	At least 99.999999 percent	At least 99.999999 percent	At least 99.999999 percent	At least 99.999999 percent	At least 99.999999 percent	At least 99.999999 percent
Availability SLA for read	At least 99.9 percent	At least 99.9 percent	At least 99.9 percent	At least 99.99 percent	At least 99.99 percent	At least 99.99 percent

requests	(99 percent for cool access tier)	(99 percent for cool access tier)	(99 percent for cool access tier)	(99.9 percent for cool access tier)	(99.9 percent for cool access tier)	(99.9 percent for cool access tier)
Availability SLA for write requests	At least 99.9 percent (99 percent for cool access tier)	At least 99.9 percent (99 percent for cool access tier)	At least 99.9 percent (99 percent for cool access tier)	At least 99.9 percent (99 percent for cool access tier)	At least 99.9 percent (99 percent for cool access tier)	At least 99.9 percent (99 percent for cool access tier)

Changing storage account replication mode

Storage accounts can be moved freely between the LRS, GRS, and RA-GRS replication modes. Azure will replicate the data asynchronously in the background as required.

Migrating to or from the ZRS, GZRS, and RA-GZRS replication mode works differently. The recommended approach is to simply copy the data to a new storage account with the desired replication mode, using a tool such as AzCopy. This might require application downtime. Alternatively, you can request a live data migration via Azure Support.

You can set the replication mode for a storage account after it is created through the Azure portal by clicking the **Configuration** link on the storage account and selecting the **Replication Type** (see [Figure 2-24](#)).

Home > examref | Configuration

examref | Configuration

Storage account

Search (Ctrl+ /) Save Discard

Overview

The cost of your storage account depends on the usage and the options you choose below.

Learn more

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Private endpoint connections

Advanced security

Static website

Properties

Locks

Export template

Blob service

Account kind

StorageV2 (general purpose v2)

Performance

Standard Premium

Secure transfer required

Disabled Enabled

Access tier (default)

Cool Hot

Replication

Read-access geo-redundant storage (RA-GRS)

Large file shares

Disabled Enabled

Info The current combination of storage account kind, performance, replication and location does not support large file shares.
Learn more.

Identity-based access for file shares

Azure Active Directory Domain Service (AAD DS)

Disabled Enabled

Active Directory (AD)

How to domain join this storage account

Data Lake Storage Gen2

Hierarchical namespace

Disabled Enabled

FIGURE 2-24 The configuration blade of an Azure storage account

More Info More examples with PowerShell

There are many variations for using the async copy service with PowerShell. For more information, see the following:

<https://docs.microsoft.com/powershell/module/az.storage/start-azstorageblobcopy>.

More Info More examples with CLI

There are many variations for using the async copy service with the Azure CLI. For more information, see

<https://docs.microsoft.com/cli/azure/storage/blob/copy>.

Configure blob object replication

Azure Storage blob object replication provides asynchronous replication of block blobs from one storage account to another. The blobs are replicated based on the defined replication rules.

You can leverage object replication only when blob versioning is enabled for both the source and destination storage accounts and the blob change feed is enabled for the source storage account.

Note blob versioning and blob change feed

Blob versioning captures the state of a blob when it is modified or deleted, Azure storage creates a new version ID for a blob with each change. The blob change feed provides all the changes with the blobs and its metadata in form of transactional logs.

There are various benefits you can get by using object replication:

- For large data processing jobs, you can analyze the data in a single region, and you can distribute results to additional regions as needed. This saves

processing time and compute resources to perform the same in all regions.

- With replication, the users can read data from the replicated region as well. Hence, you can reduce latency for your read requests by giving them the flexibility to choose the nearest region to read the data.
- Compute workloads can now process the same sets of block blobs in different regions using object replication.
- You can reduce the costs by moving your replicated data to the archive tier using Lifecycle Management policies.

Keep in mind, object replication performs multiple read and write transactions against the source and destination accounts. This can incur additional costs.

To set up the object replication rules, open the storage account, browse to **Object Replication** under **Blob Service**, and click **Set Up Replication Rules** (see [Figure 2-25](#)). You can define up to 10 replication rules per policy.

examref | Object replication

Storage account

Search (Ctrl+ /)

+ Set up replication rules

Upload replication rules

Refresh

Feedback

Blob service

Containers

Custom domain

Data protection

Object replication

Azure CDN

Add Azure Search

Lifecycle Management

Blob inventory (preview)

File service

File shares

When object replication is enabled, blobs are copied asynchronously from a source storage account to a destination account. Cross-tenant policies will show up under "Other accounts". The storage accounts may be in different Azure regions.

[Learn more about object replication](#)

Your accounts Other accounts

Objects copied from this account

Destination account

Source container

Destination container

Filters

No replication policies found

Objects copied into this account

Source account

Source container

Destination container

Filters

No replication policies found

FIGURE 2-25 Set up replication rules option on the Object Replication blade

You need to select the **Destination Subscription** and **Destination Storage Account** that will be used for replication. You also need to select the **Source Container** and **Destination Container** in a pair. You can limit the replication scope with filters by specifying the prefix match for blobs. See [Figure 2-26](#).

Create replication rules for your organization

Tip When you create object replication rules, blob change feed and blob versioning are automatically enabled for the source and destination storage accounts. Enabling these features may

Destination details

To begin replicating objects, specify the source storage account and the destination storage account.

[Learn more about copying objects in object replication](#)

Destination subscription *

Visual Studio Ultimate with MSDN



Destination storage account *

examrefdest



[Don't see your account?](#)

Container pair details

A container pair consists of a container in the source account and a container in the destination account. Objects in the source container are copied over to the destination container according to the replication rule. You can optionally filter which objects are copied by specifying a prefix match and by copying objects created only after a specified date and time.

Source container

Destination container

Filters

Copy over

data



datadest



1 (add)

Only new objects (change)



Select a source container



Select a destination container



FIGURE 2-26 Destination details for a replication rule under Object replication

You can also control how objects are copied over to the destination container using three options: **Everything**, **Only New Objects**, and **Custom** (see [Figure 2-53](#)). If you select **Everything**, then all the blobs matching the filters will be copied over to the destination container, but if you select **Only New Objects**, then only the newly added blobs matching the filters will be copied over to the destination container. If you select **Custom**, then you will have a chance to manually specify a date and time to copy the blobs created later, as shown in [Figure 2-27](#).

Copy over

X

To manage how many objects are copied to the destination container, specify that objects are copied based on when they were added to the source container.

Copy over

- Everything
- Only new objects
- Custom

Copy objects that were created starting from *

12/06/2020



10:00:00 AM

(UTC-05:00) Eastern Time (US & Canada)



FIGURE 2-27 Copy Over option for a replication rule

Once created, the consolidated view of replication rules can be viewed by visiting the **Object Replication** blade. You can also right-click a rule and select **Edit Rules**, **Download Rules**, or **Delete Rules**, as shown in [Figure 2-28](#). The downloaded rules can be edited and reused using the **Upload Replication Rules** options instead of re-creating them.

To check the replication status of a source blob, you can select the source blob to see its properties. You can see the **Object Replication** section, which can show the replication **Policy ID** and **Rule ID** along with replication **Status** (see [Figure 2-29](#)).

examref | Object replication



Storage account

Search (Ctrl+ /)



Set up replication rules



Upload replication rules



Refresh



Feedback

Blob service

Containers

Custom domain

Data protection

Object replication

Azure CDN

Add Azure Search

Lifecycle Management

Blob inventory (preview)

File service

File shares

When object replication is enabled, blobs are copied asynchronously from a source storage account to a destination account. Cross-tenant policies will show up under "Other accounts". The storage accounts may be in different Azure regions.

[Learn more about object replication](#)

Your accounts

Other accounts

Objects copied from this account

Destination account

Source container

Destination container

Filters

▼ examrefdest

Edit rules



Download rules



Delete rules



data

datadest

1

Source account

Source container

Destination container

Filters

No replication policies found

FIGURE 2-28 Replication rules

0.pdf

Blob

Save Discard Download Refresh | Delete | Change tier

[Overview](#) [Versions](#) [Snapshots](#) [Edit](#) [Generate SAS](#)

Properties

URL	https://cs21003bffd965...
LAST MODIFIED	10/24/2020, 9:07:24 AM
CREATION TIME	10/24/2020, 9:07:24 AM
VERSION ID	-
TYPE	Block blob
SIZE	304.65 KiB
ACCESS TIER	Hot (Inferred)
ACCESS TIER LAST MODIFIED	N/A
SERVER ENCRYPTED	true
ETAG	0x8D8781DC039DB48
CONTENT-TYPE	application/pdf
CONTENT-MD5	1pKqomPc11X7QCKm7cqz2g==
LEASE STATUS	Unlocked
LEASE STATE	Available
LEASE DURATION	-
COPY STATUS	-
COPY COMPLETION TIME	-

[Undelete](#)

Object replication

Policy ID	Rule ID	Status
b13b0e94-fb7b-4507-afe8-...	48d5e482-9331-4e84-8086...	Complete

Metadata

Key	Value
<input type="text"/>	<input type="text"/>

FIGURE 2-29 Replication status

There are certain limitations with blob object replication that are crucial to review before implementation:

- Object replication doesn't work with the Archive tier.
- Blob snapshots and immutable snapshots are not supported with object replication.
- Object replication doesn't work with accounts with a hierarchical namespace (Azure Data Lake Storage Gen2).
- Because block blob data is replicated asynchronously, there is no SLA on when accounts are in sync. However, you can check the replication status of a blob.
- The source account can only have a maximum of two destination accounts.
- Once you create a replication policy, the destination container is read-only, and you can no longer perform write operations against it.

Skill 2.3: Configure Azure Files and Azure Blob Storage

Azure Files is a fully managed file share service that offers endpoints for the Server Message Block (SMB) protocol, also known as Common Internet File System or CIFS. This allows you to create one or more file shares in the cloud (with default max size as 5 TiB per share). You can enable large file share for a storage account and create file shares up to 100 TiB. Also, if you are using the Premium SKU, you get 100 TiB by default. Azure Files can be used for similar uses as a regular Windows file server, such as shared storage or for new uses such as part of a lift and shift migration strategy.

This section covers how to:

- [Create an Azure file share](#)
- [Create and configure Azure File Sync service](#)

- Configure Azure Blob Storage
- Configure storage tiers for Azure blobs
- Configure blob Lifecycle Management

Create an Azure Fileshare

There are several common use cases for using Azure files. A few examples include the following:

- Migration of existing applications that require a file share for storage
- Shared storage of files, such as web content, log files, application configuration files, or even installation media
- Replace an existing fileserver

[Figure 2-30](#) shows the hierarchy of files stored in Azure Files.

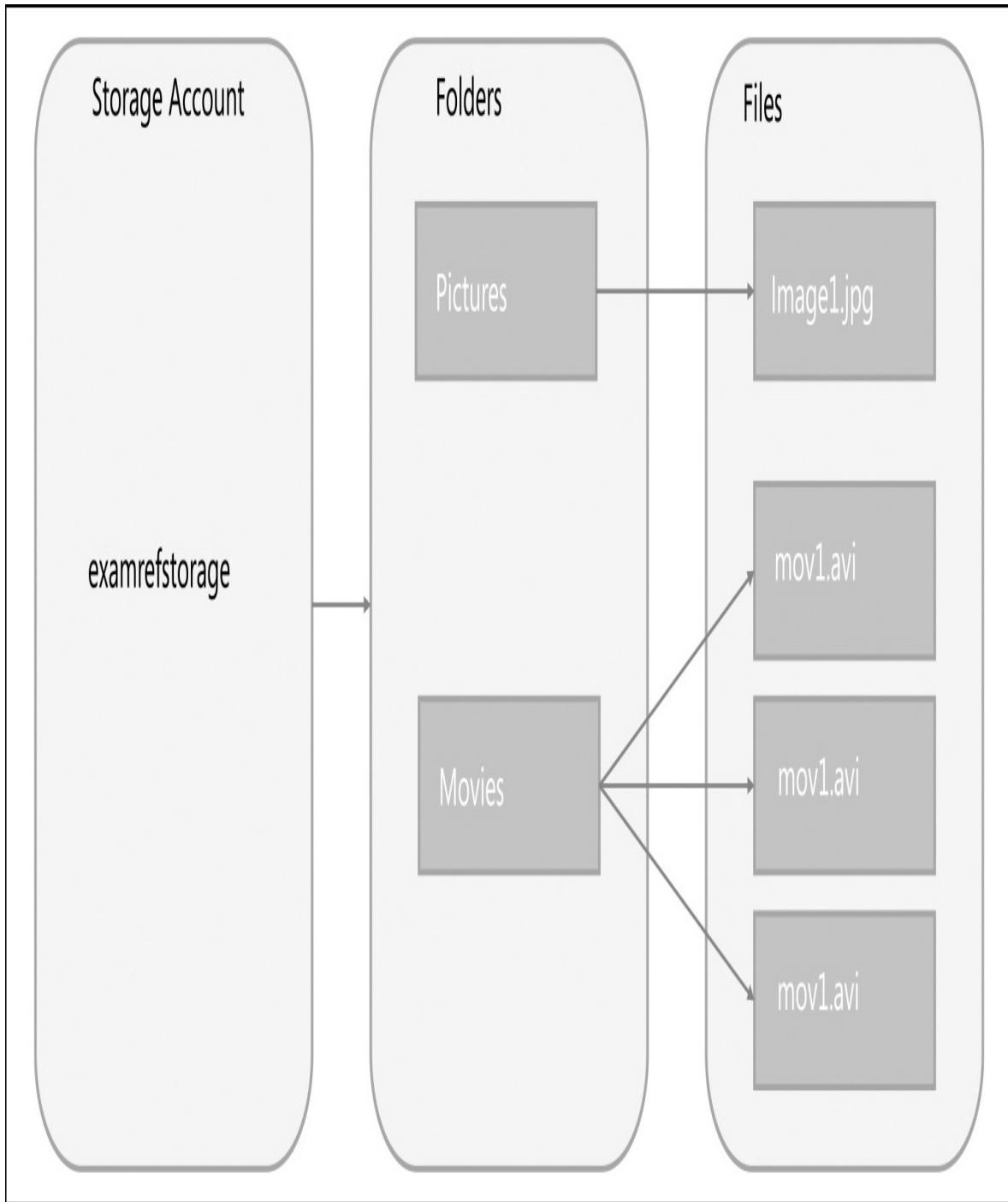


FIGURE 2-30 Azure Files entities and relationship hierarchy

Creating an Azure file share

To create a new Azure file share using the Azure portal, open an Azure Storage account, click the **File Shares** link, and then click the **+ File Share** button. In the

dialog box shown in [Figure 2-31](#), you must provide the file share name and the quota size, which can be a maximum size of 5,120 GiB.

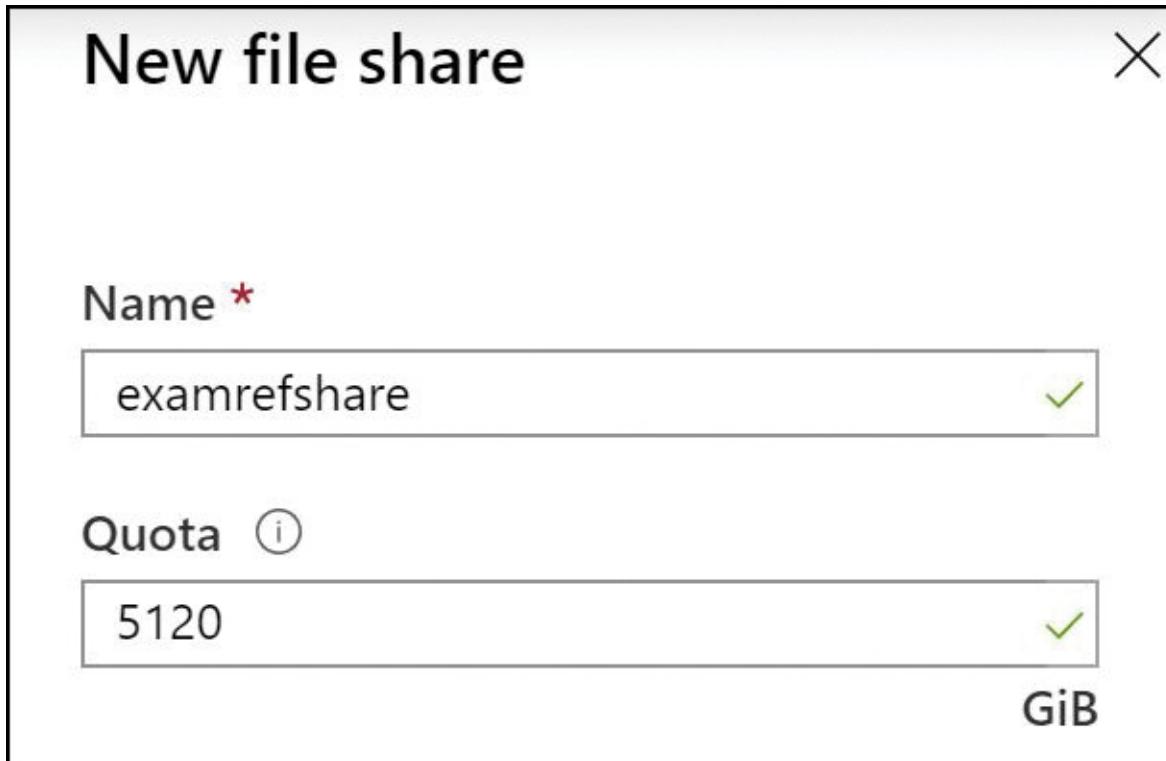


FIGURE 2-31 Adding a new share with Azure files

Connecting to Azure Files outside of Azure

Because Azure Files provides support for SMB 3.0, it is possible to connect directly to an Azure file share from a computer running outside of Azure. In this case, remember to open the outbound TCP port 445 in your local network. Many companies block 445 because of the insecure nature of SMB 1.0. Please check your network connections if you have problems connecting. Alternatively, you can leverage virtual private network or ExpressRoute where port 445 can't be unblocked. Note that Windows 7 and Windows Server 2008 R2 do not support SMB 3.0.

More Info How to remove SMB v1

In order to disable SMB v1 from your environment, you can disable the smb1protocol feature. See following link for more details.

<https://docs.microsoft.com/windows-server/storage/file-server/troubleshoot/detect-enable-and-disable-smbv1-v2-v3#how-to-gracefully-remove-smb-v1-in-windows-81-windows-10-windows-2012-r2-windows-server-2016-and-windows-server-2019>.

Connect and mount with Windows File Explorer

There are several ways to mount an Azure file share from Windows. The first is to use the Map Network Drive feature within Windows File Explorer. Open **File Explorer**, right-click **This PC**, and then click the **Map Network Drive** option, as shown in [Figure 2-32](#).

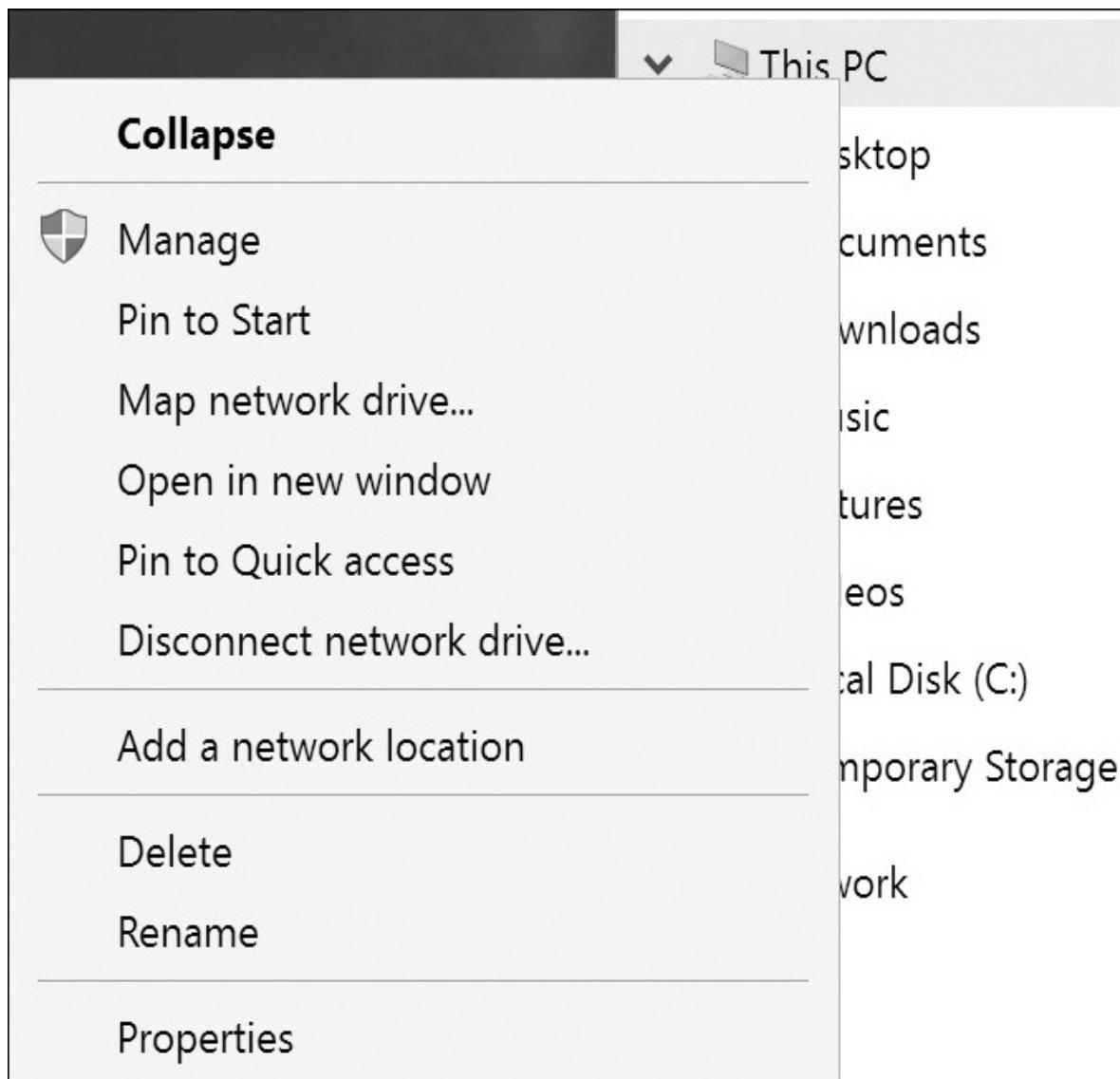


FIGURE 2-32 The Map Network Drive option from This PC

When the dialog box opens, specify the following configuration options, as shown in [Figure 2-33](#):

- **Folder.** \\[name of storage account].files.core.windows.net\[name of share]
- **Connect Using Different Credentials.** Select this option.

X



What network folder would you like to map?

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: Z:

Folder: \\erstandard01.file.core.windows.net\logs

Example: \\server\share

Reconnect at sign-in

Connect using different credentials

Connect to a Web site that you can use to store your documents and pictures.

Finish

Cancel

FIGURE 2-33 Mapping a Network Drive to an Azure file share

When you click **Finish**, you will see another dialog like the one shown in [Figure 2-34](#) that requests the username and password to access the file share. The username should be in the following format: `Azure\ [name of storage account]`, and the password should be the access key for the Azure Storage account.

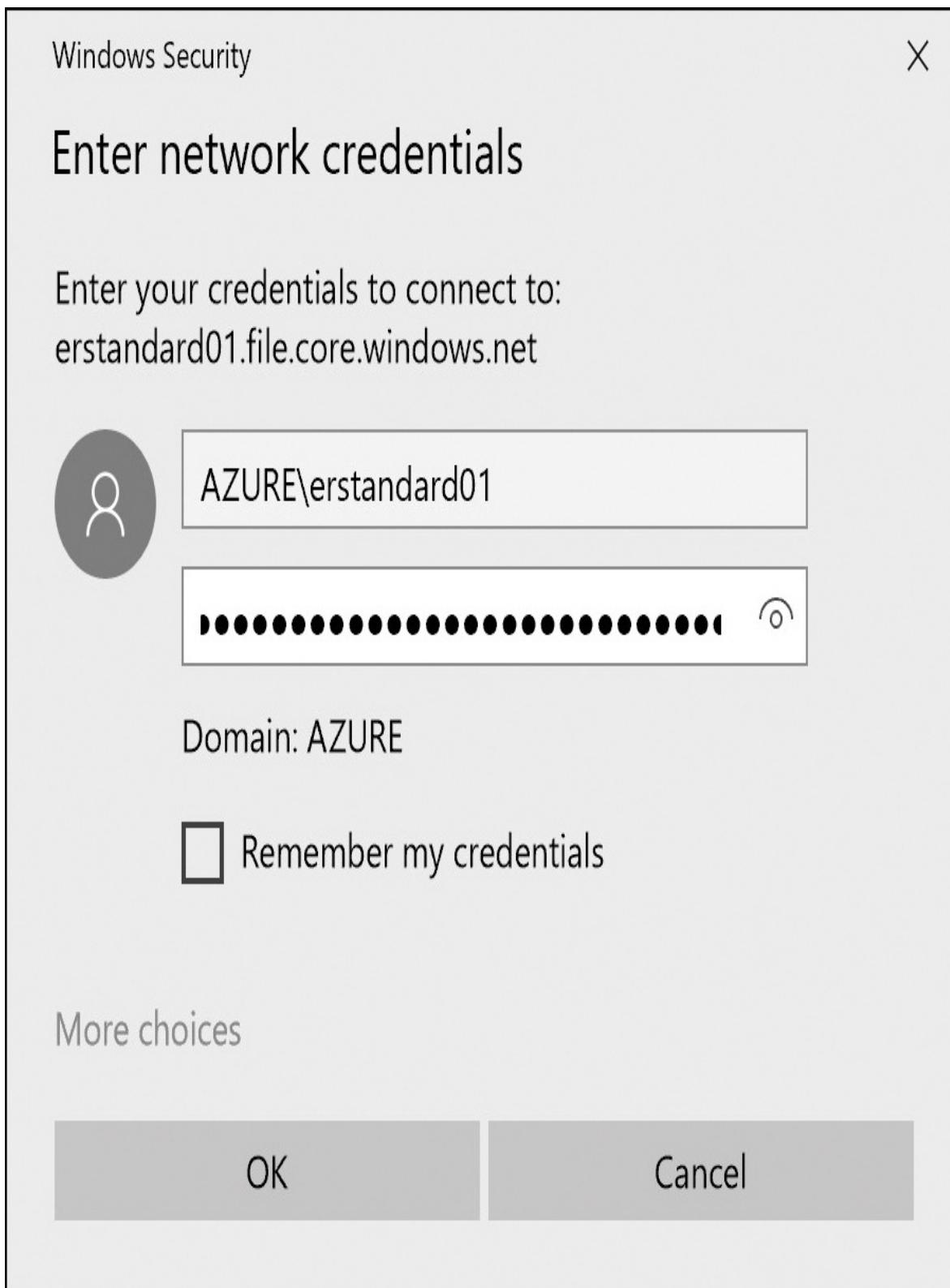


FIGURE 2-34 Specifying credentials to the Azure file share

Connect and mount with the net use command

You can also mount the Azure file share using the Windows `net use` command as the following example demonstrates:

[Click here to view code image](#)

```
net use x \\erstandard01.file.core.windows.net\logs  
/u:AZURE\erstandard01  
r21Dk4qgY1HpcbriySWrBxnXnbedZLmnRK3N49Pfail1t3ragpQaIB7FqK5zbez/sMn  
DEzEu/dgA9Nq/W7IF4A==
```

Automatically reconnect after reboot in Windows

To make the file share automatically reconnect and map to the drive after Windows is rebooted, use the following command (ensuring you replace the placeholder values):

[Click here to view code image](#)

```
cmdkey /add:<storage-account-name>.file.core.windows.net  
/user:AZURE\<storage-  
account-name> /pass:<storage-account-key>  
  
net use Z: \\<storage-account-name>.file.core.windows.net\<file-  
share-name> /  
persistent:yes
```

Connect and mount from Linux

Use the `mount` command (elevated with `sudo`) to mount an Azure file share on a Linux virtual machine. In this example, the `logs` file share would be mapped to the `/logs` mount point.

[Click here to view code image](#)

```
sudo mount -t cifs //<storage-account-  
name>.file.core.windows.net/logs /logs -o  
vers=3.0,username=<storage-account-name>,password=<storage-  
account-key>,  
dir_mode=0777,file_mode=0777,sec=ntlmssp
```

Create and configure Azure File Sync service

Azure File Sync extends Azure Files to allow on-premises file services to be extended to Azure while maintaining performance and compatibility.

Some of the key functionality Azure File Sync provides:

- **Multi-site access** The ability to write files across Windows and Azure Files.
- **Cloud tiering** Storage only recently accessed data on local servers. The rest of the data gets tiered to Azure in a storage account.
- **Azure Backup integration** Backup in the cloud.
- **Fast disaster recovery** Restore file metadata immediately and recall as needed.

Create the Storage Sync Service instance in the portal by navigating to **Create A Resource** and then search for **Azure File Sync**. The creation blade requires the name of the **Storage Sync Service**, the **Subscription**, **Resource Group**, and the **Region** to create the Azure File Sync Service, as shown in [Figure 2-35](#).

Deploy Azure File Sync

Basics * Tags Review + create

Azure File Sync in combination with Azure file shares allows you to centralize your organization's file shares in Azure, while keeping the flexibility, performance, and compatibility of an on-premises file server. [Learn more](#)



Storage Sync Service

Microsoft

Deploying this storage sync service resource will allow you to transform your Windows Server into a quick cache for Azure file shares with optional cloud tiering and multi-server sync functionality. Keep in mind that servers registered to different storage sync service resources cannot exchange data with each other. It's best to register all servers to the same storage sync service if they will ever have a need to sync the same Azure file share.

Subscription *

Visual Studio Ultimate with MSDN



Resource group *

(New) examrefRG



[Create new](#)

Storage sync service name *

examrefStorageSync



Region *

(Canada) Canada East



FIGURE 2-35 Deploy Azure File Sync—Basic blade

Create Azure sync group

You can create a sync group to define the topology for how your file synchronization will take place. Within a sync group, you will add server endpoints, which are file servers and paths within the file server you want the sync group to sync with each other. Figure 2-36 shows the settings for creating a sync group using the Azure portal.

Deploying the Azure File Sync agent

To add endpoints to your Azure Files sync group, Internet Explorer Enhanced Security configuration must be disabled before installing the agent. It can be re-enabled after the initial installation.

1. Install the latest Azure PowerShell module on the server. See the following for installation instructions: <https://go.microsoft.com/fwlink/?linkid=856959>.

Sync group

□ X

Start by specifying an Azure file share to sync with - this is the sync group's first cloud endpoint.

You can specify a folder on your servers you want to sync later.

[Learn more](#)

Sync group name

examrefSyncGroup



1st Cloud endpoint

Subscription

Visual Studio Ultimate with MSDN



Storage account

Select storage account

/subscriptions/00b72028-9dce-4729-9b2a-a10e9205... ✓



If you have previously configured Azure Data Box to import data to the cloud, do not specify a file share that contains that data. Instead, use an empty file share as a cloud endpoint.

Specify the share with Azure Data Box content when you add the server endpoint later.

Azure File Share

examrefshare



Create

Cancel

FIGURE 2-36 Creating a Sync Group and specifying the Azure file share

2. Register a server to the sync group by installing the Azure File Sync Agent on each server. The agent can be downloaded from the Microsoft Download Center at <https://go.microsoft.com/fwlink/?LinkId=858257>. The installer is pictured in Figure 2-37.

Azure File Sync - Server Registration

- X



Sign in and register this server

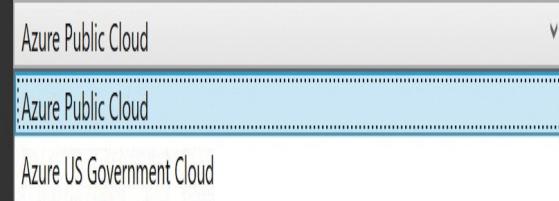
Sign in to Azure to register with an existing Storage Sync Service.
To create a new Storage Sync Service, go to the Azure portal.

I am signing in as a Cloud Solution Provider



Tenant ID

Azure Environment



Sign in

FIGURE 2-37 Installing the Azure File Sync Agent

3. After the agent is installed, sign in with the Azure credentials for your subscription, as shown in [Figure 2-38](#).

Azure File Sync - Server Registration

- X

Sign in to your account X



Sign in

Email, phone, or Skype

To crea

No account? [Create one!](#)

Can't access your account?

I am si

[Sign-in options](#)

Tenan

Next

Azure

Azure

orange Sync Service.
e Azure portal.



v

[Terms of use](#) [Privacy & cookies](#) ...

Sign in

FIGURE 2-38 Signing into the Azure Storage Sync Agent

4. Next, register the server with the Storage Sync Service, as shown in [Figure 2-39](#).

Azure File Sync - Server Registration

- X

Choose a Storage Sync Service

Azure Subscription

Visual Studio Ultimate with MSDN

Subscription ID: 00b72028-9dce-4729-9b2a-a10e92054447

Resource Group

ExamRefRg

Storage Sync Service

examreffilesync

Register

FIGURE 2-39 Registering the server with the Storage Sync Service

Adding a server endpoint

After the server is registered, you must navigate back to the sync group in the Azure portal and click **Add Server Endpoint**. In the **Registered Server** drop-down menu, you will find all the servers that have the agent installed and associated with this sync service.

Enable cloud tiering to only store frequently accessed files locally on the server while all your other files are stored in Azure Files. This is an optional feature that is configured by a policy.

More Info Cloud Tiering Overview

You can learn more about configuring cloud tiering at

<https://docs.microsoft.com/azure/storage/files/storage-sync-cloud-tiering>.

[Figure 2-40](#) shows the blade in the Azure portal to add the server endpoint. Ensure that you are only syncing the location to one sync group at a time and that the path entered exists on the server.

Add server endpoint

A server endpoint integrates an entire volume or a subfolder of a volume from a registered server as a location to sync. The following considerations apply:

- Servers must be registered to the storage sync service that contains this sync group before you can add a location on them here.
- A specific location on the server can only sync with one sync group. Syncing the same location or even a part of it – with a different sync group doesn't work.
- Make sure that the path you specify for this server is correct.

[Learn more](#)

Registered Server

ExamRefFS



Path

D:\Data



^ Cloud Tiering

Enabled

Disabled

Cloud Tiering transforms your server endpoint into a cache for your files in the Azure file share. Different policies help you to fine tune your cache behavior.

[Learn more](#)

Always preserve the specified percentage of free space on the volume:

Specify the percentage of free space



Cache files that were accessed within the specified number of days: [?](#)

Specify the number of days

Create

Cancel

FIGURE 2-40 Adding a server endpoint to the Azure Storage Sync Service.

Monitoring synchronization health

Open the sync group in the Azure portal. A health indicator is displayed by each of the server endpoints; green indicates a healthy status. Click the endpoint to see stats such as the number of files remaining, size, and any resulting errors, as shown in Figure 2-41

More Info Troubleshooting Azure File Sync

Keep up with the latest issues and learn more about troubleshooting Azure File Sync at <https://docs.microsoft.com/azure/storage/files/storage-sync-files-troubleshoot>.

The screenshot shows the Azure portal interface for managing a Sync group named 'examrefsyncgroup'. At the top, there are buttons for 'Add cloud endpoint', 'Add server endpoint', 'Refresh', and 'Delete'. Below this, the 'cloud endpoints' section shows 1 entry for 'examreffiles' under 'AZURE FILE SHARE', which is in a 'Provisioning state' of 'Pending' and belongs to the 'ExamRefRG' resource group. In the 'server endpoints' section, there is 1 entry for 'ExamRefFS' with a 'Health' status of 'Pending', a 'Path' of 'C:\Data', and the last status update was 'Never'.

AZURE FILE SHARE	PROVISIONING STATE	RESOURCE GROUP
examreffiles	Pending	ExamRefRG

SERVER	HEALTH	FILES NOT SYNCING	SYNC ACTIVITY	PATH	CLOUD TIERING	LAST STATUS
ExamRefFS	Pending			C:\Data		Never

FIGURE 2-41 Monitoring the health of a new server endpoint

Configure Azure Blob Storage

This section describes the key features of the blob storage provided by each storage account. Azure Blob Storage is used for large-scale storage of arbitrary

data objects, such as media files, log files, and so on.

Blob containers

[Figure 2-42](#) shows the layout of the blob storage. Each storage account can have one or more blob containers and all blobs must be stored within a container. Containers are similar in concept to a hard drive on your computer, in that they provide a storage space for data in your storage account. Within each container, you can store blobs, much as you would store files on a hard drive. Blobs can be placed at the root of the container or organized into a folder hierarchy.

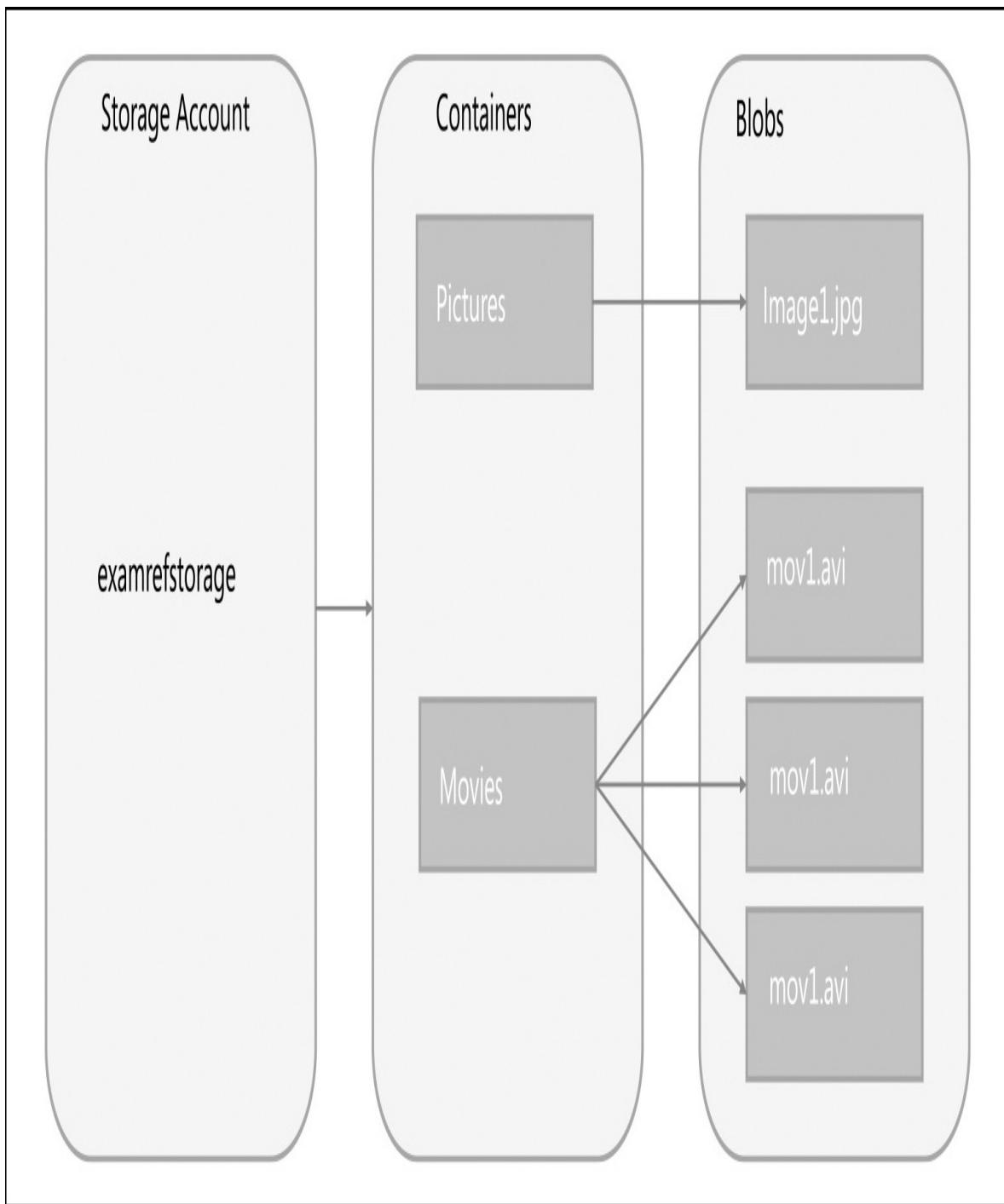


FIGURE 2-42 Azure storage account entities and hierarchy relationships

Each blob has a unique URL. The format of this URL is as follows:
`https://[account name].blob.core.windows.net/[container name]/[blob path and name]`.

Optionally, you can create a container at the root of the storage account, by specifying the special name `$root` for the container name. This allows you to store blobs in the root of the storage account and reference them with URLs such as: `https://[account name].blob.core.windows.net/fileinroot.txt`.

Understanding blob types

Blobs come in three types, and it is important to understand when each type of blob should be used and what the limitations are for each.

- **Page Blobs.** Optimized for random-access read and write operations. Page Blobs are used to store virtual disk (VHD) files which using unmanaged disks with Azure virtual machines. The maximize Page Blob size is 8 TB.
- **Block Blobs.** Optimized for efficient uploads and downloads, for video, images, and other general-purpose file storage. The maximum Block Blob size is slightly more than 4.75 TB.
- **Append Blobs.** Optimized for append operations. Updating or deleting existing blocks in the blob is not supported. Up to 50,000 blocks can be added to each Append Blob, and each block can be up to 4MB in size, giving a maximum Append Blob size of slightly more than 195 GB. Page Blobs are most commonly used for log files.

Blobs of all three types can share a single blob container.



Exam Tip

The type of the blob is set at creation and cannot be changed after the fact. A common problem that might show up on the exam is if a `.vhd` file was accidentally uploaded as a Block Blob instead of a Page Blob. The blob must be deleted first and reuploaded as a Page Blob before it can be mounted as an OS or data disk to an Azure VM.

More Info Blob types

You can learn more about the intricacies of each blob type here:
<https://docs.microsoft.com/rest/api/storageservices/understanding-block-blobs--append-blobs--and-page-blobs>.

Managing blobs and containers (Azure portal)

You can create and manage containers through the Azure portal, Azure Storage Explorer, third-party storage tools, or through the command-line tools. To create a container in the Azure management portal, open a storage account by clicking **All Services > Storage Accounts**, and then choosing your storage account. Within the storage account blade, click the **Blobs** tile, and then click the **+ Container** button, as shown in [Figure 2-43](#). See [Skill 2.1](#) for more information on setting the public access level.

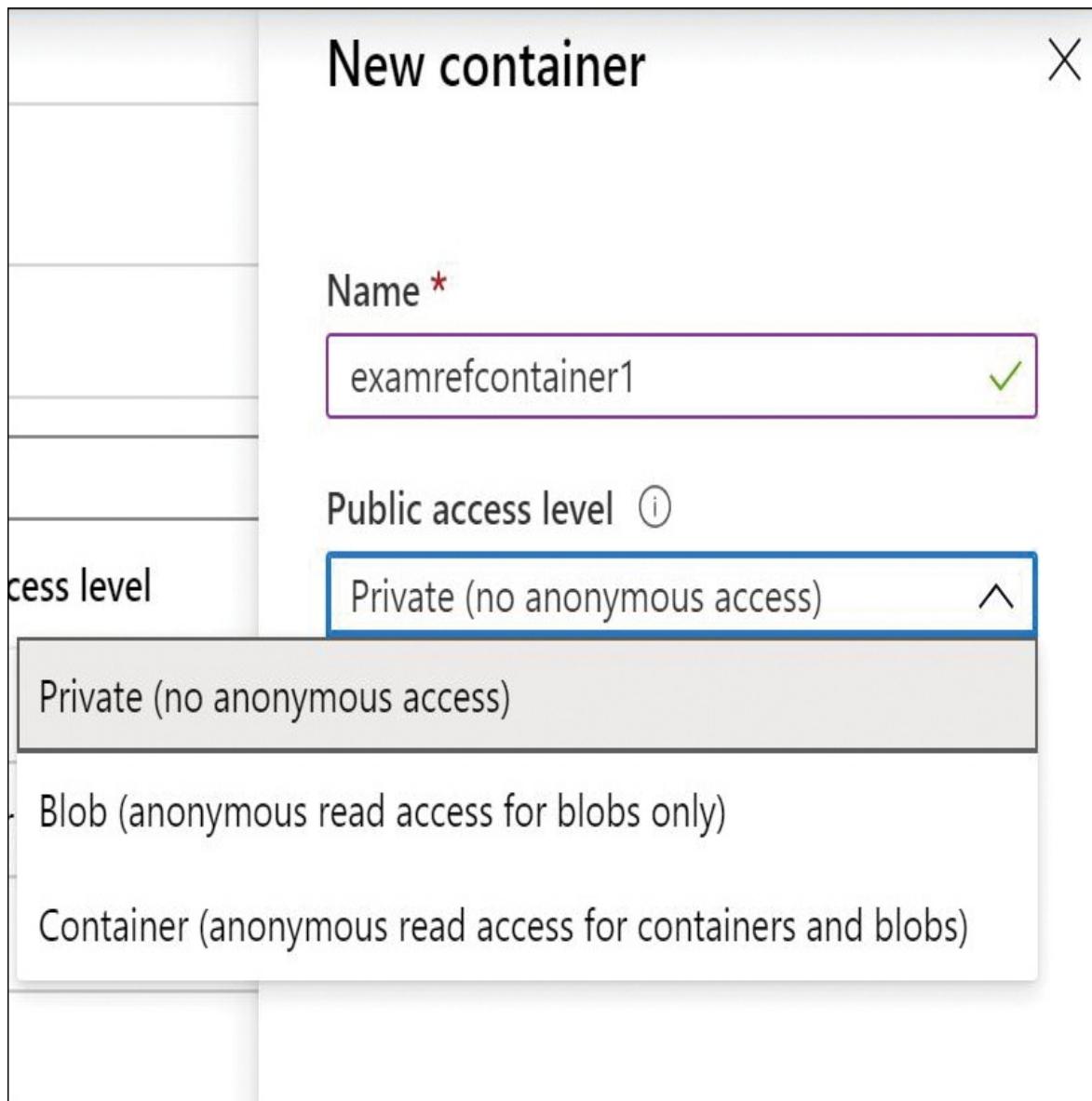


FIGURE 2-43 Creating a container using the Azure management portal

After a container is created, you can also use the portal to upload blobs to the container, as demonstrated in [Figure 2-44](#). Click the **Upload** button in the container and then browse to the blob to upload. If you click the **Advanced** button, you can select the blob type (Blob, Page, or Append), the block size, and optionally, a folder to which the blob is to be uploaded.

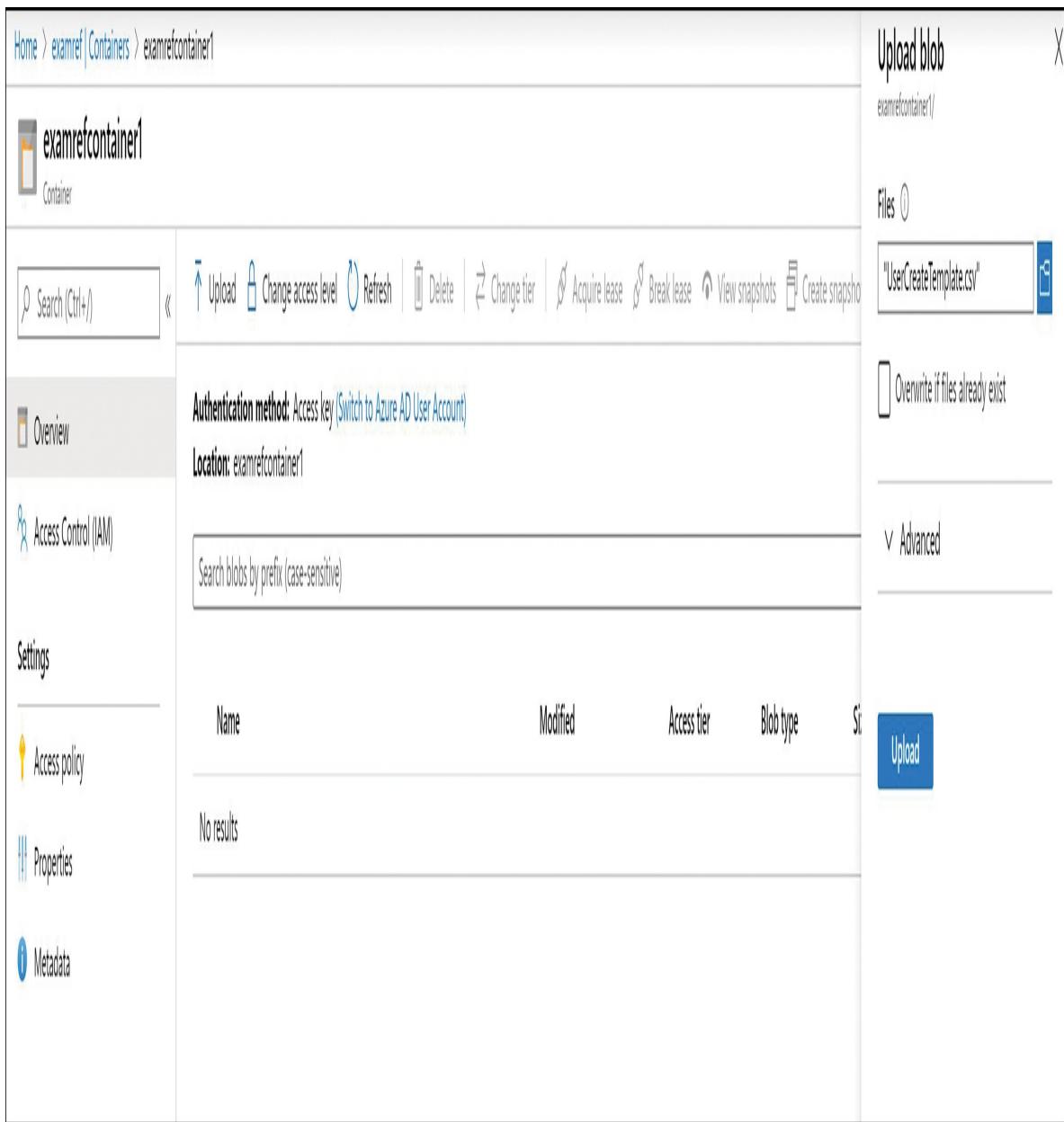


FIGURE 2-44 Uploading a blob to a storage account container

More Info Managing Blob Storage with PowerShell

The Azure PowerShell cmdlets offer a rich set of capabilities for managing blobs in storage. You can learn more about their capabilities here:

<https://docs.microsoft.com/azure/storage/blobs/storage-how-to-use-blobs-powershell>.

More Info Managing Blob Storage with the Azure CLI

The Azure CLI also offers a rich set of capabilities for managing blobs in storage. You can learn more about their capabilities here:

<https://docs.microsoft.com/azure/storage/common/storage-azure-cli>.

Managing blobs and containers (Storage Explorer)

Azure Storage Explorer provides rich functionality for managing storage data, including blobs and containers. To create a container, expand the **Storage Accounts** node, expand the storage account you want to use, and right-click the **Blob Containers** node. This will open a new menu item where you can create a blob container, as shown in [Figure 2-45](#).

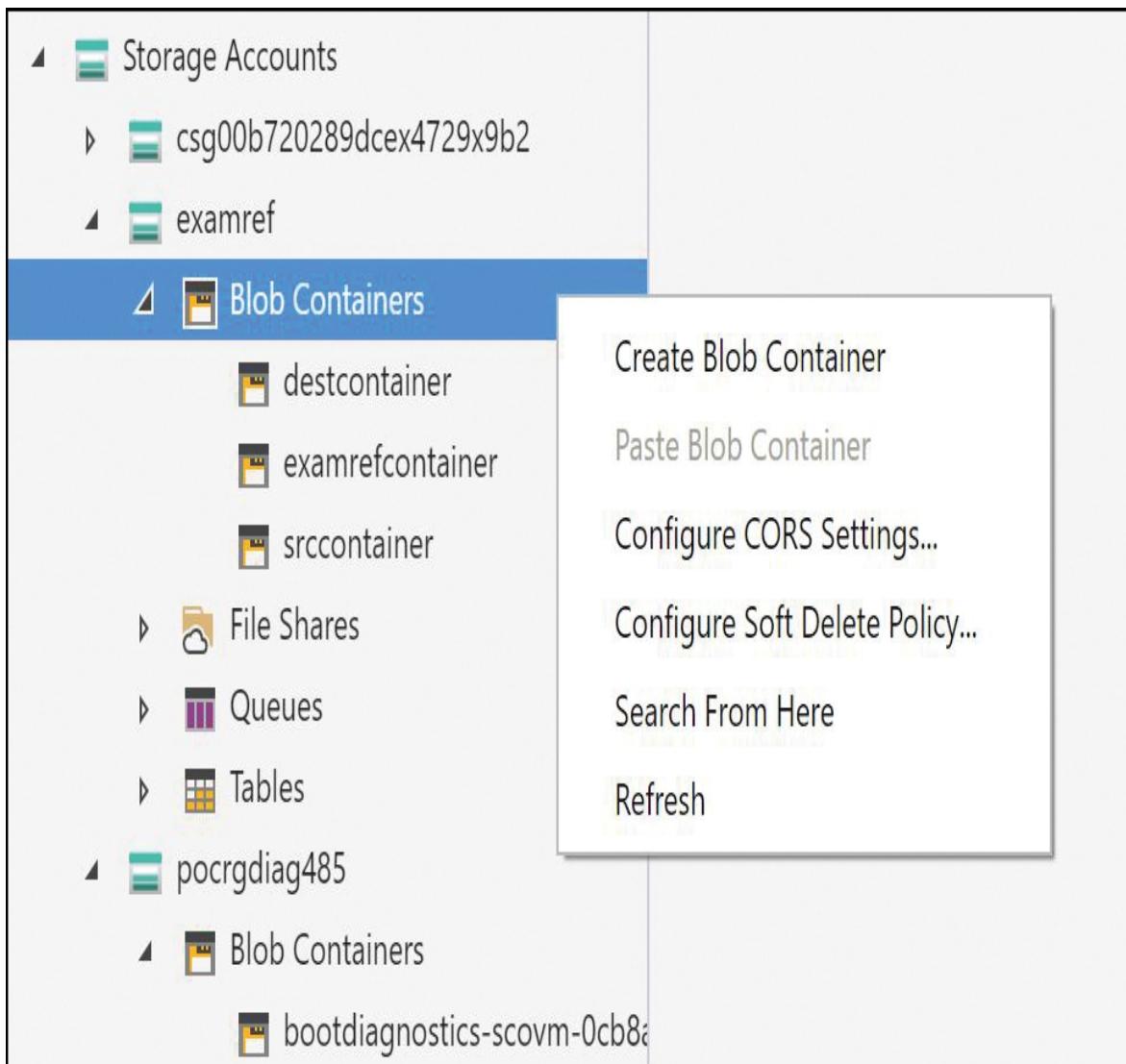


FIGURE 2-45 Creating a container using the Azure Storage Explorer

Azure Storage Explorer provides the ability to upload a single file or multiple files at once. The **Upload Folder** feature provides the ability to upload the entire contents of a local folder, re-creating the hierarchy in the Azure Storage Account. [Figure 2-46](#) shows the two upload options.

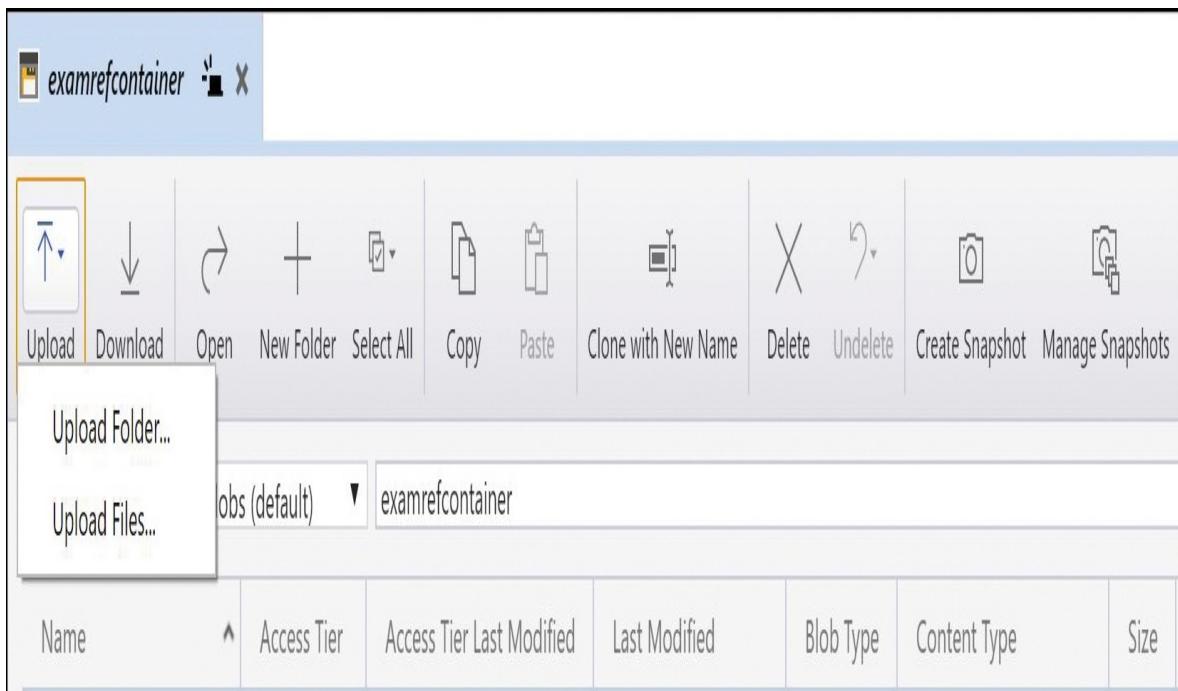


FIGURE 2-46 Uploading files and folders using Azure Storage Explorer

Soft Delete for Azure Storage blobs

The default behavior of deleting a blob is that the blob is deleted and lost forever. Soft Delete is a feature that allows you to save and recover your data when blobs or blob snapshots are deleted even in the event of an overwrite. This feature must be enabled on the Azure Storage account, and a retention period must be set for how long the deleted data is available (see [Figure 2-47](#)).

Home > examref | Data protection

examref | Data protection

Storage account

Search (Ctrl+ /) Save Discard Refresh

Blob service

Containers Custom domain Data protection Azure CDN Add Azure Search Lifecycle Management

Data protection provides options for recovering your data when it is erroneously modified or deleted.

Blob soft delete ⓘ

Disabled Enabled

Retention policies

Blob retainment period in days ⓘ

30 days

i You must opt-in on a per-subscription basis to utilize container soft delete. [Opt-in for container soft delete ↗](#)

File service

File shares

The screenshot shows the Azure Storage account 'Data protection' settings. On the left, there's a sidebar with links like 'Containers', 'Custom domain', 'Data protection' (which is selected and highlighted in grey), 'Azure CDN', 'Add Azure Search', and 'Lifecycle Management'. The main area has a search bar and navigation buttons ('Save', 'Discard', 'Refresh'). It displays information about blob soft delete, retention policies, and a blob retainment period of 30 days. A note at the bottom says you must opt-in for container soft delete on a per-subscription basis.

FIGURE 2-47 Enabling soft delete on an Azure Storage account



Exam Tip

The maximum retention period for soft delete is 365 days.

More Info Soft delete for Azure Storage blobs

You can learn more about using Soft Delete with Azure Blob Storage here:
<https://docs.microsoft.com/en-us/azure/storage/blobs/soft-delete-blob-overview>

Configure storage tiers for Azure blobs

As discussed in [Skill 2.1](#), Azure Blob Storage supports three access tiers: Hot, Cool, and Archive. Each represents a trade-off of performance, availability, and cost. There is no trade-off on the durability (probability of data loss), which is extremely high across all tiers.

Account-level tiering

The storage account blobs can coexist between three tiers within the same account. If any blob does not have an assigned tier, it infers the access tier from the account access tier setting by default. In such a scenario, you will see that the access tier's **Inferred** blob property is set to `true`, and the Access Tier blob property matches the account level tier. In the Azure portal, the **Inferred** property for the Access Tier is displayed, as shown in [Figure 2-48](#).

examrefcontainer
Container

Search (Ctrl+ /) | Upload | Change access level | Refresh | Delete | Change tier | Acquire lease | Break lease | View snapshots | Create snapshot

Authentication method: Access key ([Switch to Azure AD User Account](#))
Location: examrefcontainer

Search blobs by prefix (case-sensitive)

Name	Modified	Access tier	Blob type	Size
Technical.doc	5/9/2020, 9:49:33 AM	Hot	Block blob	44.5 kB
UserCreateTemplate.csv	5/8/2020, 10:52:13 PM	Hot (Inferred)	Block blob	682 B

FIGURE 2-48 Access tier property for account level tiering

Note Change Account Access Tier

Changing the account access tier applies to all access tier-inferred objects stored in the account that don't have an explicit tier set.

Blob-level tiering

The blobs can be assigned with the desired access tier while you upload them to the container (see [Figure 2-49](#)). You can also change access tier among the Hot, Cool, or Archive tiers (because usage patterns change) without having to move data between accounts. All requests to change tier will take place immediately between Hot and Cool tiers.

Note Archive storage tier

Data in the Archive storage tier is stored offline and must be rehydrated to the Cool or Hot tier before it can be accessed. This process can take up to 15 hours.

When the access tier is changed, the access tier's **Last Modified** property will be updated with the time when recent change is made to the tier (see [Figure 2-50](#)).

Upload blob

X

examrefcontainer/

Files ⓘ

"Technical.doc"



Overwrite if files already exist

Advanced

Authentication type ⓘ

Azure AD user account **Account key**

Blob type ⓘ

Block blob



Upload .vhdx files as page blobs (recommended)

Block size ⓘ

4 MB



Access tier ⓘ

Hot (Inferred)



Hot (Inferred)

Hot

Cool

Archive

Upload

FIGURE 2-49 Changing access tier while uploading the blobs to container

Change access tier

Changing access tier can occur either at account level or at individual blob level. At the account level, it can be performed by setting the access tier in the **Configuration** blade (the default option unless assigned it explicitly) or by using the new **Lifecycle Management** feature. At an individual blob level, the same can be achieved by using the **Change Tier** option for the blob.

Technical.doc

Blob

Save Discard Download Refresh | Delete | Change tier Acquire lease Break lease

Overview Snapshots Edit Generate SAS

Properties

URL <https://examref.blob.co...>

LAST MODIFIED 5/9/2020, 9:49:33 AM

CREATION TIME 5/9/2020, 9:49:33 AM

VERSION ID -

TYPE Block blob

SIZE 44.5 KiB

ACCESS TIER Hot

ACCESS TIER LAST MODIFIED 5/9/2020, 9:52:45 AM

SERVER ENCRYPTED true

ETAG 0x8D7F41FCDEB43A8

CONTENT-TYPE application/msword

CONTENT-MD5 EprmeoXJmr3VlKocfF03Bw==

LEASE STATUS Unlocked

LEASE STATE Available

LEASE DURATION -

COPY STATUS -

COPY COMPLETION TIME -

Undelete

Metadata

Key	Value
<input type="text"/>	<input type="text"/>

FIGURE 2-50 Access tier Last Modified property for the blob

Note Changing the access tier

Changing the account access tier will result in tier change charges for access tier-inferred blobs stored in the account that does not have an explicit tier set.

To make a change at account level, browse to the storage account, click the **Configuration** blade, and change **Access Tier(Default)** to **Cool** or **Hot** (see [Figure 2-51](#)).

Similarly, to make a change at blob level, browse to a blob, then click the **Change Tier** option, and select the **Access Tier** from the drop-down menu; your options are **Hot**, **Cool**, or **Archive** (see [Figure 2-52](#)).



examref | Configuration

Storage account

Search (Ctrl+ /)

Save Discard

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Private endpoint connections

Advanced security

Static website

Properties

Locks

Export template

The cost of your storage account depends on the usage and the options you choose below.

[Learn more](#)

Account kind

StorageV2 (general purpose v2)

Performance ⓘ

Standard Premium

Secure transfer required ⓘ

Disabled Enabled

Access tier (default) ⓘ

Cool Hot



Changing the access tier may result in additional charges. Click here to learn more.

Replication ⓘ

Read-access geo-redundant storage (RA-GRS)

Large file shares ⓘ

Disabled Enabled



The current combination of storage account kind, performance, replication and location does [Learn more](#).

Identity-based access for file shares

Azure Active Directory Domain Service (AAD DS) ⓘ

Disabled Enabled

Active Directory (AD) ⓘ

[How to domain join this storage account](#)

FIGURE 2-51 Change the access tier

Configure blob Lifecycle Management

Azure Storage has a lifecycle-management capability, and it can be used to transition data to lower-access tiers automatically based on pre-configured rules. You can also delete the data at the end of its lifecycle. These rules can be executed against the storage account once per day. Specific blobs and containers can be targeted using filter sets.

To configure the Lifecycle Management rules, open the storage account, browse to **Lifecycle Management** under **Blob Service**, and click **Add A Rule** (see [Figure 2-53](#)). You can define up to 100 rules.

UserCreateTemplate.csv

Blob

Save Discard Download Refresh Delete Change tier A

Overview Snapshots Edit Generate SAS

Properties

URL	https://examref.blob.co...
LAST MODIFIED	5/8/2020, 10:52:13 PM
CREATION TIME	5/8/2020, 10:52:13 PM
VERSION ID	-
TYPE	Block blob
SIZE	682 B
ACCESS TIER	Hot (Inferred)
ACCESS TIER LAST MODIFIED	N/A
SERVER ENCRYPTED	true
ETAG	0x8D7F3C3FA2453ED
CONTENT-TYPE	application/vnd.ms-excel
CONTENT-MD5	I1A6qKviAX4a5lOQUAkVg==
LEASE STATUS	Unlocked
LEASE STATE	Available
LEASE DURATION	-
COPY STATUS	-
COPY COMPLETION TIME	-

Metadata

Key	Value
<input type="text"/>	<input type="text"/>

Change tier

Optimize storage costs by placing your data in the appropriate access tier. [Learn more about Azure blob storage access tiers](#)

Access tier

Cool

Hot (Inferred)

Cool

Archive

FIGURE 2-52 Change Tier on the Configuration blade

The screenshot shows the Azure Storage Account Lifecycle Management blade for a storage account named 'examref'. The top navigation bar includes 'Home > Microsoft.StorageAccount-20201207222950 > examref'. The main title is 'examref | Lifecycle Management' with a 'Storage account' icon. A search bar and a toolbar with 'Add a rule', 'Enable', 'Disable', 'Refresh', and 'Delete' buttons are visible. On the left, a sidebar lists 'Blob service' options: 'Containers', 'Custom domain', 'Data protection', 'Object replication', 'Azure CDN', 'Add Azure Search', 'Lifecycle Management' (which is selected and highlighted in grey), and 'Blob inventory (preview)'. The central area contains a message about lifecycle management rules and two view options: 'List View' (selected) and 'Code View'. Below this, a section titled 'Access tracking (preview) disabled' shows a toggle switch that is off. A table at the bottom lists 'No rules' with columns for 'Name' and 'Status'.

Name	Status
No rules	

FIGURE 2-53 Add Rule option on the Lifecycle Management blade

You can limit the rule scope with a filter set by selecting **Limit Blobs With Filters**, as shown in [Figure 2-54](#). You can also select the **Blob Type** and **Blob Subtype** that should be applicable to this rule. Under **Blob Type**, you can choose **Block Blobs** or **Append Blobs**. Under **Blob Subtype**, your choices are **Base Blobs**, **Snapshots**, and **Versions**. Now click **Next**.

Add a rule

1 Details **2** Base blobs **3** Filter set

A rule is made up of one or more conditions and actions that apply to the entire storage account. Optionally, specify that rules will apply to particular blobs by limiting with filters.

Rule name *

ExamRefRule

Rule scope *

Apply rule to all blobs in your storage account

Limit blobs with filters

Blob type *

Block blobs

Append blobs

Blob subtype *

Base blobs

Snapshots

Versions

FIGURE 2-54 Setting rule options on the Add A Rule blade

You can configure rules in the **Base Blobs** section to define the blob lifecycle policy. You can create multiple if-then blocks to define the conditions. For example, you can move blob data to Cool storage if it is not modified for specified number of days. Similarly, you can also create rules to move blobs to archive storage or delete them if not modified for defined number of days. In [Figure 2-55](#), the condition has been created for 30 days, and all three actions are shown in the drop-down menu. Click **Next** to configure the filter set.

In the **Filter Set**, you can specify the prefix to find items within the container. You need to specify the container name/prefix. For example, you could choose use `data/cost` where `data` is the name of the container and `cost` is the prefix, as shown in [Figure 2-56](#). You can also use the **Blob Index Match** if you have indexed the items with keys and values in your containers. You can specify up to 10 various prefixes per rule.

Add a rule

Details 2 Base blobs 3 Filter set

Lifecycle management uses your rules to automatically move blobs to cooler tiers or to delete them. If you create multiple rules, the associated actions must be implemented in tier order (from hot to cool storage, then archive, then deletion).

+ Add if-then block

If Delete

Base blobs were *

Last modified

More than (days ago) *

30

↓

Then

Move to cool storage *

Move to cool storage
This is the most reliable option if cost is not a priority.

Move to archive storage
Archive storage does not fully delete the blob. However, it cannot be moved back to cool storage.

Delete the blob
This is the most efficient option if backing up a blob is not a priority.

FIGURE 2-55 Base Blobs section for a rule

Add a rule

Details Base blobs 3 Filter set

Prefix match

A prefix match will find items like folders and blobs that start with the specified input. For example, inputting "a" would return any folders or blobs that start with "a". To find items in a specific container, enter the name of the container first, and then provide the desired prefix query for any contents within the container, for example: "myContainer/prefix".

data/cost



Enter a prefix or file path such as "myContainer/prefix"

Blob index match

If you have indexed items in containers with keys and values, you can filter for them.

Key

Value

Enter an index key

=
=

Enter a value

FIGURE 2-56 Filter Set for a rule

Now click **Add** to create the rule.

Once created, the consolidated view of the code is shown on the **Code View** tab, as shown in [Figure 2-57](#).

examref | Lifecycle Management

Storage account

Search (Ctrl+ /)

«

Save Discard Download Refresh



CORS



Configuration



Encryption



Shared access signature



Networking



Security



Static website



Properties



Locks

Blob service



Containers



Custom domain



Data protection



Object replication



Azure CDN



Add Azure Search



Lifecycle Management



Blob inventory (preview)

Lifecycle management offers a rich, rule-based policy for general purpose v2 and blob storage accounts.

List View

Code View

```
1  {
2      "rules": [
3          {
4              "enabled": true,
5              "name": "ExamRefRule",
6              "type": "Lifecycle",
7              "definition": {
8                  "actions": {
9                      "baseBlob": {
10                         "tierToCool": {
11                             "daysAfterModificationGreaterThanOrEqual": 30
12                         }
13                     }
14                 },
15                 "filters": {
16                     "blobTypes": [
17                         "blockBlob"
18                     ],
19                     "prefixMatch": [
20                         "data/cost"
21                     ]
22                 }
23             }
24         }
25     ]
26 }
```

FIGURE 2-57 Action set for a rule under Lifecycle Management blade

Note Lifecycle management Effect

The policy can take up to 24 hours to go into effect, and then the action can take an additional 24 hours to run. Overall, it takes up to 48 hours for policy actions to complete once you set up Lifecycle Management.

You can delete the rule anytime if not required anymore by visiting the **Lifecycle Management** blade.

Thought experiment

In this thought experiment, apply what you have learned about this objective. You can find answers to these questions in the next section.

You are asked to design Azure Storage solution for one of large insurance company. The company wants the data to be accessible based on the role of individual user within the organization. Various departments have their separate dataset which they access on a daily basis. The company wants to restrict users from modifying the data to other departments, but all the users must be able to access the data across departments.

Also, there is a requirement to store that data forever with minimal cost possible. The data is rarely used after two years from the date it was last modified.

- What steps should you take to assign the storage access based on their department?
- What changes need to be made in order to keep storing data forever with minimal cost?

Thought experiment answers

This section contains the solution to the thought experiment for the chapter.

To solve this problem, we can leverage various capabilities of Azure Storage, such as Azure AD authentication with role-based access control and lifecycle-

management for the Blob Storage.

1. Create an Azure Storage account and create container for each department to store its data. Next, you can assign the Storage Blob Data Reader role for all the department groups but assign the Storage Blob Data Contributor role for their department group. This allows users to access all department data, but they can modify only their department data.
2. Create a rule under Lifecycle Management for the storage account and select the Apply Rule To All Blobs In Your Storage Account option. Then, add an if-then block to move data to archive tier after 730 days (two years). This will let you store the data forever with minimal cost in the Archive tier.

Chapter summary

This chapter covered several key services related to implementing storage in Microsoft Azure. Topics included how to create and manage Azure Storage Accounts, Blob Storage, Azure Files, import and exporting data, Storage Explorer, AzCopy, Lifecycle Management, and object replication.

Below are some of the key takeaways from this chapter:

- Azure Storage accounts provide four separate services: Blob Storage, Table Storage, Queue Storage, and Azure Files. It is important to understand the usage scenarios of each service.
- Standard storage accounts use magnetic drives and provide the lowest cost per GB. This type of account is best suited for applications that require bulk storage or where data is accessed infrequently.
- Premium storage accounts use solid state drives and offer consistent, low-latency performance. This type of account can only be used with Azure virtual machine disks and are best for I/O-intensive applications, like databases.
- Storage accounts must specify a replication mode. The options are locally redundant, zone-redundant, geo-redundant, read-access geo-redundant storage, geo zone-redundant, and read-access geo zone-redundant.
- Blob Storage supports three types of blobs (Block, Page, and Append blobs), and three access tiers (Hot, Cool, and Archive).

- There are three kinds of storage accounts: General-Purpose V1, General-Purpose V2, and Blob Storage. The availability of features varies between storage account types.
- Azure Storage can be managed through several tools directly from Microsoft: Azure portal, PowerShell, CLI, Storage Explorer, and AzCopy. It is important to know when to use each tool.
- Access to storage accounts can be controlled using several techniques. Among them are: Azure AD authentication; storage account name and key; SAS; SAS with access policy; and using the storage firewall and virtual network service endpoints. Access to Blob Storage can also be controlled using the public access level of the storage container.
- You can also use AzCopy to copy files between storage accounts or from outside publicly accessible locations to your Azure Storage account.
- Azure Storage has a lifecycle-management capability, and it can be used to transition data to lower-access tiers automatically based on preconfigured rules. You can also delete the data at the end of its lifecycle. These rules can be executed against the storage account once per day. Specific blobs and containers can be targeted using filter sets.
- Azure Storage also provides blob object replication capabilities that provide asynchronous replication of Block Blobs from one storage account to another. The blobs are replicated based on the defined replication rules.

You can leverage object replication only when blob versioning is enabled for both the source and destination storage accounts, and the blob change feed is enabled for the source storage account.

Chapter 3

Deploy and manage Azure compute resources

Microsoft Azure offers many features and services that can be used to create inventive solutions for almost any IT problem. Some of the most common services for designing these solutions are Microsoft Azure virtual machines (VM) and VM scale sets (VMSS). Virtual machines are one of the key compute options for deploying workloads in Microsoft Azure.

The flexibility of virtual machines makes them a key scenario for many workloads. For example, you have a choice of server operating systems with various supported versions of Windows and Linux distributions. Azure virtual machines also provide you full control over the operating system along with advanced configuration options for networking and storage. In addition to VM capabilities, VM scale sets provide the unique ability to scale out certain types of workloads to handle large processing problems, and they optimize cost by only running instances when needed.

In addition to this, you have other compute services, Azure Kubernetes Service (known as AKS) and Azure Container Instances (known as ACI) which are comparatively new in the market. With the wide adoption of containerized workloads across many IT companies, Microsoft is also heavily investing in enhancing their current product set to support container-based workloads. We also have services, such as Azure App Service and its App Service Plans to manage and host Web applications.

In this chapter, you will learn the ins-and-outs of deploying and managing these compute resources in Azure as we cover creation through the Azure portal and the command line tools, automation with templates, as well as core management tasks.

Skills covered in this chapter:

- Skill 3.1: Automate deployment of virtual machines (VMs) by using

Azure Resource Manager templates

- Skill 3.2: Configure VMs for high availability and scalability
- Skill 3.3: Configure VMs
- Skill 3.4: Create and configure containers
- Skill 3.5: Create and configure Azure App Service

Skill 3.1: Automate deployment of virtual machines (VMs) by using Azure Resource Manager templates

The ability to provision virtual machines on-demand using the Azure portal is incredibly powerful. The true power of the cloud, however, is the ability to automatically deploy one or more resources defined in code, such as a script or a template. Use cases such as defining an application configuration, and automatically deploying it on-demand allow teams to be more agile by providing dev, test, or production environments in a fast and repeatable fashion. Because the configuration is stored as code, changes to infrastructure can also be tracked in a version control system. In this skill, you will learn some of the core capabilities for automating workload deployments in Azure.

This section covers how to:

- Modify Azure Resource Manager (ARM) template
- Configure a virtual hard disk (VHD) template
- Deploy from a template
- Save a deployment as an Azure Resource Manager (ARM) template
- Deploy virtual machine extensions

ARM Template Overview

Azure Resource Manager templates are authored using JavaScript Object Notation (JSON) and provide the ability to define the configuration of resources, such as virtual machines, storage accounts, and so on in a declarative manner. Templates go beyond just providing the ability to create the resources; some

resources such as virtual machines also allow you to customize them and create dependencies between them. This allows you to create templates that have capabilities for orchestrated deployments of completely functional solutions.

The Azure team maintains a list of ARM templates with examples for most resources. This list is located at <https://azure.microsoft.com/resources/templates/> and is backed by a source code repository in GitHub. If you want to go directly to the source to file a bug, you can access it at: <https://github.com/Azure/azure-quickstart-templates>.

The basic structure of a resource manager template has most of the following elements:

[Click here to view code image](#)

```
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-  
01/deploymentTemplate.json#",  
  "contentVersion": "1.0.0.0",  
  "parameters": { },  
  "variables": { },  
  "functions": [ ],  
  "resources": [ ],  
  "outputs": { }  
}
```

- **\$schema** The JSON schema file is the reference to the standard structure defined for an ARM template, which can help you determine when something is wrong with your template in comparison to the schema file syntax. The JSON schema will be used by features, such as code completion or Intellisense, which allows you to make changes in the templates easily.

For resource group targeted deployments use:

[Click here to view code image](#)

<https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#>

For subscription targeted deployments use:

[Click here to view code image](#)

<https://schema.management.azure.com/schemas/2019-04-01/subscriptionDeployment>

Template.json#

- `contentVersion` This provides source control to track the changes made in your template. You can provide any value for this element. When deploying resources using the template, this value can be used to make sure that the right template is being used.
- `parameters` Using parameters, we can define the various values that are passed at runtime without changing the exact template file. The parameters can be changed by the `azuredeploy.parameters.json` file or in the PowerShell script that is used to deploy your template. The parameters are key elements when dealing with nested templates to pass the values from parent template to the child templates.
- `variables` Defines values which are used in your template to simplify template language. Mostly, variables are hard-coded values, but they also can be created dynamically using parameters or standard template functions.
- `functions` Users can create functions that can be used within the template. The complex expressions that are being used multiple times in the template can be defined as a function once. You need to create your own namespace and create member functions as needed. You cannot access variables or any other user-defined functions within your function.
- `resources` This contains resource that are deployed or updated in a resource group. You can define the `condition` to control the provisioning of each resource. Also, the `dependsOn` determines which resources must be deployed first before a specific resource.
- `outputs` Here, you can define the type of values that are returned after deployment. This section is used to keep a track of resources that are being deployed or updated.

Defining a virtual network

This skill is focused on learning how to deploy Windows and Linux virtual machines. A pre-requisite of deploying a virtual machine is a virtual network. In [Listing 3-1](#), we will define the structure of the virtual network using several variables that describe the address space and subnet allocation.

LISTING 3-1 Variables for a virtual network creation

[Click here to view code image](#)

```
"ExamRefRGPrefix": "10.0.0.0/16",
"ExamRefRGSubnet1Name": "FrontEndSubnet",
"ExamRefRGSubnet1Prefix": "10.0.0.0/24",
"ExamRefRGSubnet2Name": "BackEndSubnet",
"ExamRefRGSubnet2Prefix": "10.0.1.0/24",
"ExamRefRGSubnet1Ref": "[concat(variables('vnetId'), '/subnets/',
variables('ExamRefRGSubnet1Name'))]",
"VNetId": "[resourceId('Microsoft.Network/virtualNetworks',
variables('VirtualNetwork
Name'))]",
"VirtualNetworkName": "ExamRefVNET",
```

After the variables are defined you can then add the virtual network resource to the resource's element in your template. Listing 3-2 creates a virtual network named ExamRefVNET, with an address space of 10.0.0.0/16 and two subnets: FrontEndSubnet 10.0.0.0/24 and BackEndSubnet 10.0.1.0/24. Note the syntax to read the value of variables: [variables('variablename')] is used heavily when authoring templates. The virtual network's location is set based on the return value of the built-in resourceGroup() function, which returns information about the resource group the resource is being created or updated in.

LISTING 3-2 Template structure for creating a virtual network

[Click here to view code image](#)

```
{
  "name": "[variables('VirtualNetworkName')]",
  "type": "Microsoft.Network/virtualNetworks",
  "location": "[resourceGroup().location]",
  "apiVersion": "2019-12-01",
  "dependsOn": [],
  "properties": {
    "addressSpace": {
      "addressPrefixes": [
        "[variables('ExamRefRGPrefix')]"
      ]
    },
    "subnets": [
      {
        "name": "[variables('ExamRefRGSubnet1Name')]",
        "properties": {
          "addressPrefix": "[variables('ExamRefRGSubnet1Prefix')]"
        }
      },
    ],
  }
},
```

```

    {
        "name": "[variables('ExamRefRGSubnet2Name')]",
        "properties": {
            "addressPrefix": "[variables('ExamRefRGSubnet2Prefix')]"
        }
    }
]
}
}

```

Defining a network interface

Every virtual machine has one or more network interfaces. To create one with a template, add a variable to the variables section to store the network interface resource name as the following snippet demonstrates:

```
"VMNicName": "VMNic"
```

[Listing 3-3](#) defines a network interface named `WindowsVMNic`. This resource has a dependency on the `ExamRefVNET` virtual network. This dependency will ensure that the virtual network is created prior to the network interface creation when the template is deployed and is a critical feature of orchestration of resources in the correct order. The network interface is associated to the subnet by referencing the `ExamRefRGSubnet1Ref` variable.

LISTING 3-3 Creating a network interface

[Click here to view code image](#)

```

{
    "name": "[variables('VMNicName')]",
    "type": "Microsoft.Network/networkInterfaces",
    "location": "[resourceGroup().location]",
    "apiVersion": "2019-12-01",
    "dependsOn": [
        "[resourceId('Microsoft.Network/virtualNetworks',
'ExamRefVNET')]"
    ],
    "properties": {
        "ipConfigurations": [
            {
                "name": "ipconfig1",
                "properties": {
                    "privateIPAllocationMethod": "Dynamic",
                    "subnet": {

```

```
        "id": "[variables('ExamRefRGSubnet1Ref')]"  
    }  
}  
]  
}  
}  
}
```



Exam Tip

To specify a static private IP address in template syntax, specify an address from the assigned subnet using the `privateIpAddress` property and set the `privateIpAllocation` method to `Static`.

[Click here to view code image](#)

```
"privateIpAddress": "10.0.0.10",  
"privateIpAllocationMethod": "Static",
```

Adding a Public IP Address

To add a public IP address to the virtual machine, you must make several modifications . The first is to define a parameter that the user will use to specify a unique DNS name for the public IP. The following code goes in the parameters block of a template:

[Click here to view code image](#)

```
"VMPublicIPDnsName": {  
    "type": "string",  
    "minLength": 1  
}
```

The second modification is to add the public IP resource itself. Before adding the resource, add a new variable in the variables section store the name of the public IP resource.

[Click here to view code image](#)

```
"VMPublicIPName": "VMPublicIP"
```

[Listing 3-4](#) shows a public IP address resource with the public IP allocation method set to `Dynamic` (it can also be set to `Static`). The `domainNameLabel` property of the IP address `dnsSettings` element is populated by the parameter. This makes it easy to specify a unique value for the address at deployment time.

LISTING 3-4 Creating a network interface

[Click here to view code image](#)

```
{
    "name": "[variables('VMPublicIPName')]",
    "type": "Microsoft.Network/publicIPAddresses",
    "location": "[resourceGroup().location]",
    "apiVersion": "2019-12-01",
    "dependsOn": [],
    "properties": {
        "publicIPAllocationMethod": "Dynamic",
        "dnsSettings": {
            "domainNameLabel": "[parameters('VMPublicIPDnsName')]"
        }
    }
}
```

The next modification is to update the network interface resource that the public IP address is associated with. The network interface must now have a dependency on the public IP address to ensure it is created before the network interface. The following example shows the addition to the `dependsOn` array, as the following example demonstrates:

[Click here to view code image](#)

```
"dependsOn": [
    "[resourceId('Microsoft.Network/virtualNetworks',
'ExamRefVNET')]",
    "[resourceId('Microsoft.Network/publicIPAddresses',
variables('VMPublicIPName'))]"
],
```

The `ipConfigurations -> properties` element must also be modified to reference the `publicIPAddress` resource. See [Listing 3-5](#).

LISTING 3-5 IP Configurations

[Click here to view code image](#)

```
"ipConfigurations": [
    {
        "name": "ipconfig1",
        "properties": {
            "privateIPAllocationMethod": "Dynamic",
            "subnet": {
                "id": "[variables('ExamRefRGSubnet1Name')]"
            },
            "publicIPAddress": {
                "id": "[resourceId('Microsoft.Network/publicIPAddresses',
                    variables('VMPublicIPName'))]"
            }
        }
    }
]
```

Defining a virtual machine resource

Before creating the virtual machine resource, you will add several parameters and variables to define. Each virtual machine requires administrative credentials. To enable a user to specify the credentials at deployment time, add two additional parameters for the administrator account and the password.

[Click here to view code image](#)

```
"VMAdminUserName": {
    "type": "string",
    "minLength": 1
},
"VMAdminPassword": {
    "type": "string",
    "minLength": 1
}
```

Several variables are needed to define the configuration of the virtual machine resource. The following variables define the VM name, operating system image, and the VM size. These should be inserted into the variables section of the template.

[Click here to view code image](#)

```
"VMName": "MyVM",
"VMImagePublisher": "MicrosoftWindowsServer",
```

```

"VMImageOffer": "WindowsServer",
"VMOSVersion": "WS2019-Datacenter",
"VMOSDiskName": "VM20SDisk",
"VMSize": "Standard_D2_v2",
"VM2ImagePublisher": "MicrosoftWindowsServer",
"VM2ImageOffer": "WindowsServer",
"VM20SDiskName": "VM20SDisk",
"VMSize": "Standard_D2_v2"

```

The VM has a dependency on the network interface. It doesn't have to have a dependency on the virtual network, because the network interface itself does. This VM is using managed disks, so there are no references to storage accounts for the VHD file. [Listing 3-6](#) shows a sample virtual machine resource.

LISTING 3-6 Virtual machine resource

[Click here to view code image](#)

```

{
  "name": "[parameters('VMName')]",
  "type": "Microsoft.Compute/virtualMachines",
  "location": "[resourceGroup().location]",
  "apiVersion": "2019-12-01",
  "dependsOn": [
    "[resourceId('Microsoft.Network/networkInterfaces',
variables('VMNicName'))]"
  ],
  "properties": {
    "hardwareProfile": {
      "vmSize": "[variables('vmSize')]"
    },
    "osProfile": {
      "computerName": "[variables('VMName')]",
      "adminUsername": "[parameters('VMAdminUsername')]",
      "adminPassword": "[parameters('VMAdminPassword')]"
    },
    "storageProfile": {
      "imageReference": {
        "publisher": "[variables('VMImagePublisher')]",
        "offer": "[variables('VMImageOffer')]",
        "sku": "[variables('VMOSVersion')]",
        "version": "latest"
      },
      "osDisk": {
        "createOption": "FromImage"
      }
    },
  }
}

```

```

"networkProfile": {
    "networkInterfaces": [
        {
            "id": "[resourceId('Microsoft.Network/networkInterfaces',
variables('VMNicName'))]"
        }
    ]
}
}

```

There are several properties of a virtual machine resource that are critical to its configuration.

- **hardwareProfile** This element is where you set the size of the virtual machine. Set the `vmSize` property to the desired size, such as `Standard_D2_v2`.
- **osProfile** This element at a basic level is where you set the `computerName` and `adminUsername` properties. The `adminPassword` property is required if you do not specify an SSH key. This element also supports three sub elements: `windowsConfiguration`, `linuxConfiguration` and `secrets`.
- **osProfile, windowsConfiguration** While the example doesn't use this configuration, this element provides the ability to set advanced properties on Windows VMs:
 - `provisionVMAgent` This is enabled by default, but you can disable it. Specify whether extensions can be added.
 - `enableAutomaticUpdates` Specify whether windows updates are enabled.
 - `timeZone` Specify the time zone for the virtual machine.
 - `additionalUnattendContent` Pass unattended install configuration for additional configuration options.
 - `winRM` Configure Windows PowerShell remoting.
 - `provisionVMAgent` Enabled by default, but you can disable. Specify whether extensions can be added.
 - `disablePasswordAuthentication` If set to true you must specify an SSH key.
 - `ssh, publicKeys` Specify the public key to use for authentication with

the VM.

- `osProfile`, `secrets` This element `secrets` is used for deploying certificates that are in Azure Key Vault.
- `storageProfile` This element is where OS image is specified, and the OS and data disk configuration are set.
- `networkProfile` This element is where the network interfaces for the virtual machine are specified.

More Info Resource Manager Template Schema

Reading through the Azure resource manager template schema is a great way to learn the capabilities of templates. The latest virtual machine schema is published at <https://docs.microsoft.com/en-in/azure/templates/microsoft.compute/2019-12-01/virtualmachines>.

Modify an Azure Resource Manager template

Often you will need to modify a template that you have previously used to change the configuration. As previously mentioned, one of the key concepts of using templates to describe your infrastructure (commonly referred to as Infrastructure as Code) is so you can modify it and deploy in a versioned manner. To accommodate this behavior ARM supports two different deployment modes: complete and incremental.

More Info Infrastructure as code

Infrastructure as Code (known as IaC) is a descriptive model to manage the infrastructure. More information can be found at <https://docs.microsoft.com/azure/devops/learn/what-is-infrastructure-as-code>.

In complete mode, Azure Resource Manager deletes resources that exist in the resource group that are not in the template. This is helpful if you need to remove a resource from Azure and you want to make sure your template matches the deployment. You can remove the resource from the template,

deploy using complete mode, and it will be removed.

More Info REST API version and deployment mode

The behavior discussed here really depends on the REST API version. If you use a version earlier than 2019-05-10, resources are not deleted. Also, there might be other possibilities such as resource locks or policy that prevent resources from being deleted.

In incremental mode, Azure Resource Manager leaves unchanged resources that exist in the resource group but aren't in the template. It will update the resources in the resource group if the settings in the template differ from what is deployed. Incremental mode can have unintended impacts on resource properties. If your template doesn't cover all the properties of a resource, then at the time of deployment unspecified properties will be reset to default values that can potentially affect to the environment.

Incremental is the default mode for the Azure portal and when you are deploying through the command-line tools or Visual Studio. To use Complete mode, you must use the REST API or the command-line tools with the `-Mode/- --mode` parameter set to `Complete`.

The following example deploys a template in `Complete` mode using PowerShell.

[Click here to view code image](#)

```
New-AzResourceGroupDeployment '  
    -Mode Complete '  
    -Name simpleVMDeployment '  
    -ResourceGroupName ExamRefRG '  
    -TemplateFile C:\ARMTemplates\deploy.json'
```

The example listed below deploys a template in `Complete` mode using the Azure CLI.

[Click here to view code image](#)

```
az group deployment create \  
    --name simpleVMDeployment \  
    --mode Complete \  
    --resource-group ExamRefRG \  
    --template-file deploy.json
```

Configure a virtual hard disk template

It is assumed that you already know the structure of the ARM template. For detailed structure and syntax, please refer to <https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/template-syntax>.

In the `storageProfile` section of a virtual machine resource, you can specify the `imageReference` element that references an image from the Azure Marketplace:

[Click here to view code image](#)

```
"imageReference": {  
    "publisher": "[variables('VMImagePublisher')]",  
    "offer": "[variables('VMImageOffer')]",  
    "sku": "[parameters('VMOSVersion')]",  
    "version": "latest"  
}
```

You also can specify a generalized VHD that you have previously created. To specify a user image, you must specify the `osType` property (Windows or Linux), and the URL to the VHD itself, and the URL to where the disk will be created in Azure Storage (`osDiskVhdName`). The following alternative code snippet demonstrates this. (This sample does not build on the previous example.)

[Click here to view code image](#)

```
"storageProfile": {  
    "osDisk": {  
        "name": "[concat(variables('vmName'), '-osDisk')]",  
        "osType": "[parameters('osType')]",  
        "caching": "ReadWrite",  
        "image": {  
            "uri": "[parameters('vhndl')]"  
        },  
        "vhd": {  
            "uri": "[variables('osDiskVhdName')]"  
        },  
        "createOption": "FromImage"  
    }  
}
```

For context, the following `vhndl` parameter and `osDiskVhdName` variable is shown:

[Click here to view code image](#)

```

    "vhdUrl": {
        "type": "string",
        "metadata": {
            "description": "VHD Url..."
        }
    }
}
"osDiskVhdName": [
    concat('http://', parameters('userStorageAccountName'),
    '.blob.core.windows.net/', parameters('userStorageContainerName'), '/',
    parameters('vmName'), 'osDisk.vhd')]
]

```

See the following for a complete template example:
<https://docs.microsoft.com/mt-ml/azure/marketplace/partner-center-portal/azure-vm-image-certification>

Deploy from a template

You can deploy templates using the Azure portal, the command line tools, or directly using the REST API. Let's start with deploying a template that creates a virtual machine using the Azure portal. To deploy a template from the Azure portal, click the **Create Resource** button and search for **template deployment**, select the template deployment name from the search results, and then click **Create**, as shown in [Figure 3-1](#).

From there, you have the option to build your own template using the Azure portal's editor (you can paste your own template in or upload from a file using this option, too) or choose from one of the most common templates. Last of all, you can search the existing samples in the Quickstart samples repository in GitHub and choose one of them as a starting point. [Figure 3-2](#) shows the various options after clicking the template deployment search result.

Template deployment (deploy using custom templates)

Microsoft



Template deployment (deploy using custom templates)

[Save for later](#)

Microsoft

[Create](#)

[Overview](#)

[Plans](#)

Applications running in Microsoft Azure usually rely on a combination of resources, like databases, servers, and web apps. Azure Resource Manager templates enable you to deploy and manage these resources as a group, using a JSON description of the resources and their deployment settings.

Edit your template with IntelliSense and deploy it to a new or existing resource group.

[Useful Links](#)

[Documentation](#)

FIGURE 3-1 The Template Deployment option

Custom deployment

Deploy from a custom template

Learn about template deployment

 [Read the docs](#) ↗

 [Build your own template in the editor](#)

Common templates

 [Create a Linux virtual machine](#)

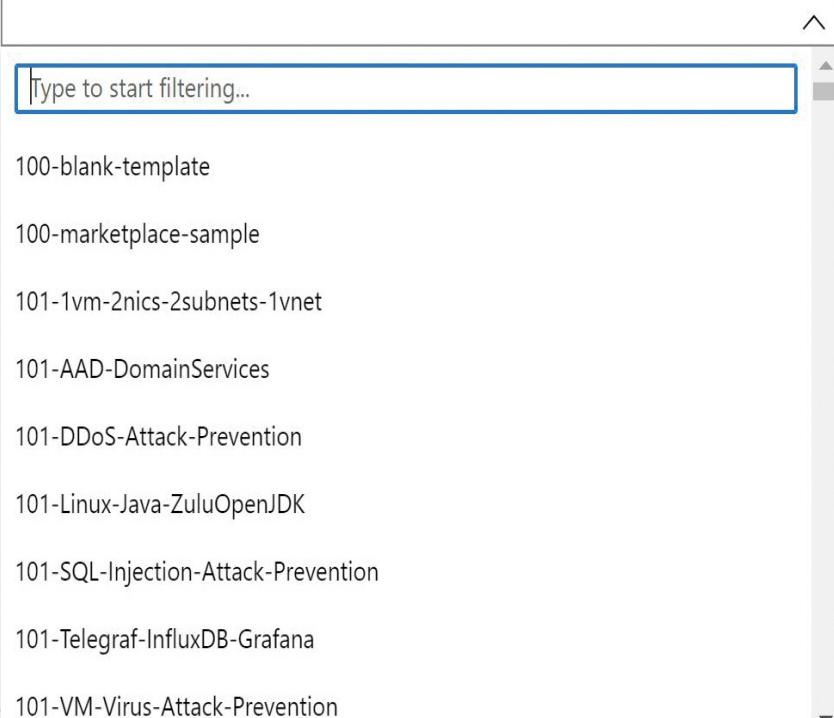
 [Create a Windows virtual machine](#)

 [Create a web app](#)

 [Create a SQL database](#)

Load a GitHub quickstart template

Select a template (disclaimer) ⓘ



Type to start filtering...

- 100-blank-template
- 100-marketplace-sample
- 101-1vm-2nics-2subnets-1vnet
- 101-AAD-DomainServices
- 101-DDoS-Attack-Prevention
- 101-Linux-Java-ZuluOpenJDK
- 101-SQL-Injection-Attack-Prevention
- 101-Telegraf-InfluxDB-Grafana
- 101-VM-Virus-Attack-Prevention

FIGURE 3-2 Options for configuring a template deployment

Clicking the **Build Your Own Template In The Editor** option allows you to paste in template code directly. This allows you to author and then deploy templates using the Azure portal for simple testing. In [Figure 3-3](#), you can see the Edit Template window.

Edit template

Edit your Azure Resource Manager template

[Add resource](#) [Quickstart template](#) [Load file](#) [Download](#)

Parameters (6)

- vmList (string)
- location (string)
- domainJoinUserName (string)
- domainJoinUserPassword (securestring)
- domainFQDN (string)
- ouPath (string)

Variables (2)

- domainJoinOptions
- vmListArray

Resources (1)

- [concat(trim(variables('vmListArray')[copyIndex()]), '/joindomain')]
(Microsoft.Compute/virtualMachin

```
1  {
2      "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3      "contentVersion": "1.0.0.0",
4      "parameters": {
5          "vmList": {
6              "type": "string",
7              "minLength": 1,
8              "metadata": {
9                  "description": "List of virtual machines to be domain joined"
10             }
11         },
12         "location": {
13             "type": "string",
14             "metadata": {
15                 "description": "Location name of the virtual machine"
16             },
17             "defaultValue": "[resourceGroup().location]"
18         },
19         "domainJoinUserName": {
20             "type": "string",
21             "metadata": {
22                 "description": "User name of a domain user with rights to perform domain join operation"
23             }
24         },
25         "domainJoinUserPassword": {
26             "type": "securestring",
27             "metadata": {
28                 "description": "Domain user password"
29             }
30         },
31         "domainFQDN": {
32             "type": "string",
33             "metadata": {
34                 "description": "Domain FQDN where the virtual machine will be joined"
35             }
36         },
37     }
```

Save

Discard

FIGURE 3-3 Editing a template using the Azure portal editor

More Info ARM template Validation

While creating the ARM template using the Azure portal editor, the template validation is performed by default. Parameters, variables, and resources will not populate if there are any template errors, and red indicators in the right margin will call out any errors.

Clicking **Save** on the previous screen takes you to the page shown in [Figure 3-4](#) where you can specify the resource group and any parameters needed to deploy the template.

More Info ARM template UI

While creating the ARM template using the Azure portal editor, the settings shown in [Figure 3-4](#), including parameter sets and values, come from the ARM template itself. Based on what you define in your template, the screen will be updated accordingly.

Custom deployment

Deploy from a custom template

TEMPLATE

 Customized template
1 resource

 Edit template  Edit paramet...  Learn more

BASICS

Subscription *

Resource group * [Create new](#)

Location

SETTINGS

Vm List *

Location

Domain Join User Name *

Domain Join User Password *

Domain FQDN *

Ou Path

TERMS AND CONDITIONS

[Azure Marketplace Terms](#) | [Azure Marketplace](#)

By clicking "Purchase," I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

I agree to the terms and conditions stated above

Purchase

FIGURE 3-4 The template editor view

The **Edit Parameters** button allows you to edit a JSON view of the parameters for the template, as shown in [Figure 3-5](#). This file can also be downloaded and is used to provide different behaviors for the template at deployment time without modifying the entire template.

Common examples of using a parameters file:

- Defining different instance sizes or SKUs for resources based on the intended usage (small instances for test environments for example)
- Defining different number of instances
- Different regions
- Different credentials

It is recommended that you use the `securestring` type for the parameters while passing the confidential data, such as passwords and secrets.

Edit parameters

 Load file  Download

```
1  {
2      "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
3      "contentVersion": "1.0.0.0",
4      "parameters": {
5          "vmList": {
6              "value": "VM1, VM2, VM3"
7          },
8          "location": {
9              "value": "[resourceGroup().location]"
10         },
11         "domainJoinUserName": {
12             "value": "domainuser"
13         },
14         "domainJoinUserPassword": {
15             "value": "domain@pass1234"
16         },
17         "domainFQDN": {
18             "value": "examref.test"
19         },
20         "ouPath": {
21             "value": "OU=MyComputers,DC=examref,DC=test"
22         }
23     }
24 }
```

FIGURE 3-5 Editing template parameters using the Azure portal

More Info ARM template best practices

The recommended practices while working with ARM templates can be found at <https://docs.microsoft.com/azure/azure-resource-manager/templates/template-best-practices>.

The last step to creating a template using the Azure portal is to click the **Purchase** button after reviewing and agreeing to the terms and conditions on the screen. Clicking the **Purchase** button will trigger the deployment.

The Azure command line tools can also deploy resources using templates. The template files can be located locally on your file system or accessed via HTTP/HTTPs. Common deployment models have the templates deployed into a source code repository or an Azure storage account to make it easy for others to deploy the template.



Exam Tip

The parameters to a template can be passed to the `New-AzResourceGroupDeployment` cmdlet using the `TemplateParameterObject` parameter for values that are defined directly in the script as .json. The `TemplateParameterFile` parameter can be used for values stored in a local .json file. The `TemplateParameterUri` parameter for values that are stored in a .json file at an HTTP endpoint.



Exam Tip

The parameters to a template can be passed to the `az group deployment create` command using the `parameters` section for values that are defined directly in the script as `.json`. The `template-file` parameter can be used for values stored in a local `.json` file. The `template-uri` parameter can be used for values that are stored in a `.json` file at an HTTP endpoint.

Save a deployment as an Azure Resource Manager template

An existing deployment can be exported as a template that you can use to regenerate the environment or to just gain a better understanding of how the deployment is configured. There are two ways of exporting a template from a deployment within a resource group.

The first way is to export the actual template used for the deployment. This method exports the template exactly as it was used, including the values for parameters and variables during the original execution. This approach does not capture any changes made to the deployment after it was deployed. To export this template, navigate to the resource group in the Azure portal and click **Deployments**, select the deployment to export, and click **View Template** on the top navigation. [Figure 3-6](#) depicts a deployment selected inside a **Resource Group**.

ExamRefRg | Deployments

Resource group

Search (Ctrl+ /)



Refresh



Cancel



Redeploy



Delete



View template

Overview

Activity log

Access control (IAM)

Tags

Events

Settings

Quickstart

Deployments

Policies

Export template

Properties

Locks

Filter by deployment name or resources in the deployment...

	Deployment name	Status	Last modified
	Microsoft.VirtualNetwork-20200914233329	Succeeded	9/14/2020, 11:33:51 PM
	Microsoft.StorageSync-20200909234114	Succeeded	9/9/2020, 11:41:21 PM
<input checked="" type="checkbox"/>	CreateVm-MicrosoftWindowsServer.WindowsServer...	Succeeded	9/9/2020, 11:19:03 PM
	Microsoft.StorageAccount-20200907234922	Succeeded	9/7/2020, 11:50:52 PM

FIGURE 3-6 The deployments view of an Azure Resource Group in the Azure portal

Clicking the **View Template** link opens in the **Template** deployment view, as shown in [Figure 3-7](#). From here, you can click **Download** to download the template locally, you can click **Deploy** to re-deploy the template using different parameters, or you can click **Add To Library** to save to your template gallery for later deployment.

Home > Resource groups > examref | Deployments > Template

Template

Microsoft.Template

[Download](#) [Add to library \(preview\)](#) [Deploy](#)

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Define resources and configurable input parameters and deploy with script or code. [Learn more about template deployment.](#)

[Include parameters](#) [?](#)

Template Parameters Scripts

Parameters (5)

- virtualMachineSize (string)
- adminUsername (string)
- adminPassword (securestring)
- storageAccountType (string)
- location (string)

Variables (12)

- virtualMachineName
- nic1
- nic2
- virtualNetworkName
- subnet1Name
- subnet2Name
- publicIPAddressName
- subnet1Ref
- subnet2Ref
- diagStorageAccountName
- networkSecurityGroupName
- networkSecurityGroupName2

Resources (8)

- [variables('virtualMachineName') (Microsoft.Compute/virtualMachine)]
- [variables('diagStorageAccountName') (Microsoft.Storage/storageAccount)]

```

1 {
2   "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "virtualMachineSize": {
6       "defaultValue": "Standard_DS1_v2",
7       "type": "String",
8       "metadata": {
9         "description": "Virtual machine size (has to be at least the size of Standard_A3 to support 2 NICs)"
10      }
11    },
12    "adminUsername": {
13      "type": "String",
14      "metadata": {
15        "description": "Default Admin username"
16      }
17    },
18    "adminPassword": {
19      "type": "SecureString",
20      "metadata": {
21        "description": "Default Admin password"
22      }
23    },
24    "storageAccountType": {
25      "defaultValue": "Standard_LRS",
26      "allowedValues": [
27        "Standard_LRS",
28        "Premium_LRS"
29      ],
30      "type": "String",
31      "metadata": {
32        "description": "Storage account type (Standard_LRS or Premium_LRS). Standard_LRS is recommended for most workloads. Premium_LRS is recommended for workloads that require low latency access to data stored in the account. Premium_LRS is also required for certain features like geo-redundant storage and snapshot support. Note that Premium_LRS accounts incur higher costs than Standard_LRS accounts. For more information, see Storage account types."}
33    }
34  }
35 }

```

FIGURE 3-7 The deployments view of an Azure resource group in the Azure portal

The second approach to generating an ARM template is to use the Automation Script menu option for the resource group. It generates a template that represents the current state of the resource group. The state might have been updated by multiple templates, or it might have updated by changes from the Azure portal or changes via the REST API or command line. It might include many hard-coded values and probably not as many parameters as you would expect in a template that was designed for reusability. This template is useful for redeploying to the same resource group because of the hard-coded values. Using it for other resource groups may require a significant amount of editing. You can access this template by navigating to the resource group and clicking the **Automation Script** link on the left pane.

Deploy virtual machine extensions

Azure virtual machines have a variety of built-in extensions that can enable configuration management as well as a variety of other operations such as installing software agents and even enabling remote debugging for live troubleshooting purposes. The two most common extensions for configuration management are the Windows PowerShell Desired State Configuration (DSC) extension and the more generic Custom Script Extension. Both extensions can be executed at provisioning time or after the virtual machine has already been started. The Windows PowerShell DSC Extension allows you to define the state of a virtual machine using the PowerShell Desired State Configuration language and apply it as well as perform continuous updates when integrated with the Azure Automation DSC service. The custom script extension can be used to execute an arbitrary command such as a batch file, regular PowerShell script, or a bash script. In addition to these extensions there are also more specific extensions that allow you to configure your virtual machines to use open-source configuration management utilities such as Chef or Puppet and many others.

While creating VMs, you can add an extension using Advanced blade, which is shown in [Figure 3-8](#). This blade allows the user creating the virtual machine to install an extension, such as the custom script extension or one of many others.

Create a virtual machine

Basics Disks Networking Management **Advanced** Tags Review + create

Add additional configuration, agents, scripts or applications via virtual machine extensions or cloud-init.

Extensions

Extensions provide post-deployment configuration and automation.

Extensions ⓘ

Select an extension to install

Cloud init

Cloud init is a widely used approach to customize a Linux VM as it boots for the first time. You can use cloud-init to install packages and write files or to configure users and security. [Learn more](#)



The selected image does not support cloud init.

FIGURE 3-8 The Advanced blade in the Azure portal

[Figure 3-9](#) depicts the custom script extensions that appear when you click the **Select An Extension To Install** option, as shown previously in [Figure 3-8](#).

Using the custom script extension

The Azure custom script extension is supported on Windows and Linux-based virtual machines and is ideal for bootstrapping a virtual machine to an initial configuration. To use the Azure custom script extension, your script must be accessible via a URI, such as an Azure storage account, and must either accessed anonymously or passed with a shared access signature (SAS URL). The custom script extension takes as parameters the URI and the command to execute including any parameters to pass to the script. You can execute the script at any time the virtual machine is running.

New resource



	Agent for Cloud Workload Protection (Windows) Symantec Corp.
	Agent for Windows Server Monitoring Site24x7
	APM Insight .NET Agent Site24x7
	SentinelOne Windows Extension SentinelOne
	Rapid7 Insight Agent Rapid7 Inc.
	Puppet Agent Puppet
	Octopus Deploy Tentacle Agent Octopus Deploy Pty. Ltd.
	Azure Pipelines Agent Microsoft Corp.
	NVIDIA GPU Driver Extension Microsoft Corp.
	Network Watcher Agent for Windows Microsoft Corp.
	PowerShell Desired State Configuration Microsoft Corp.
	Custom Script Extension

FIGURE 3-9 The select a custom script extension menu

To add the Custom Script Extension to an existing virtual machine, open the virtual machine in the Azure portal, click the **Extensions** link on the left, and choose the **Custom Script Extension** option. The script file is specified as well as any arguments passed to the script. Figure 3-10 shows how to enable this extension using the Azure portal.



Exam Tip

There are many other ways of configuring and executing the Custom Script Extension using the Azure CLI. The following article has several relevant examples that might be used in an exam, which you can find at <https://docs.microsoft.com/en-in/azure/virtual-machines/extensions/custom-script-linux>.

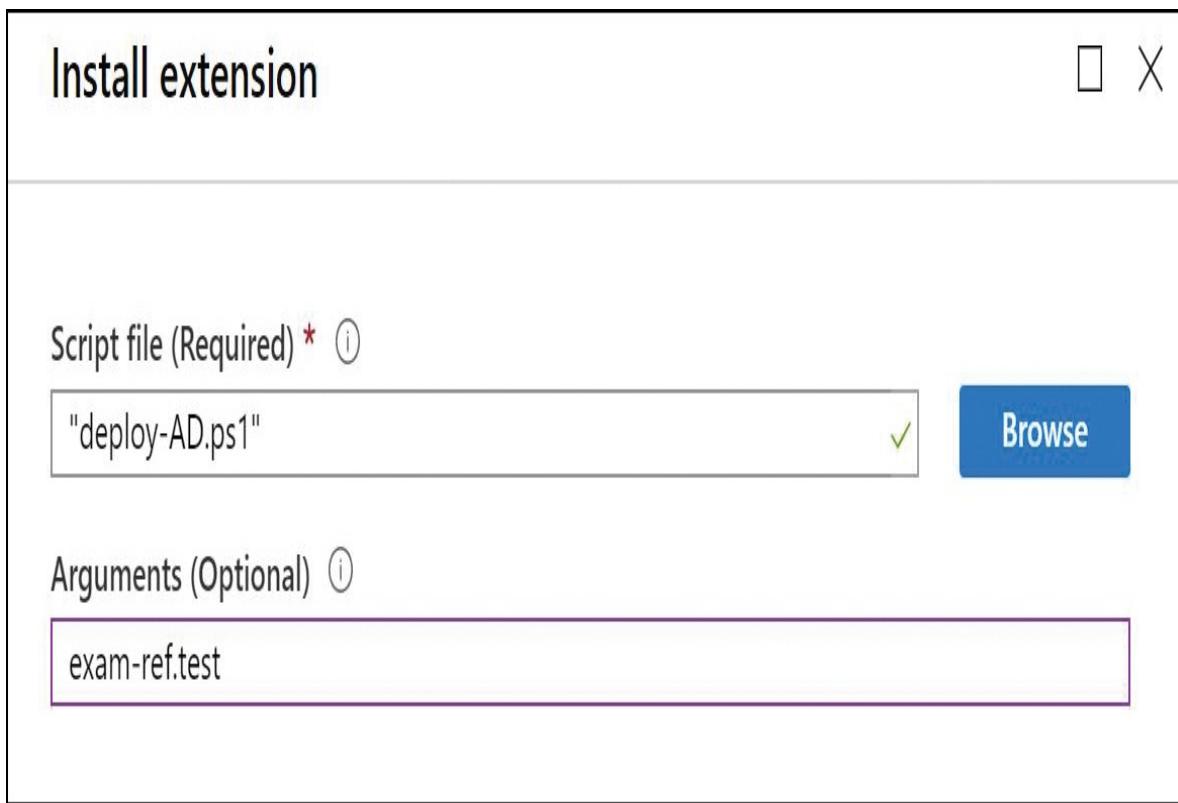


FIGURE 3-10 Specifying the Custom Script Extension configuration

More Info Troubleshooting using virtual machine extension logs

In the event your Custom Script Extension fails to execute, it's a good idea to review the log files. In Windows, the logs are located at `C:\WindowsAzure\Logs\Plugins\Microsoft.Compute.CustomScriptExtension`. In Linux, the command output is located at `/var/lib/waagent/Microsoft.OSTCExtensions.CustomScriptForLinux-<version>/download/1`.

Skill 3.2: Configure VMs for high availability and scalability

Microsoft Azure virtual machines are a flexible and powerful option for deploying workloads into the cloud. The support of both Windows and Linux-

based operating systems allows for the deployment of a wide variety of workloads that traditionally run in an on-premises environment. This section will cover how to configure high availability for Azure VMs. It also covers how to deploy and configure virtual machine scale sets using various tools.

This section covers how to:

- [Configure high availability](#)
- [Deploy and configure scale sets](#)

Configure high availability

Resiliency is a critical part of any application architecture. Azure provides several features and capabilities to make it easier to design resilient solutions. The platform helps you to avoid a single point of failure at the physical hardware level and provides techniques to avoid downtime during host updates. Using features such as availability zones, availability sets, and load balancers provides you the capabilities to build highly resilient and available systems.

Availability zones

Availability Zones are separate units—each with its own power, cooling, and networking—which provide higher resiliency and protect applications and data from disruption in the data centers. To ensure resiliency, there is a minimum of three separate zones in all enabled regions. The physical and logical separation of availability zones within a region protects applications and data from zone-level failures. Availability zones provide a 99.99 percent SLA uptime when two or more VMs are deployed into two or more availability zones. [Figure 3-11](#) demonstrates how a three-tier application can be deployed with a virtual machine from each tier deployed in each of the three zones for increased availability.

When you create VMs in three Availability Zones, those will be automatically distributed across three fault domains and three update domains. A fault domain represents a group of servers, which have shared power, cooling, and networking. An update domain represents a group of servers that can be rebooted at the same time.

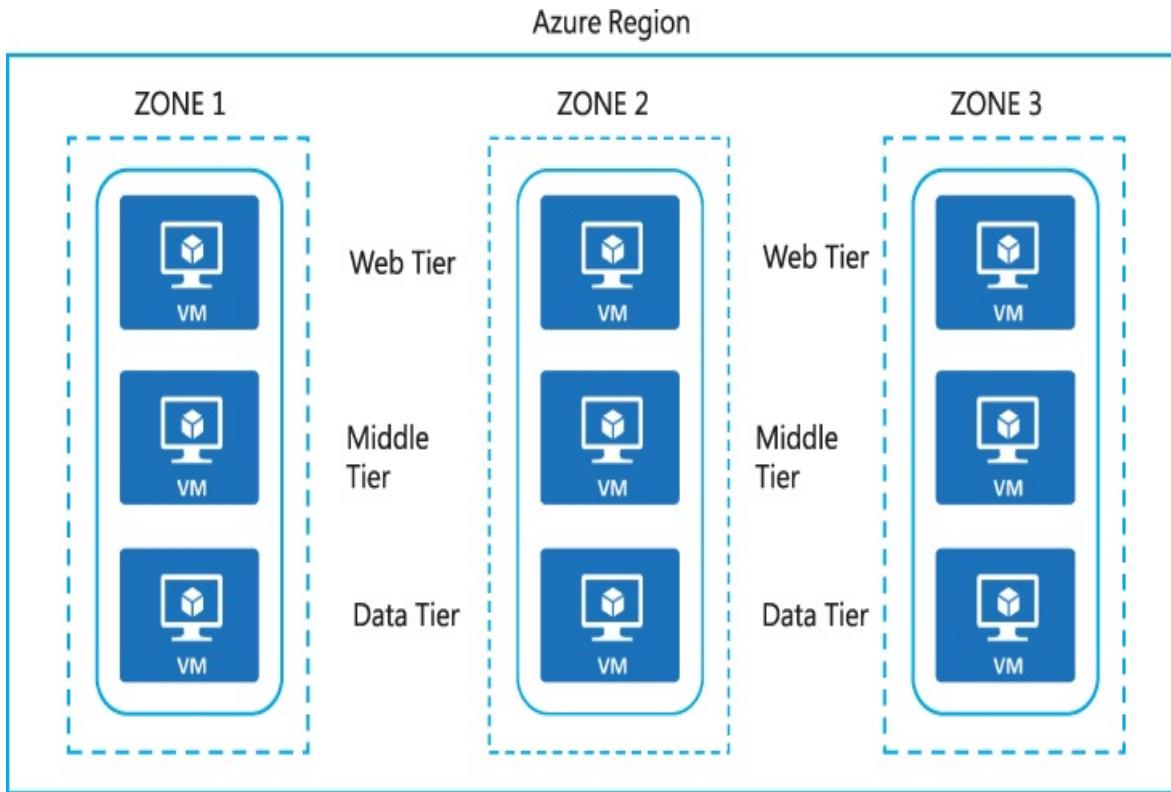


FIGURE 3-11 Architectural view of an availability zone

To deploy a VM to an availability zone, select the zone you want to use on the Basics blade of the virtual machine creation dialog, as shown in [Figure 3-12](#).

More Info Availability zones and Azure region

If you are unable to set an availability zone, it is most likely because you have selected a region where availability zones are not available.

Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource group * ⓘ

 ▼

[Create new](#)

Instance details

Virtual machine name * ⓘ

 ✓

Region * ⓘ

 ▼

Availability options ⓘ

 ▼

Availability zone * ⓘ

 ^

1

2

3

Azure Spot instance ⓘ

Size * ⓘ

Standard DS1 v2

FIGURE 3-12 Specifying the availability zone for a VM

At the time of this writing the following services are supported with availability zones (see: <https://docs.microsoft.com/en-us/azure/availability-zones/az-region#services-support-by-region> for the complete list):

- Linux Virtual Machines
- Windows Virtual Machines
- Virtual Machine Scale Sets
- Azure App Service Environments
- Azure Kubernetes Service
- Managed Disks
- Azure Firewall Load Balancer
- Public IP address
- Zone-redundant storage
- SQL Database
- Event Hubs
- Service Bus
- VPN Gateway
- ExpressRoute
- Application Gateway

Currently supported regions:

- Central US
- East US
- East US 2
- West US 2
- France Central
- North Europe
- UK South

- West Europe
- Japan East

Southeast Asia Azure services supported by availability zones are divided into two categories: zonal services and zone-redundant services. Zonal services are the services where the availability zone is assigned explicitly, such as when a virtual machine needs to be placed in the availability zone at the time of creation. Zone-redundant services are replicated by the Azure portal itself. For example, zone-redundant storage account which gets replicated automatically.

Availability sets

Deploying a multitier application into an availability set can provide redundancy and high availability to the virtual machines. To provide redundancy for your virtual machines, you must place at least two virtual machines in an availability set. This configuration ensures that at least one virtual machine is available in the event of a host update, or a problem with the physical hardware the virtual machines are hosted on. Having at least two virtual machines in an availability set is a requirement for the service level agreement (SLA) for virtual machines of 99.95 percent.

You can place a single instance virtual machine in an availability set, too, but doing so provides comparatively lower SLAs. A Premium SSD provides an SLA of 99.9 percent, while Standard SSD and Standard HDD provide SLAs of 99.5 percent and 95 percent, respectively.

Virtual machines should be deployed into availability sets according to their workload or application tier. For instance, if you are deploying a three-tier solution that consists of web servers, a middle tier, and a database tier, each tier would have its own availability set, as [Figure 3-13](#) demonstrates.

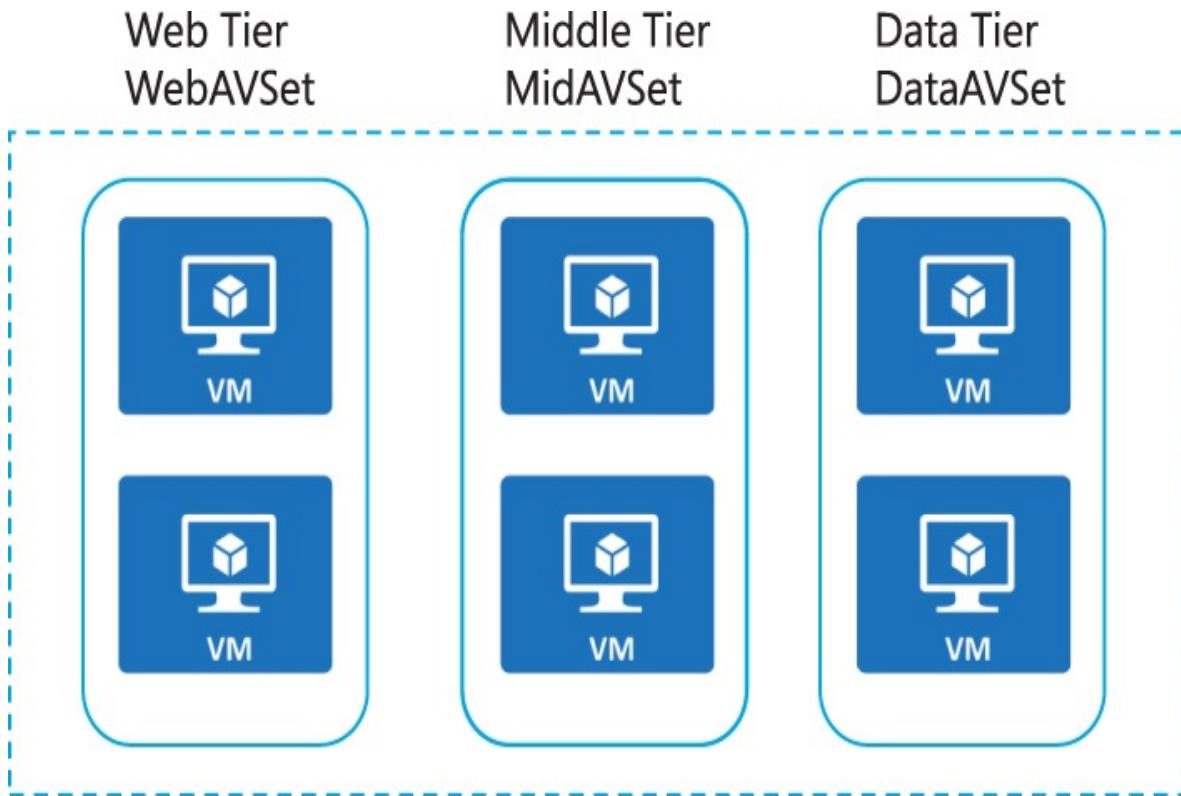


FIGURE 3-13 Availability set configurations for a multi-tier solution

Availability sets can be configured by assigning a fault domain and an update domain. Fault domain represents a group of servers that have shared power, cooling, and networking, while an update domain represents a group of servers that can be rebooted at the same time. Each availability set can have up to 20 update domains and 3 fault domains. This reduces the impact to VMs from physical hardware failures, such as server, network, or power interruptions on one of the physical racks. It is important to understand that the availability set must be set at creation time of the virtual machine.

Create an availability set

To create an availability set, specify a name for the availability set that is not in use by any other availability sets within the resource group, along with the number of fault and updates domains, as well as whether you will use managed disks with the availability set or not. In order to create the availability set at the time of virtual machine creation, please go to homepage, then click **+ Create A Resource**, and then search for **virtual machine** and click **Create**. You will be presented with the **Basics** blade, as shown in [Figure 3-14](#). On the **Create New**

pane, you can create a new availability set. You can select the number of **Fault Domains** and **Update Domains**.

You can also create availability set by clicking + **Create A Resource**, searching for **availability set**, and clicking **Create**. The Basics blade appears, where you can select subscription, resource group, and region, and you can specify the availability set name, fault domain, and update domain. On the **Advanced** blade, you have an option to select proximity placement group if it's already created. Click the **Review + Create** button at the bottom to create availability set. Now you can place resources such as VMs in this newly created availability set by selecting it at the time of resource creation.

Home > Subscriptions > Resource groups > examrefRG > New > Create a virtual machine

Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ① [Create new](#)

Resource group * ① [Create new](#)

Instance details

Virtual machine name * ① ExamRefHA-VM

Region * ① (US) East US

Availability options ① Availability set

Availability set * ① No existing availability sets in current resource group and location. [Create new](#)

Image * ① Windows Server 2016 Datacenter [Browse all public and private images](#)

Azure Spot instance ① Yes No

[Review + create](#) [< Previous](#) [Next : Disks >](#)

Create new X

Group two or more VMs in an availability set to ensure that at least one is available during planned or unplanned maintenance events. [Learn more](#)

Name * WebAvSet ✓

Fault domains ①  2

Update domains ①  5

Use managed disks ① No (Classic) Yes (Aligned)

OK

FIGURE 3-14 Creating an availability set

More Info Proximity Placement Group

A proximity placement group is a logical grouping of VMs to reduce the latency by keeping them closer to each other. If the VMs are placed in the same proximity placement group, they will be physically located closer to each other.

Availability sets and managed disks

Availability sets and managed disks complement each other. When the VM uses managed disks and is placed in an availability set (known as an aligned availability set), it ensures that the VM disks are placed in different storage fault domains as shown in [Figure 3-15](#). This alignment ensures that all the managed disks attached to a VM are within the same managed disk fault domain. The number of fault domains for an availability set depends on the region it belongs to, with either two or three fault domains per region.

More Info Understanding Availability in Azure VMs

You can learn more about update and fault domains and how to manage availability of your Azure VMs at <https://docs.microsoft.com/azure/virtual-machines/windows/manage-availability>.

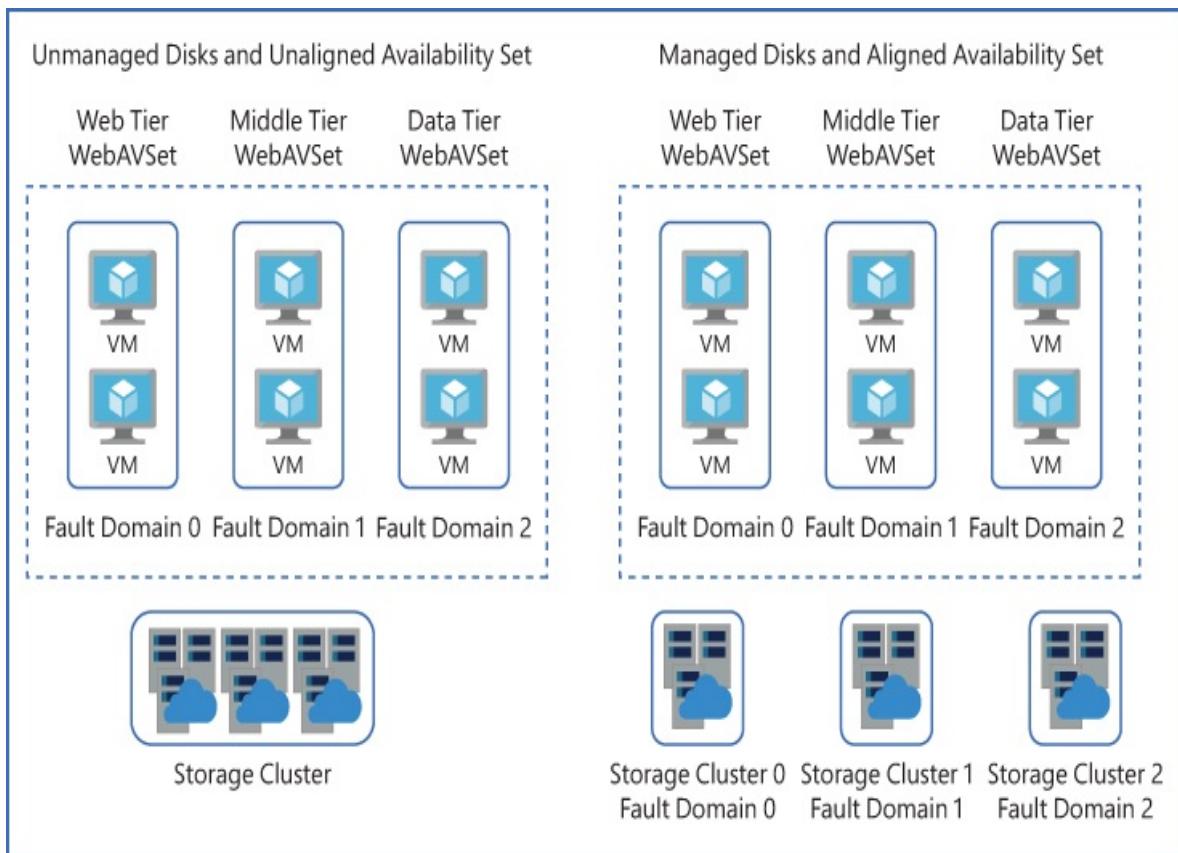


FIGURE 3-15 Aligning managed disks with an availability set

Deploy and configure scale sets

A VMSS is a compute resource that you can use to deploy and manage a set of identical virtual machines.

By default, a VMSS supports up to 100 instances. However, it is possible to create a scale set up to 1,000 instances by placing instances into multiple placement groups. A placement group is a construct, such as an Azure availability set, with its own fault domains and upgrade domains. If you define an instance count that is higher than 100 in the Azure portal when the scale set is created, the Azure portal will automatically enable the scale set for multiple placement groups. By default, a scale set consists of a single placement group with a maximum size of 100 VMs. If the scale set property called `singlePlacementGroup` is set to `false` or if you define an instance count higher than 100 in the Azure portal, the scale set can be composed of multiple placement groups and has a range of 0-1,000 VMs.

Using multiple placement groups is commonly referred to as a “large scale set.” The `singlePlacementGroup` property can be set using ARM templates or the command-line tools. Working with large scale sets does have a few conditions to be aware of:

- If you are using a custom image (not a default available image from marketplace), your scale set supports up to 600 instances instead of 1,000.
- The basic SKU of the Azure Load Balancer can scale up to 300 instances.
- For a large-scale set (> 100 instances), you should use the Standard SKU (supports up to 1,000 instances) or the Azure Application Gateway.

Creating a virtual machine scale set (Azure portal)

[Figure 3-16](#) shows a portion of the creation dialog for creating a new VM scale set using the Azure portal. Like other Azure resources, you must specify a name and the resource group to deploy to. All instances of the VMSS will use the same operating system disk image specified here.

Create a virtual machine scale set

Basics Disks Networking Scaling Management Health Advanced Tags Review + create

Azure virtual machine scale sets let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update a large number of VMs.

[Learn more about virtual machine scale sets](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Resource group *

[Create new](#)

Scale set details

Virtual machine scale set name *

Region *

Availability zone ⓘ

Instance details

Image * ⓘ

[Browse all public and private images](#)

Azure Spot instance ⓘ

Yes No

Size * ⓘ

Standard D2s v3

FIGURE 3-16 Creating a VM scale set



Exam Tip

A scale set can be deployed to an availability zone to provide higher redundancy and resiliency. If the scale set is created with a single availability zone, then all the instances will be deployed within a single zone. If the scale set is deployed in multiple availability zones (known as a zone-redundant scale set), based on scaling rules, the instances can be deployed to multiple zones if needed.

[Figure 3-17](#) shows further down the blade. This allows you to configure networking options, such as the virtual network and subnet to use, as well as which type of load balancer to use.

Create a virtual machine scale set

Azure Virtual Network (VNet) enables many types of Azure resources to securely communicate with each other, the internet, and on-premises networks. [Learn more about VNets](#)

Virtual network * ⓘ

(New) examrefRG-vnet (recommended)



[Create virtual network](#)

Network interface

A network interface enables an Azure virtual machine to communicate with internet, Azure, and on-premises resources. A VM can have one or more network interfaces.

[Create new nic](#) [Delete](#)

<input type="checkbox"/>	NAME	CREATE PUBLI...	SUBNET	NETWORK SECURI...	ACCELERATED N...
<input type="checkbox"/>	examrefRG-vnet-nic01	No	default (10.0.1.0/24)	Basic	Off

Load balancing

You can place this virtual machine scale set in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Use a load balancer

Yes No

Load balancing settings

- **Application Gateway** is an HTTP/HTTPS web traffic load balancer with URL-based routing, SSL termination, session persistence, and web application firewall. [Learn more about Application Gateway](#)
- **Azure Load Balancer** supports all TCP/UDP network traffic, port-forwarding, and outbound flows. [Learn more about Azure Load Balancer](#)

Load balancing options * ⓘ

Azure load balancer



Select a load balancer * ⓘ

(new) ExamRefVMSST-lb



[Create new](#)

Select a backend pool * ⓘ

(new) bepool



[Create new](#)

[Review + create](#)

[< Previous](#)

[Next : Scaling >](#)

FIGURE 3-17 Configuring the network and the load balancer for a VM scale set

On next screen, you can configure the **Scaling Policy** as **Manual** or **Custom**. When you set the **Scaling Policy** to **Custom**, you see the configuration options for setting the default rules, as shown in [Figure 3-18](#). Here, you can specify the minimum and maximum number of VMs in the set, and you can set the actions to scale out (add more) or to scale in (remove instances).

During the lifecycle of running a virtual machine scale set you may need to upgrade the instances with the latest scale set model. The VMSS resource property `upgrade policy` determines how VMs will be upgraded once a new update is available. Three options are available: **Automatic**, **Rolling**, and **Manual** (see [Figure 3-19](#)). If you set this to **Automatic**, all instances are updated in the random order when an update is available, which can cause downtime. If you set this to **Rolling**, the scale set updates VMs in multiple batches, and you can set a pause time between two batches, which can avoid total downtime. If the property is set to **Manual**, it is up to you to programmatically step through and update each instance using PowerShell with the `Update-AzVmssInstance` cmdlet or the Azure CLI `az vmss update-instances` command.

Create a virtual machine scale set

Basics Disks Networking **Scaling** Management Health Advanced Tags Review + create

An Azure virtual machine scale set can automatically increase or decrease the number of VM instances that run your application. This automated and elastic behavior reduces the management overhead to monitor and optimize the performance of your application. [Learn more about VMSS scaling](#)

Instance

Initial instance count * ⓘ

100



Scaling

Scaling policy ⓘ

Manual Custom

Minimum number of VMs * ⓘ

1

Maximum number of VMs * ⓘ

10

Scale out

CPU threshold (%) * ⓘ

75

Duration in minutes * ⓘ

10

Number of VMs to increase by * ⓘ

1

Scale in

CPU threshold (%) * ⓘ

25

Number of VMs to decrease by * ⓘ

1

Review + create

< Previous

Next : Management >

FIGURE 3-18 Configuring scaling rules for a virtual machine scale set

More Info Upgrading a virtual machine scale set

You can learn more about upgrading virtual machine scale sets at
<https://docs.microsoft.com/azure/virtual-machine-scale-sets/virtual-machine-scale-sets-upgrade-scale-set>.

Create a virtual machine scale set

Basics Disks Networking Scaling **Management** Health Advanced Tags Review + create

Configure monitoring and management options for your virtual machine scale set instances.

Azure Security Center

Enable basic plan for free [\(i\)](#) On Off

This will apply to every VMSS in the selected subscription

Azure Security Center provides unified security management and advanced threat protection across hybrid cloud workloads.

[Learn more](#)

Upgrade policy

Upgrade mode [*](#) [\(i\)](#)

Manual - Existing instances must be manually upgraded

^

Automatic – Instances will start upgrading immediately in random order

Manual - Existing instances must be manually upgraded

Monitoring

Boot diagnostics [\(i\)](#)

Rolling - Upgrades roll out in batches with optional pause

Diagnostics storage account [*](#) [\(i\)](#)

[Create new](#)

▼

Identity

System assigned managed identity [\(i\)](#) On Off

Automatic OS upgrades

Automatic OS upgrades [\(i\)](#) On Off

Instance termination

Instance termination notification [\(i\)](#) On Off

[Review + create](#)

[< Previous](#)

[Next : Health >](#)

FIGURE 3-19 Configuring Management rules for a virtual machine scale set

You can also add a layer of health monitoring to your application when you create VMSS. Health monitoring is required when you plan to use managed infrastructure and automatic OS upgrades. On the **Health** tab, you can enable application health monitoring and configure options by choosing the extension, protocol, port, and application endpoint path (see [Figure 3-20](#)).

Create a virtual machine scale set

Basics Disks Networking Scaling Management **Health** Advanced Tags Review + create

You can configure health monitoring on an application endpoint to update the status of the application on that instance. This instance status is required to enable platform managed upgrades like automatic OS updates and virtual machine instance upgrades. [Learn more about application health monitoring](#)

Health

Monitor application health i Disabled Enabled

Application health monitor i Application health extension ▼

Protocol i HTTP ▼

Port number i 80

Path i /

i The Application Health extension will probe the application health endpoint and update the status of the application. When the health endpoint is not set up correctly the status of the application will be reported as unhealthy. [Learn more ↗](#)

Automatic repair policy

Before enabling the automatic repairs policy, review the requirements for opting in [here](#)

Automatic repairs i On Off

Grace period (min) i 30

[Review + create](#)

[< Previous](#)

[Next : Advanced >](#)

FIGURE 3-20 Configuring Health monitoring for a virtual machine scale set

Some advanced options, such as Allocation Policy, include a spreading algorithm. Also, you can select among options such as Proximity Placement Group and VM Generation (see [Figure 3-21](#)).

Create a virtual machine scale set

Basics Disks Networking Scaling Management Health **Advanced** Tags Review + create

Add additional configuration, agents, scripts or applications via virtual machine extensions or cloud-init.

Allocation policy

Enable scaling beyond 100 instances No Yes

Force strictly even balance across zones No Yes

Spreading algorithm Max spreading Fixed spreading (not recommended with zones)

Cloud init

Cloud init is a widely used approach to customize a Linux VM as it boots for the first time. You can use cloud-init to install packages and write files or to configure users and security. [Learn more](#)

 The selected image does not support cloud init.

Proximity placement group

Proximity placement groups allow you to group Azure resources physically closer together in the same region. [Learn more](#)

Proximity placement group

No proximity placement groups found



VM generation

Generation 2 VMs support features such as UEFI-based boot architecture, increased memory and OS disk size limits, Intel® Software Guard Extensions (SGX), and virtual persistent memory (vPMEM).

VM generation

Gen 1 Gen 2

 Generation 2 VMs do not yet support some Azure platform features, including Azure Disk Encryption.

Review + create

< Previous

Next : Tags >

FIGURE 3-21 Configuring Advanced rules for a virtual machine scale set



Exam Tip

The spreading algorithm decides how scale set instances will be placed in a fault domain. With max spreading, the instances are distributed in the maximum fault domains possible for each zone. Fixed spreading restricts instances to exactly five fault domains. If a scale set is using a fixed spreading algorithm and if there are less than five fault domains available, the deployment will fail.

The Azure portal creation process does not directly support applying additional configuration management options, such as VM extensions. However, they can be applied to a VMSS later using the command line tools or an ARM template.

More Info Virtual Machine Scale Sets

You can learn more about virtual machine scale sets here:

<https://docs.microsoft.com/azure/virtual-machine-scale-sets/>.

Skill 3.3: Configure VMs

There are multiple ways to create and configure virtual machines, depending on your intended use. The easiest way to create an individual virtual machine is to use the Azure portal. If you have a need for automated provisioning (or you just enjoy the command line), the Azure PowerShell cmdlets and the Azure cross-platform command line interface (CLI) are a good fit. For more advanced automation—even including orchestration of multiple virtual machines—Azure Resource Manager templates can also be used. Each method brings its own capabilities and tradeoffs, and it is important to understand which tool should be used in the right scenario. In this section, we will cover various aspects and

features to efficiently manage VMs and supporting resources in an Azure environment.

This section covers how to:

- Configure Azure Disk Encryption
- Move VMs from one resource group to another
- Manage VM sizes
- Add data discs
- Configure networking
- Redeploy VMs

Configure Azure Disk Encryption

Encrypting Azure VM disk was always a difficult task, and you had to rely on the Azure AD app to perform the work. Now you have a straightforward way to encrypt your Azure VM disks using integration with Azure Key Vault. In this section, you will learn how to manage Azure Disk Encryption with few scenarios using Azure portal. Please note these steps can be performed using PowerShell or Azure CLI.

More Info Charges for Azure Disk Encryption

There is no charge for encrypting VM disks with Azure Disk Encryption, but there are charges associated with the use of Azure Key Vault. Key Vault pricing can be accessed at <https://azure.microsoft.com/en-in/pricing/details/key-vault/>.

Enable encryption on an existing VM

Follow these steps to enable encryption on an existing VM:

1. Browse to the VM resource in Azure portal and under **Settings**, select

Disks (see Figure 3-22).

examrefVM | Disks
Virtual machine

Search (Ctrl+ /) « Edit Refresh Encryption Swap OS Disk

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings

Networking Connect Disks Size Security Extensions Continuous delivery

Managed disks created since June 10, 2017 are encrypted at rest with Storage Service Encryption (SSE). You may also want to enable Azure Disk Encryption.

Ultra Disk compatibility is not available for this location.

Disk settings

Enable Ultra Disk compatibility ①
 Yes No

OS disk

Name	Size	Storage account type
examrefVM_OsDisk_1_057520c968084f1784a6d47f3730aeac	127 GB	Premium SSD

Data disks

None

+ Add data disk

FIGURE 3-22 Disks blade for Azure VM

- Now, click **Encryption**. Under **Disk To Encrypt**, choose **None**, **OS Disk**, or **OS And Data Disks**, as shown in [Figure 3-23](#).

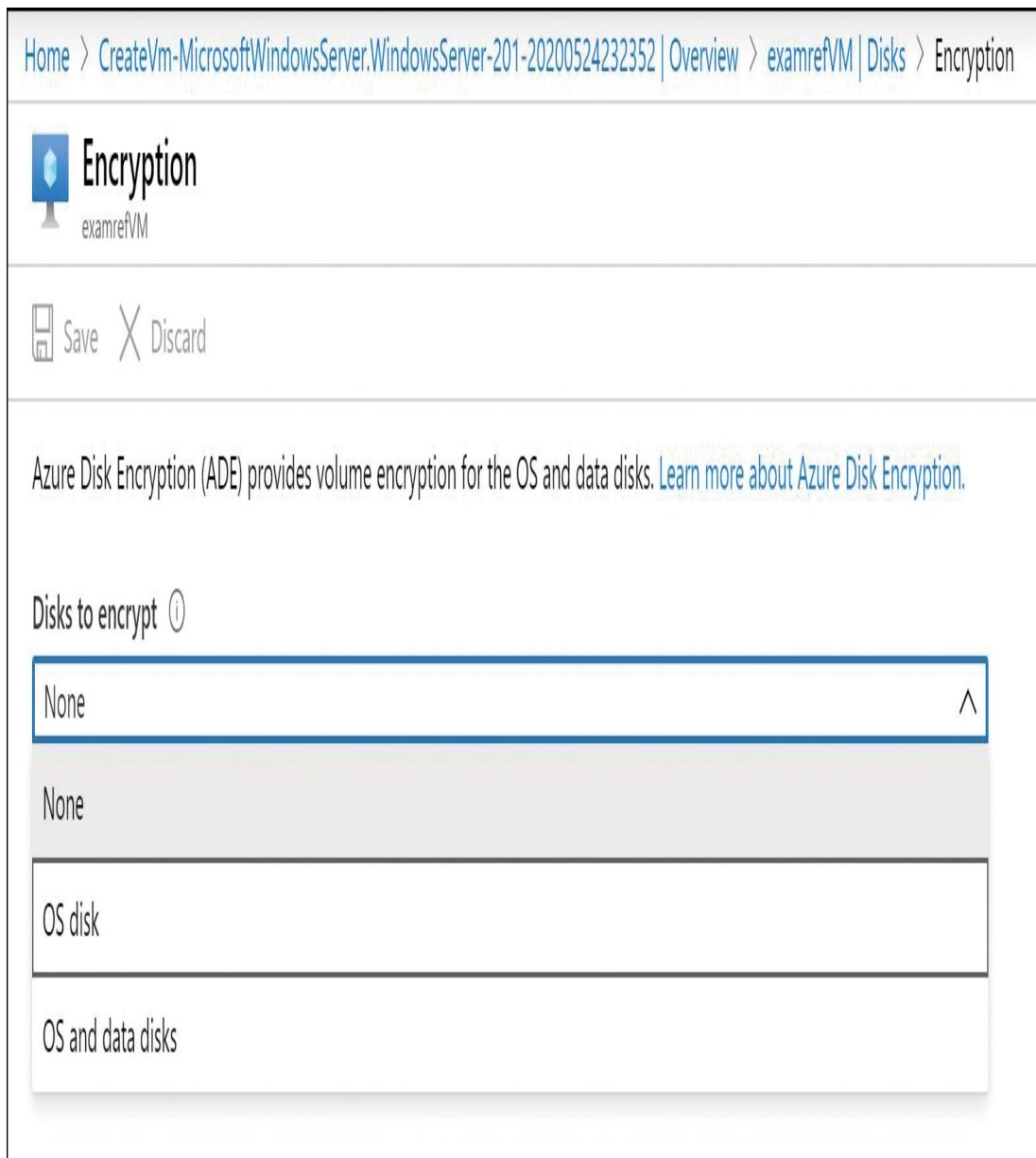


FIGURE 3-23 Encryption options for Azure VM disks

- On the next screen, click **Select A Key Vault And Key For Encryption**

(see Figure 3-24).



Encryption

examrefVM



Save



Discard

Azure Disk Encryption (ADE) provides volume encryption for the OS and data disks. [Learn more about Azure Disk Encryption.](#)

Disks to encrypt (i)

OS disk



Encryption settings

Azure Disk Encryption is integrated with Azure Key Vault to help manage encryption keys. As a prerequisite, you need to have an existing key vault with encryption permissions set. For additional security, you can create or choose an optional key encryption key to protect the secret.

Select a key vault and key for encryption

Key vault* (i)

None

Key (i)

None

Version (i)

None

FIGURE 3-24 Encryption options for Azure VM disks

4. Select **Create New** to create a new Key Vault to store encryption keys (see Figure 3-25).



FIGURE 3-25 Key Vault to store encryption keys

5. Create a new Key Vault by providing name along with other options, as shown in Figure 3-26.

Create key vault

Basics Access policy Networking Tags Review + create

Azure Key Vault is a cloud service used to manage keys, secrets, and certificates. Key Vault eliminates the need for developers to store security information in their code. It allows you to centralize the storage of your application secrets which greatly reduces the chances that secrets may be leaked. Key Vault also allows you to securely store secrets and keys backed by Hardware Security Modules or HSMs. The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. In addition, key vault provides logs of all access and usage attempts of your secrets so you have a complete audit trail for compliance. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription

Visual Studio Ultimate with MSDN



Resource group*

examrefRG



[Create new](#)

Instance details

Key vault name * ①

examrefKV



Region

(Canada) Canada East



Pricing tier * ①

Standard



Soft delete ①

[Enable](#) [Disable](#)

Retention period (days) * ①

90

Purge protection ①

[Enable](#) [Disable](#)

[Review + create](#)

[< Previous](#)

[Next : Access policy >](#)

FIGURE 3-26 Create Key Vault

6. On the **Access Policies** tab, select **Azure Disk Encryption For Volume Encryption** (see [Figure 3-27](#)).



FIGURE 3-27 Access policy options while creating Key Vault

7. Select **Review + Create**. After the Key Vault has passed validation, select

Create. This will return you to the **Select Key From Azure Key Vault** screen.

8. Click **Create New** to generate new secret key (see [Figure 3-28](#)).
9. On next screen, verify the details and click **Select** (see [Figure 3-29](#)). You will be redirected to Encryption blade with the newly created Key Vault and key, as shown in [Figure 3-30](#).

Create a key

Options

Generate



Name * ⓘ

examrefKey



Key Type ⓘ

RSA

EC

RSA Key Size

2048

3072

4096

Set activation date? ⓘ

Set expiration date? ⓘ

Enabled?

Yes

No

FIGURE 3-28 Create A Key

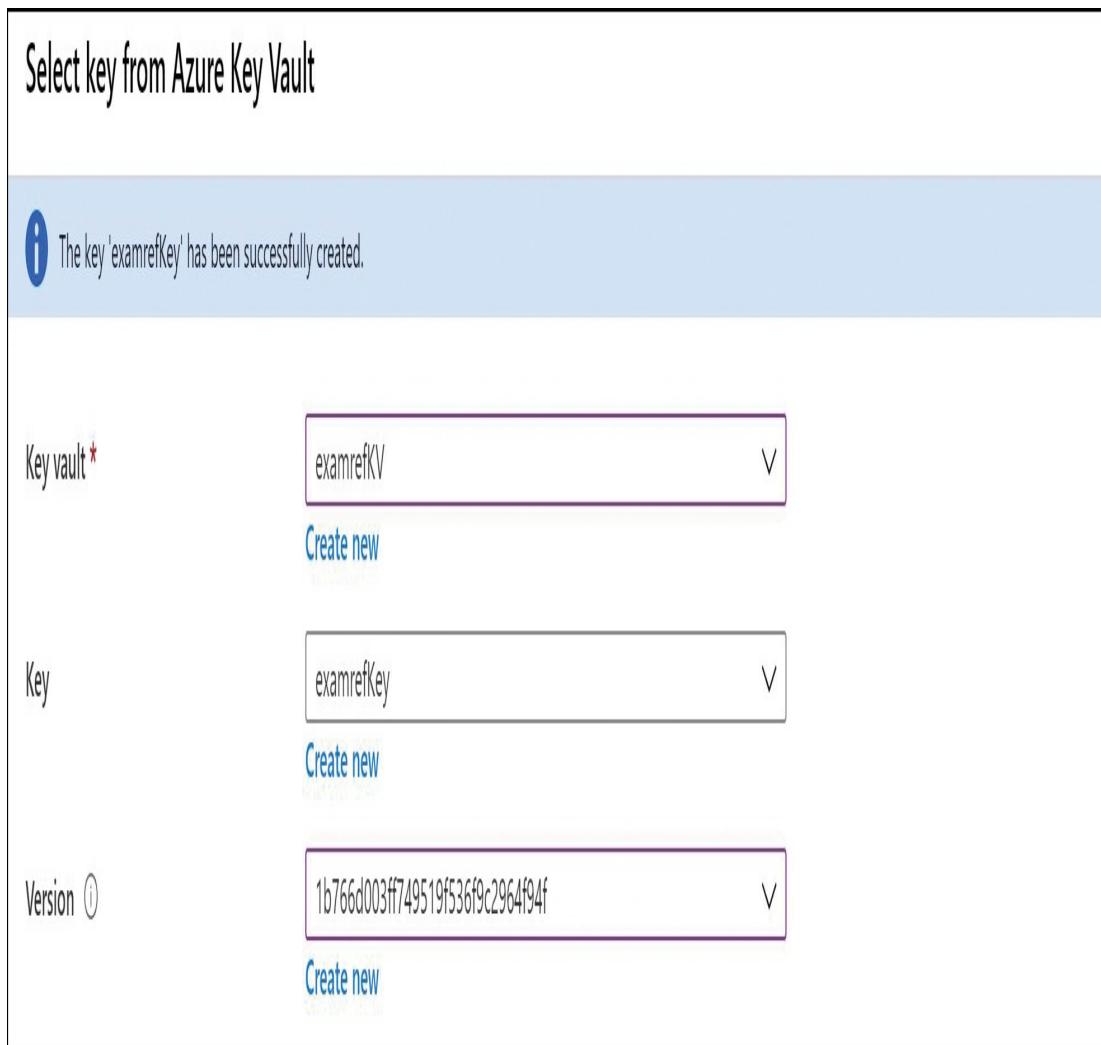


FIGURE 3-29 Select a key



Encryption

examrefVM



Save



Discard

Azure Disk Encryption (ADE) provides volume encryption for the OS and data disks. [Learn more about Azure Disk Encryption.](#)

Disks to encrypt (i)

OS disk



Encryption settings

Azure Disk Encryption is integrated with Azure Key Vault to help manage encryption keys. As a prerequisite, you need to have an existing key vault with encryption permissions set. For additional security, you can create or choose an optional key encryption key to protect the secret.

Select a key vault and key for encryption

Key vault* (i)

/subscriptions/00b72028-9dce-4729-9b2a-a10e92054447/resourceGroups/examrefRG/providers/Microsoft.KeyVault/vaul...✓

Key (i)

examrefKey

Version (i)

1b766d003ff749519f536f9c2964f94f

FIGURE 3-30 Encryption settings for Azure Disk Encryption

- When you attempt to save your encryption settings, you will be asked to restart your VM to make the changes take effect (see [Figure 3-31](#)).

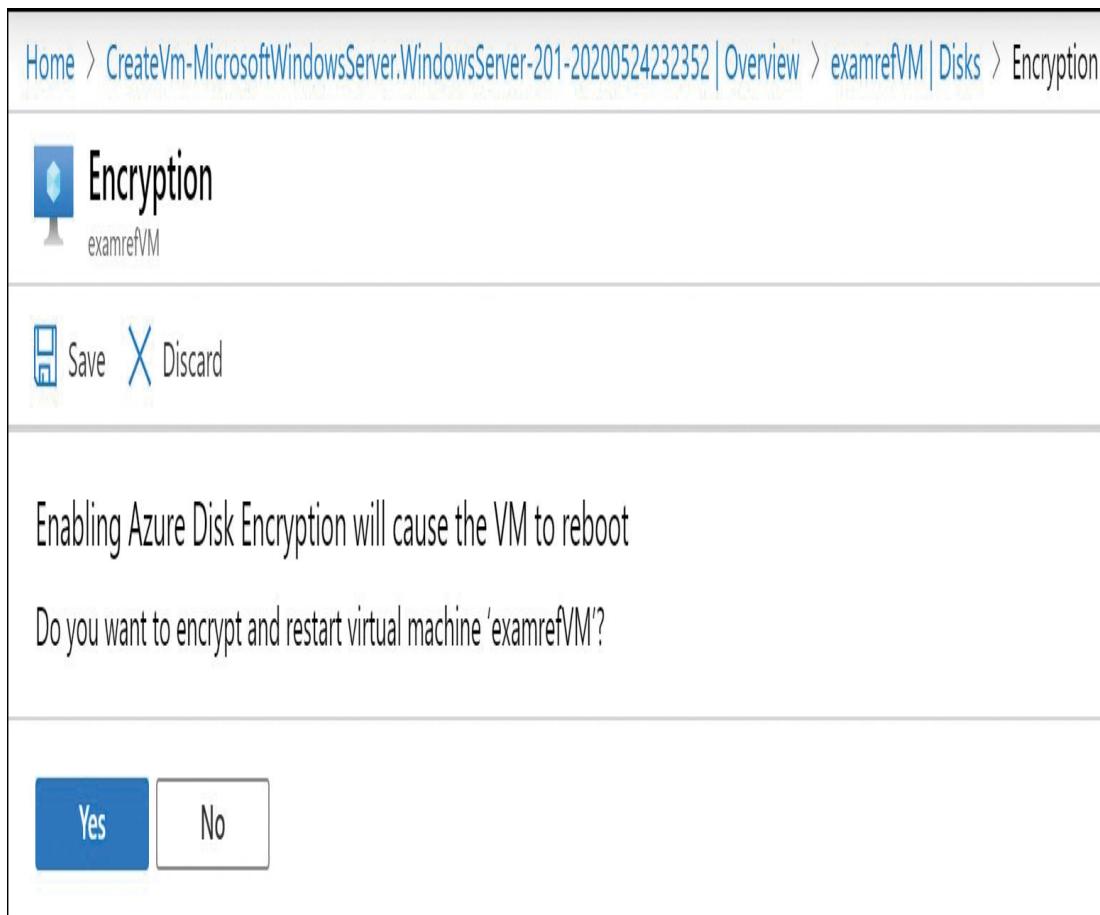


FIGURE 3-31 Save encryption settings

- As stated earlier, once your VM is restarted, the selected (OS or data) disks will be encrypted with the supplied keys.

More Info Disk encryption for Windows and Linux VMs

Windows VMs: <https://docs.microsoft.com/en-us/azure/virtual-machines/windows/disk-encryption-overview>

Linux VMs: <https://docs.microsoft.com/en-us/azure/virtual-machines/linux/disk-encryption-overview>

Create new VM with customer-managed encryption keys

You can enable encryption with customer-managed keys (known as CMK) while creating a new VM. Before creating the VM, you need to create disk encryption set first.

Follow these steps to create a disk encryption set using Azure portal.

1. Click **Create A Resource** on the homepage and choose **Disk Encryption Set > Create**.
2. On the next screen, select the **Subscription**, **Resource Group**, and **Region**, and then specify the **Disk encryption Set Name**.
3. Also, you will need to select a Key Vault and key from the **Key Vault And Key** drop-down menu (if you've already created one), as shown in [Figure 3-32](#). If you have not created a Key Vault and key for encryption, create them before creating a disk encryption set.
4. Once your disk encryption set is created, proceed with creating an Azure VM with the desired settings.

Create a disk encryption set

Basics Tags Review + create

Disk encryption sets allow you to manage encryption keys using server-side encryption for Standard HDD, Standard SSD, and Premium SSD managed disks. It will give you control of the encryption keys to meet your security and compliance needs in a few clicks. [Learn more about disk encryption sets.](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Resource group * ⓘ

examrefRG

[Create new](#)

Instance details

Disk encryption set name *

examrefDESet

Region * ⓘ

(Canada) Canada East

Key vault and key *

Key vault: examrefKV

Key: examrefKey

Version: 1b766d003ff749519f536f9c2964f94f

[Click to select a key](#)

[Review + create](#)

< Previous

Next : Tags >

FIGURE 3-32 Create a disk encryption set

5. On the **Disks** tab, under **Encryption Type**, choose **Encryption At-Rest With A Customer-Managed Key**, as shown in [Figure 3-33](#).
6. Select the **Disk Encryption Set** created in the step 1. You can apply these settings to each of the disks you choose to create.
7. Click **Review + Create** to create the VM with disk encryption using a customer-managed key.

Create a virtual machine

Basics **Disks** Networking Management Advanced Tags Review + create

Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. [Learn more](#)

Disk options

OS disk type * ⓘ

Premium SSD

Encryption type *

Encryption at-rest with a customer-managed key

(Default) Encryption at-rest with a platform-managed key

Encryption at-rest with a customer-managed key

⚠ Once a customer-managed key is used, it cannot be changed.

[Learn more about disk encryption.](#)

Disk encryption set * ⓘ

examrefDESet

Enable Ultra Disk compatibility ⓘ

Yes No

Ultra Disk compatibility is not available for this VM size and location.

Data disks

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

LUN	Name	Size (GiB)	Disk type	Host caching
-----	------	------------	-----------	--------------

[Create and attach a new disk](#) [Attach an existing disk](#)

✓ Advanced

[Review + create](#)

[< Previous](#)

[Next : Networking >](#)

FIGURE 3-33 Create a VM by enabling disk encryption using customer-managed keys

More Info Disk encryption with customer-managed keys

Once a customer-managed key is used, you cannot change the selection back to a platform-managed key.

Enable encryption on a newly added data disk

When you choose to add a new data disk to an existing VM or a new VM, you can select the encryption using a customer-managed key, as shown in [Figure 3-34](#).

Create a managed disk

Create a new disk to store applications and data on your VM. Disk pricing varies based on factors including disk size, storage type, and number of transactions.

Disk name * ⓘ

examrefDataDisk



Resource group * ⓘ

examrefRG



[Create new](#)

Location

Canada East

Availability zone ⓘ

None

Source type ⓘ

None



Size * ⓘ

1024 GiB

Premium SSD

[Change size](#)

Encryption type *

Encryption at-rest with a customer-managed key



(Default) Encryption at-rest with a platform-managed key

Encryption at-rest with a customer-managed key

[Learn more about disk encryption.](#) ↗

Disk encryption set * ⓘ

examrefDESet



[Create](#)

FIGURE 3-34 Enable disk encryption using customer-managed keys for data disk

Disable encryption

When you choose to disable encryption for OS and data disks for an existing VM, you can set the **Disks To Encrypt** option to **None**, as shown in [Figure 3-35](#).

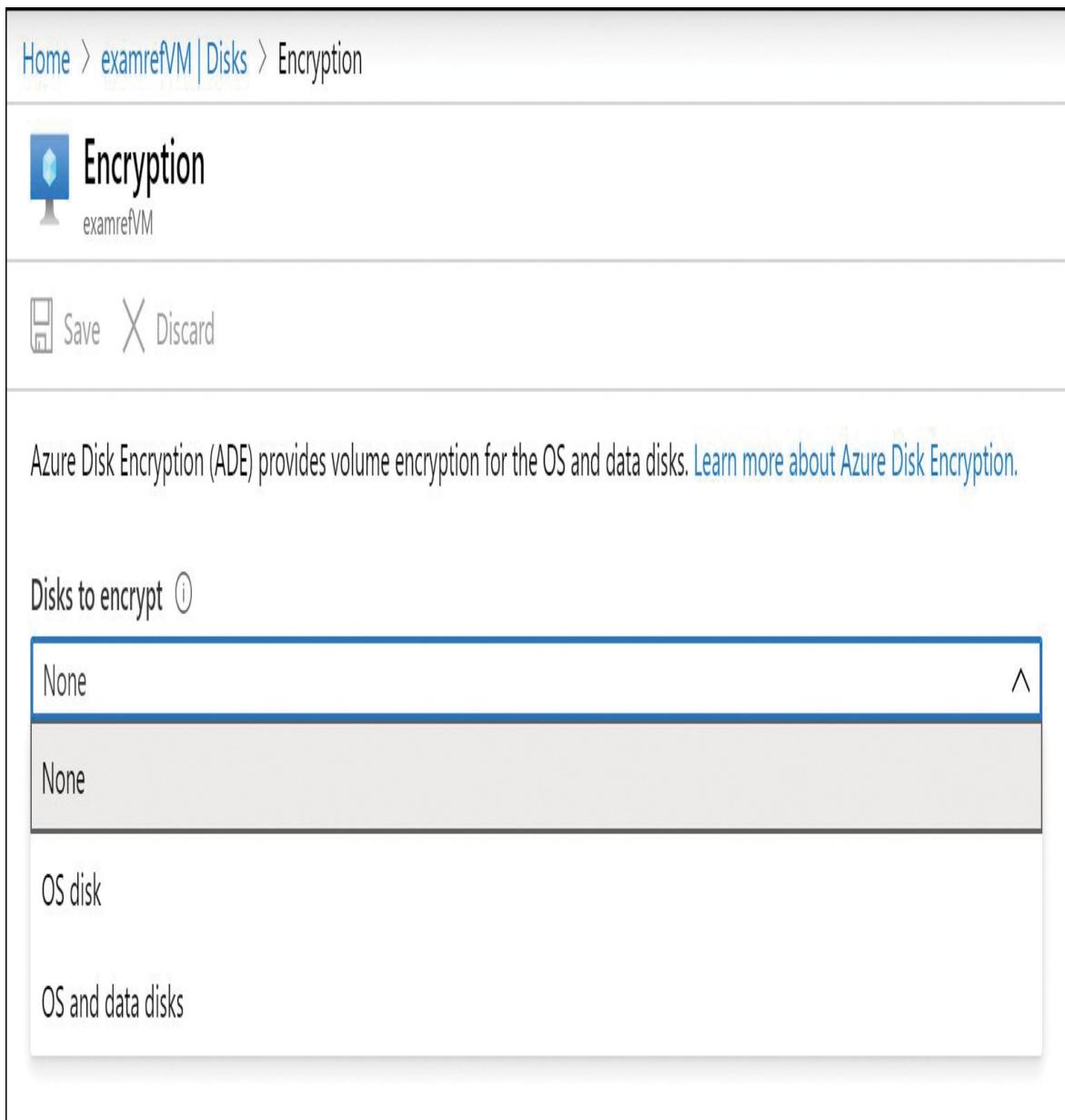


FIGURE 3-35 Disable disk encryption

Move VMs from one resource group to another

Azure provides the ability to move some resources from one subscription to another or from resource group to resource group.

Follow these steps to move a virtual machine using the Azure portal:

1. Open the VM blade, as shown in [Figure 3-36](#), and click the **Change** link next to the **Resource Group** label or click the **Change** button next to the **Subscription** label.

Home > examrefVM

 examrefVM
Virtual machine

Search (Ctrl+ /) <

Connect Start Restart Stop Capture Delete Refresh

i Advisor (1 of 2): Enable virtual machine backup to protect your data from corruption and accidental deletion →

Resource group (change)	: examrefRG
Status	: Running
Location	: Canada East
Subscription (change)	: Visual Studio Ultimate with MSDN
Subscription ID	: 00b72028-9dce-4729-9b2a-a10e92054447
Computer name	: examrefVM
Operating system	: Windows (Windows Server 2016 Datacenter)
Size	: Standard DS1 v2 (1 vcpus, 3.5 GiB memory)
Tags (change)	: Click here to add tags

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Networking

Connect

Disk

FIGURE 3-36 The virtual machine blade with the change options

visible for Resource Group and Subscription

2. Clicking the **Change** link next to the **Resource Group** name will bring up the **Move Resources** blade, as shown in [Figure 3-37](#). This blade shows the resources related to the virtual machine, such as disks, network security groups, network interfaces, and so on. From here, you can select the individual resources to move to the destination resource group.
3. The **Destination Resource Group** can be selected from the drop-down menu if it is created already; otherwise, it can be created using the **Create A New Group** option at the bottom right (see [Figure 3-37](#)).
4. Accept the terms and click **OK** to start the resource movement.
5. Because the resource group will change, any existing scripts that target resources in this resource group will no longer work until they have been updated. The Azure portal prompts you to confirm that you are aware of this change before you can continue with the move.

Home > examrefVM > Move resources

Move resources

Resources to move

 examrefVM	Virtual machine
---	-----------------

Related resources to move (optional)

<input type="checkbox"/> Select all	Type
 WebAVSet	Availability set
 WebAVSet1	Availability set
 WebAVSet2	Availability set
 examrefDESet	Disk Encryption Set
 examrefVM_OsDisk_1_057520c968084f1784a6d473730aeac	Disk
 shutdown-computevm-examrefVM	microsoft.dev/testlab/schedules
 examrefKV	Key vault
 examrefvm76	Network interface
 examrefVM-nsg	Network security group
 examrefVM-ip	Public IP address
 examrefRG-vnet	Virtual network
 examrefiq	Storage account

Move these resources to

Resource group *

I understand that tools and scripts associated with moved resources will not work until I update them to use new resource IDs ⓘ

FIGURE 3-37 The Move Resources blade showing the related resources for a virtual machine

More Info How to get resource ID

6. You can retrieve the resource ID using Azure portal, PowerShell, or CLI. To get the resource ID from Azure portal, you need to go to resource and then navigate to the resource's properties. You will find the resource ID on the right side.

More Info Supported resources for moving

Not all resources are fully supported moving between resource groups and subscriptions, and there are several caveats regarding virtual machines. See the following for more details at <https://docs.microsoft.com/en-in/azure/azure-resource-manager/management/move-support-resources>.

Manage VM sizes

There are many situations where the amount of compute processing your workload needs varies dramatically from day to day or even hour to hour. For example, in many organizations line of business (LOB) applications are used heavily during the workweek, but on the weekends, they see little actual usage. Other examples are workloads that require more processing time due to scheduled events such as backups or maintenance windows where having more compute time may make it faster to complete these tasks. Azure provides purpose built virtual machine sizes. This means that each family is designed for specific purposes to make it easier for you to choose the right VM size for the right workload.

The different types are:

- **General Purpose** This size type is most suitable for small- to medium-scale development environments. It has a balanced CPU-to-memory ratio. As name suggest recommended for a general use.

- **Compute Optimized** This size type has a higher CPU compared to memory and can be used for CPU-intensive workloads in medium-scale environments. This is ideal for network appliances or batch processes in small environments.
- **Memory Optimized** This size type provides higher memory compared to CPU and is ideal for medium-scale database servers. With high memory, these sizes can be used for caches, or it can be used in memory analytics.
- **Storage Optimized** This size type offers high disk throughput and IO, which makes it a good fit for large transactional databases, such as Cassandra, MongoDB, and so on. Also, it can be used for Big Data and data warehousing.
- **GPU Optimized** This size type provides VMs with one or many NVIDIA GPUs. It provides high compute and graphics, which are ideal for visualization workloads.
- **High Performance Compute** This size type is capable of handling batch processing, molecular modeling, and fluid dynamics. This size type offers substantial CPU power and diverse options for low-latency RDMA networking using FDR InfiniBand and several memory configurations to support memory-intensive computational requirements.

Azure virtual machines make it relatively easy to change the size of a virtual machine, even after it has been deployed. There are a few things to consider with this approach.

The first consideration is to ensure that the region your VM is deployed to supports the instance size that you want to change the VM to. In most cases this is not an issue, but if you have a use case where the desired size isn't in the region to which the existing VM is deployed, your only options are to either wait for the size to be supported in the region or to move the existing VM to a region that already supports it.

The second consideration is whether the new size is supported in the current hardware cluster in which your VM is deployed. This can be determined by clicking the **Size** link in the virtual machine configuration blade in the Azure portal of a running virtual machine, as [Figure 3-38](#) demonstrates. If the size is available, you can select it. Changing the size reboots the virtual machine.

Home > examrefVM | Size

examrefVM | Size

Virtual machine

Search (Ctrl+ /) Clear all filters

Overview Search by VM size... Clear all filters

Size : Small (0-6) **Generation : 2 selected** **Family : General purpose** **Premium disk : Supported** Add filter

Showing 11 of 124 VM sizes. | Subscription: Visual Studio Ultimate with MSDN | Region: Canada East | Current size: Standard_DS1_v2

VM Size ↑↓	Offering ↑↓	Family ↑↓	vCPUs ↑↓	RAM (GiB) ↑↓	Data disks ↑↓	Max IOPS ↑↓	Temporary storage (GiB) ↑↓	Premium disk support ↑↓
B1ls	Standard	General purpose	1	0.5	2	160	4	Yes
B1ms	Standard	General purpose	1	2	2	640	4	Yes
B1s	Standard	General purpose	1	1	2	320	4	Yes
B2ms	Standard	General purpose	2	8	4	1920	16	Yes
B2s	Standard	General purpose	2	4	4	1280	8	Yes
B4ms	Standard	General purpose	4	16	8	2880	32	Yes
D2s_v3	Standard	General purpose	2	8	4	3200	16	Yes
D4s_v3	Standard	General purpose	4	16	8	6400	32	Yes
DS1_v2	Standard	General purpose	1	3.5	4	3200	7	Yes
DS2_v2	Standard	General purpose	2	7	8	6400	14	Yes
DS3_v2	Standard	General purpose	4	14	16	12800	28	Yes

Resize Prices presented are estimates in your local currency that include only Azure infrastructure costs and any discounts for the subscription and location. The prices don't include any charges will appear in your local currency in cost analysis and billing views. [View Azure pricing calculator](#).

- Search (Ctrl+ /)
- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings
- Networking
- Connect
- Disks
- Size**
- Security
- Extensions
- Continuous delivery
- Availability + scaling
- Configuration
- Identity
- Properties
- Locks
- Export template
- Operations
- Bastion
- Auto-shutdown
- Backup

FIGURE 3-38 Changing the size of an Azure virtual machine using the Azure portal

If a desired size is not available, it means either the size is not available in the region or on the current hardware cluster. You can view the available sizes by region at <https://azure.microsoft.com/regions/services/>. If you need to change to a different hardware cluster, you must first stop the virtual machine, and if it is part of an availability set, you must stop all instances of the availability set at the same time. After all the VMs are stopped, you can then change the size, which moves all the VMs to the new hardware cluster as they are resized and started. All VMs in the availability set must be stopped before performing the resize operation to a size that requires different hardware because all running VMs in the availability set must use the same physical hardware cluster. Therefore, if you are required to change a physical hardware cluster in order to change the VM size, all VMs must be stopped and then restarted one-by-one to a different physical hardware cluster.

More Info Virtual machine sizes

There are a lot of considerations when choosing the correct virtual machine size. For more information on sizes in the context of Windows-based virtual machines see <https://docs.microsoft.com/azure/virtual-machines/windows/sizes>. For the Linux version of the article, see <https://docs.microsoft.com/azure/virtual-machines/linux/sizes>.

Add data disks

Adding a data disk to an existing Azure virtual machine using the Azure portal is almost identical to the creation process. From within the virtual machine configuration blade, click **Disks**, and then click **Add Data Disk**. This action will open the dialog displayed in [Figure 3-39](#). From there, you can choose one of the existing disks that are available to attach, or you can click **Create** to create a new disk that will walk through the create disk user experience.

Home > examrefVM | Disks

examrefVM | Disks

Virtual machine

Search (Ctrl+ /) Save Discard Refresh Encryption Swap OS Disk

Managed disks created since June 10, 2017 are encrypted at rest with Storage Service Encryption (SSE). You may also want to enable Azure Disk Encryption.

Ultra Disk compatibility is not available for this location.

Disk settings

Enable Ultra Disk compatibility Yes No

OS disk

Name	Size	Storage account type	Encryption	Host caching
examrefVM_OsDisk_1_057520e968084f17846d47f3730aeac	127 GB	Premium SSD	SSE with PMK	Read/write

Data disks

LUN	Name	Storage account type	Encryption	Host caching
0	<input type="text"/> !	Not enabled	None	...

The value must not be empty.

+ Add data disk

Create disk

Disks in resource group 'examrefRG'

examrefDataDisk

size: 1024 GiB, account type: Premium SSD, disk shares used 0 of 1

All disks

examrefDataDisk

size: 1024 GiB, account type: Premium SSD, resource group: EXAMREFRG, disk shares used 0 of 1

FIGURE 3-39 Adding data disk to an Azure virtual machine in the Azure portal

If your virtual machine was created with managed disks enabled, you will see the **Create A Managed Disk** blade shown in [Figure 3-40](#). From here, you can specify the **Name** of the disk, the **Resource Group**, the **Source** type, the **OS** type, **Size** and **Encryption type**.

You can use the following source types to create a new managed disk:

- **Snapshot** If selected, you can browse for snapshots in the current subscription and location.
 - **Storage Blob** If selected, you can browse storage accounts in all subscriptions you have access to, so you can select the VHD.
 - **None** If selected, a new empty VHD is created.
-



Exam Tip

If the virtual machine is deployed into an availability zone, use the `Zone` parameter with the `New-AzDiskConfig` cmdlet to specify which availability zone in which to create the disk if you are creating the disk using PowerShell.



Exam Tip

If the virtual machine is deployed into an availability zone, the disk is automatically placed into the same zone as the virtual machine using Azure CLI.

Create a managed disk

Create a new disk to store applications and data on your VM. Disk pricing varies based on factors including disk size, storage type, and number of transactions.

Disk name * ⓘ

examrefDataDiskExample



Resource group *

examrefRG



[Create new](#)

Location

Canada East

Availability zone ⓘ

None

Source type ⓘ

Storage blob



Source subscription

Visual Studio Ultimate with MSDN



Source blob * ⓘ

[Browse](#)

OS type ⓘ

Windows Linux **None (data disk)**

Size * ⓘ

1024 GiB

Premium SSD

[Change size](#)

Encryption type *

(Default) Encryption at-rest with a platform-managed key



Create

FIGURE 3-40 The Create Managed Disk blade in the Azure portal

Configure networking

During the virtual machine provisioning process in the Azure portal you can set the following options using the Networking blade, as shown in [Figure 3-41](#).

- The virtual network, subnet, and the public IP address
- The network security group for the network interface card (NIC)
- The public inbound ports that should be open (if any)
- If accelerated networking should be enabled
- If the VM should be included in an existing Azure Load Balancer back-end pool

Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.

[Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * ⓘ

examrefRG-vnet



[Create new](#)

Subnet * ⓘ

default (10.0.1.0/24)



[Manage subnet configuration](#)

Public IP ⓘ

(new) examrefDemoVM-ip



[Create new](#)

NIC network security group ⓘ

None Basic Advanced

Public inbound ports * ⓘ

None Allow selected ports

Select inbound ports

Select one or more ports



All traffic from the internet will be blocked by default. You will be able to change inbound port rules in the VM > Networking page.

Accelerated networking ⓘ

On Off

The selected VM size does not support accelerated networking.

Load balancing

You can place this virtual machine in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Place this virtual machine behind an existing load balancing solution?

Yes No

[Review + create](#)

[< Previous](#)

[Next : Management >](#)

FIGURE 3-41 Specifying the networking options for a virtual machine during creation

More Info Network security groups

A network security group (NSG) is a networking filter containing a list of security rules which control network traffic when applied. These rules can manage both inbound and outbound traffic. A network security group can be associated to a network interface, the subnet the network interface is in, or both. To simplify management of security rules, it's recommended that whenever possible, you associate a network security group to individual subnets, rather than individual network interfaces within the subnet. You will learn more about NSGs in [Chapter 4, “Configure and manage virtual networks.”](#)

There are several networking features you need to understand to effectively use Azure virtual machines, as well as to prepare for the exam. In this chapter, you will learn some of these concepts; in [Chapter 4, “Configure and manage virtual networking,”](#) you will learn in-depth about virtual networks, network security groups, subnets, IP addresses, and DNS management.

More Info Application security groups

An application security group (ASG) enables you to define network security policies based on workloads with rules focused on applications instead of IP and network addresses. They allow you to group virtual machines with monikers and secure applications by filtering traffic from trusted segments of your network. Like NSGs, you will learn more about ASGs in [Chapter 4, “Configure and manage virtual networks.”](#)

Accelerated networking

Accelerated networking enables single root I/O virtualization (SR-IOV) to a virtual machine, which greatly improves its networking performance. This feature improves performance by bypassing the virtual switch between the host VM and the physical switch. [Figure 3-42](#) shows two deployments: The

deployment on the left does not have accelerated networking, and the deployment on the right has accelerated networking enabled.

Accelerated networking can be enabled at the time of creation or after the virtual machine is created, if the following pre-requisites are met:

- The VM must be a supported size for accelerated networking.
- The VM must use a supported Azure Gallery image (and use a supported kernel version if you are using Linux). You can use a custom image as long as you have the appropriate drivers.
- All VMs in an availability set or VMSS must be stopped/deallocated before enabling accelerated networking on any NIC.

Supported sizes:

- Accelerated networking is supported on most general-purpose and compute-optimized instance sizes with two or more vCPUs. D/DSv2 and F/Fs series are supported.
- On instances that support hyperthreading, accelerated networking is supported on VM instances with four or more vCPUs. The following series are supported: D/DSv3, E/ESv3, Fsv2, and Ms/Mms.

The following Windows-based images from the Azure Marketplace are supported:

- Windows Server 2019 Datacenter
- Windows Server 2016 Datacenter
- Windows Server 2012 R2 Datacenter

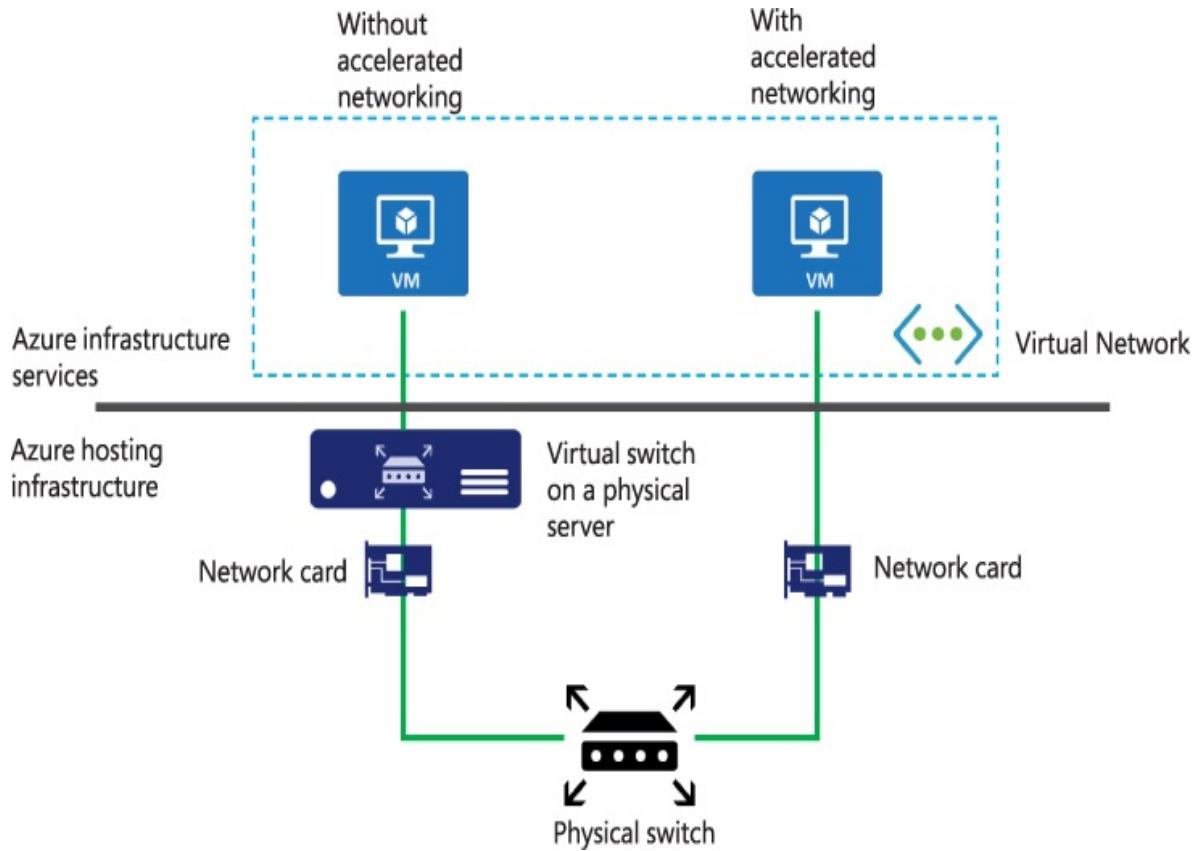


FIGURE 3-42 Virtual machines deployed with and without accelerated networking enabled

Supported Linux-based images from the Azure Marketplace:

- Ubuntu 16.04 or later
- Ubuntu 14.04 with the Linux-Azure kernel
- SLES 12 SP3 or later
- RHEL 7.4 or later
- CentOS 7.4 or later
- CoreOS Linux
- Debian “Stretch” with backports kernel
- Oracle Linux 7.4 and later with Red Hat Compatibility Kernel
- Oracle Linux 7.5 and later with UEK version 5
- FreeBSD 10.4, 11.1 and 12.0

More Info Learning more about accelerated networking

You can learn more about using accelerated networking on Windows- and Linux-based virtual machines:

- Windows <https://docs.microsoft.com/azure/virtual-network/create-vm-accelerated-networking-powershell>.
- Linux-based VMs <https://docs.microsoft.com/azure/virtual-network/create-vm-accelerated-networking-cli>.

Connecting to virtual machines

There are many ways to connect to virtual machines. You should consider options such as connecting to VMs using their public IP addresses and protecting VMs with network security groups and allowing only the port for the service you are connecting to. You should also understand how to connect to a VM on its private IP address. This introduces additional connectivity requirements, such as ExpressRoute, Site-to-Site VPN, or Point-to-Site VPN to put your client on the same network as your VMs. These technologies are discussed in [Chapter 4](#). In this section, we'll review the most common tools to connect and manage your VMs.

Authentication options

For Windows-based virtual machines, usernames can be a maximum of 20 characters in length and cannot end in a period (“.”). Many common usernames are blocked during the creation process. Examples of blocked account names include: 1, 123, a, admin, administrator, john, and several other easily guessable names. Passwords must be between 12 and 123 characters in length, and they must meet several complexity requirements.

For Linux-based virtual machines, you can specify an existing SSH public key or a password when creating a Linux VM. Linux usernames must be between 1 and 32 characters in length, and passwords must be between 6 and 72 characters. Like Windows, certain easily guessable usernames and passwords are automatically blocked when creating through the Azure portal.

If you choose to use the SSH public key option, you must enter (or paste in) the public key for your SSH certificate. You can create the SSH certificate using the following command:

```
ssh-keygen -t rsa -b 2048
```

To retrieve the public key for your new certificate, run the following command in a bash console:

```
cat ~/.ssh/id_rsa.pub
```

From there, copy all of the data starting with ssh-rsa and ending with the last character on the screen, pasting it into the SSH public key box while creating a virtual machine in the Azure portal. Make sure you don't include any extra spaces.

More Info Learn more about username and password requirements

For information on username and password requirements on Windows VMs, see <https://docs.microsoft.com/azure/virtual-machines/windows/faq> or for Linux-based VM requirements, see: <https://docs.microsoft.com/azure/virtual-machines/linux/faq>.

Add network interfaces

A network interface enables an Azure virtual machine to communicate with the Internet, Azure, and on-premises resources. Common use cases for having multiple network interfaces are as follows:

- **Network and security function** Multiple network interfaces enable virtual network appliances, such as load balancers, firewalls, and proxy servers.
- **Network isolation** Common best practices include isolating public-facing services from internal networks.
- **Bandwidth isolation** In certain cases, such as heartbeat signals, it is important to have isolated traffic to guarantee the minimal amount of bandwidth is available to the workload.

To add a new network interface to an Azure virtual machine, click the **Networking** link in the left navigation pane of the virtual machine configuration blade and then click the **Attach Network Interface** link at the top. The following screen allows you to attach an existing network interface, or you can

click **Create And Attach Network Interface** to create a new one. [Figure 3-43](#) shows the blade for creating a new network interface.

Create network interface

Basics Tags Review + create

Create a network interface and attach it to a virtual machine. A network interface enables a virtual machine to communicate with Internet, Azure, and on-premises resources. [Learn more about network interface](#)

Project details

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

examrefRG

[Create new](#)

Instance details

Name *

examrefNIC

Region *

(Canada) Canada East

Virtual network ⓘ

ExamRefVnet

[Manage selected virtual network](#)

Subnet * ⓘ

default (10.1.0.0/24)

Private IP address assignment

Dynamic Static

Private IP address *

10.1.0.100

Network security group ⓘ

None

Private IP address (IPv6)



[Review + create](#)

[< Previous](#)

[Next : Tags >](#)

[Download a template for automation](#)

FIGURE 3-43 Creating a new network interface in the Azure portal

After the network interface is created, you must first deallocate the virtual machine before you can attach it. You can deallocate the VM by clicking **Stop** in the Azure portal or by using the command-line tools. Then you can attach network interface, as shown in Figure 3-44.

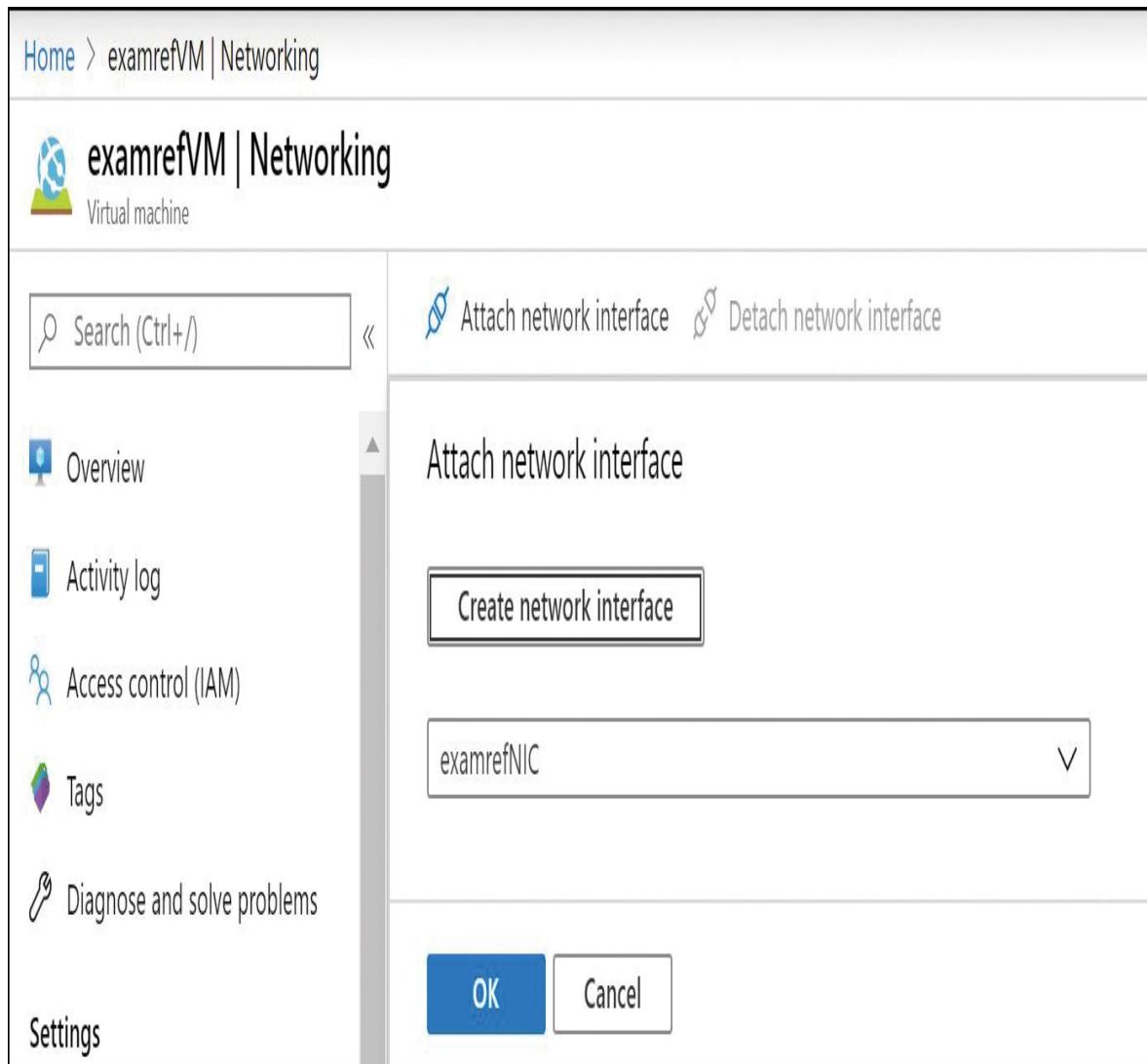


FIGURE 3-44 Attaching a new network interface to a virtual machine

By default, the first network interface attached is defined as the primary network interface. All others are secondary. You can control which network

interface you send outbound traffic to; by default, it will be the primary network interface.

More Info Multiple network interfaces

Beyond understanding the basics of adding network interfaces it is important to understand the nuances and constraints in this area. The Azure documentation does a great job of consolidating and summarizing this information and you can read more about it at

<https://docs.microsoft.com/azure/virtual-network/virtual-network-network-interface-vm>.

Connecting to a Windows VM with remote desktop

The default connectivity option for a Windows-based virtual machine is to use the remote desktop protocol (RDP) and a Remote Desktop client such as mstsc.exe. The RDP service listens on TCP port 3389 and provides full access to the Windows desktop. This service is enabled by default on all Windows-based VMs provisioned from the Azure Marketplace. The Azure portal provides a **Connect** button that will appear enabled for virtual machines that have a public IP address associated with them, as shown in [Figure 3-45](#). You can also use Azure Bastion to connect to the Windows VM. Azure Bastion is discussed in detail in [Chapter 4](#).

You can launch a remote desktop session from Windows PowerShell by using the `Get-AzRemoteDesktopFile` cmdlet. The `Get-AzRemoteDesktopFile` cmdlet performs the same validation as the Azure portal. The API it calls validates that a public IP address is associated with the virtual machine's network interface. If a public IP exists, it generates an `.rdp` file consumable with a Remote Desktop client. The `.rdp` file will have the IP address of the VIP and public port (3389) of the specified embedded virtual machine. There are two parameters that alter the behavior of what happens with the generated file.

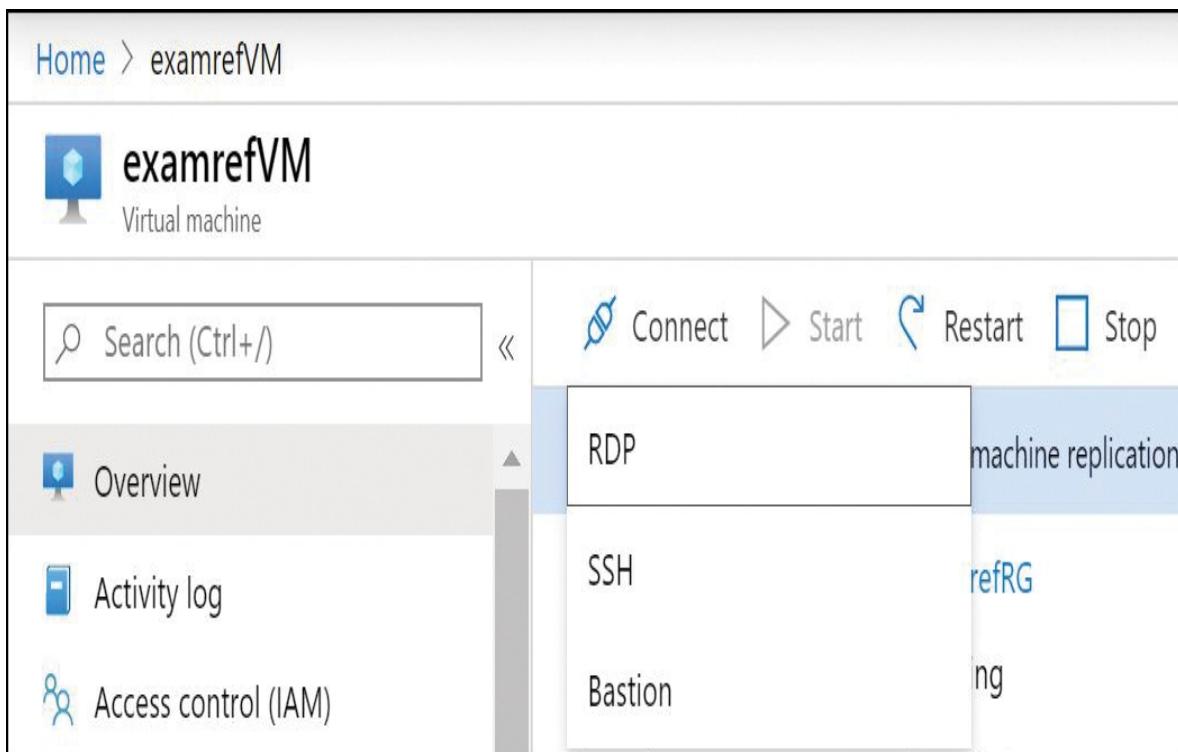


FIGURE 3-45 The Connect button for an Azure VM

Use the `Launch` parameter to retrieve the `.rdp` file and immediately open it with a Remote Desktop client. The following example launches the `Mstsc.exe` (Remote Desktop client), and the client prompts you to initiate the connection.

[Click here to view code image](#)

```
$rgName = "ExamRefRG"  
$vmName = "ExamRefVM"  
Get-AzRemoteDesktopFile -ResourceGroupName $rgName -Name $vmName -  
Launch
```

The second behavior is to specify the `LocalPath` parameter, as the following example shows. Use this parameter to save the `.rdp` file locally for later use.

[Click here to view code image](#)

```
$rgName = "ExamRefRG"  
$vmName = "ExamRefVM"  
$Path = "C:\Scratch\ExamRefVM.rdp"  
Get-AzRemoteDesktopFile -ResourceGroupName $rgName -Name $vmName -  
LocalPath $path
```

Connecting to a Linux VM using SSH

The default connectivity option for a Linux-based virtual machine is to use the secure shell (SSH) protocol. This service listens on TCP port 22 and provides full access to a command line shell. By default, this service is enabled on all Linux-based VMs. When you click the **Connect** button on a Linux-based virtual machine with a public IP associated with it, you see a dialog box advising you to use SSH to connect. [Figure 3-46](#) shows how to connect via SSH to a virtual machine. You can also use Azure Bastion to connect to the Linux VM. Azure Bastion is discussed in detail in [Chapter 4](#).

If the virtual machine is configured for password access, SSH then prompts for the password for the user you specified. If you specified the public key for an SSH certificate during the creation of the virtual machine, it attempts to use the certificate from the `~/.ssh` folder.

Windows users have many options for connecting using SSH. For example, if you install the Windows subsystem for Linux, you will also install an SSH client that can be accessed from the bash command line. You can also install one of many GUI-based SSH clients, such as PuTTY. For more about SSH certificate management and some available clients, see

<https://docs.microsoft.com/azure/virtual-machines/linux/ssh-from-windows>

RDP SSH BASTION

Connect via SSH with client

1. Open the client of your choice, e.g. [PuTTY](#) or [other clients](#).

2. Ensure you have read-only access to the private key.

```
chmod 400 examredadmin.pem
```



3. Provide a path to your SSH private key file. ⓘ

Private key path

```
~/ssh/examredadmin
```

4. Run the example command below to connect to your VM.

```
ssh -i <private key path> examredadmin@40.86.218.216
```



Can't connect?

🔗 [Test your connection](#)

🔧 [Troubleshoot SSH connectivity issues](#)

FIGURE 3-46 Steps to connect to a virtual machine via SSH

More Info Windows SUBSYSTEM FOR LINUX

Windows Subsystem for Linux (known as WSL) provides a way to run any Linux distributions on Windows 10. There are two versions: WSL1 and WSL2. You can find more details at <https://docs.microsoft.com/windows/wsl/install-win10>.

Redeploy VMs

Redeploying the VM might help with troubleshooting issues, such as RDP or SSH connectivity or application access. When you redeploy a VM, it moves the VM to a new host within Azure and reenables. The **Redeploy** button is shown in Figure 3-47.

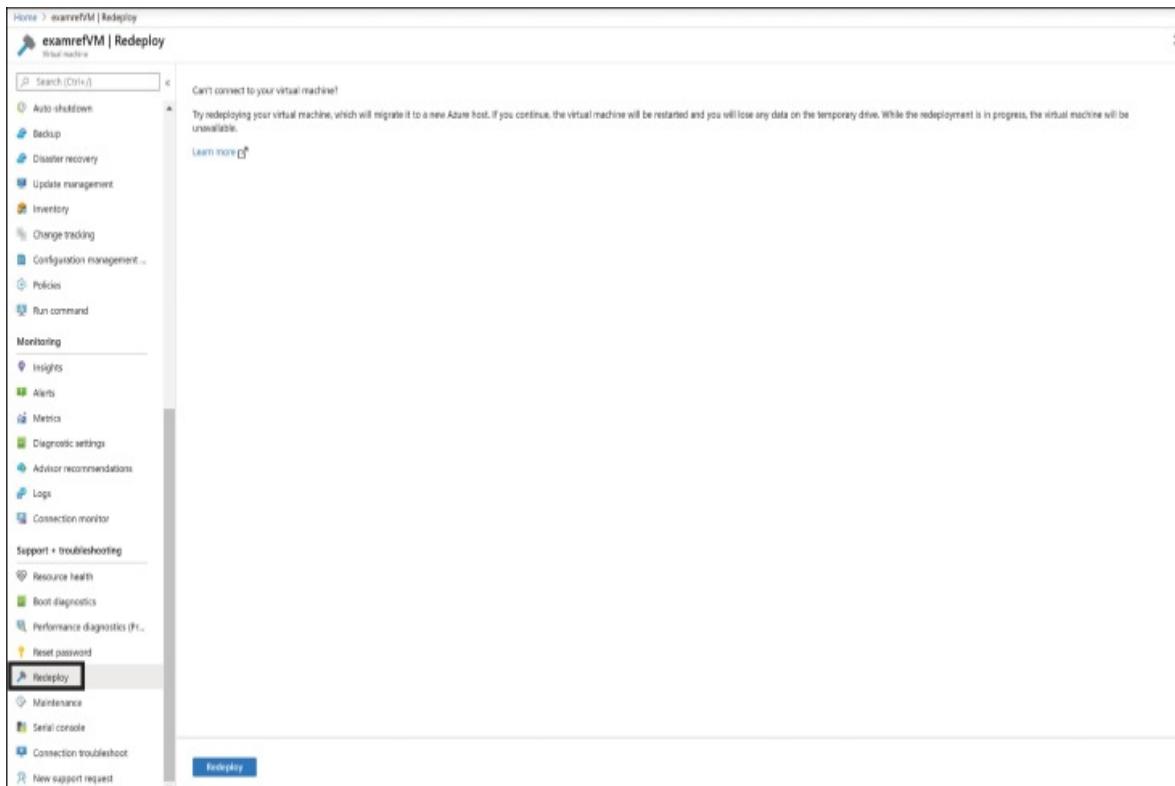


FIGURE 3-47 The redeploy blade in the Azure portal

To redeploy the VM using PowerShell, use the `Set-AzVm` cmdlet, as shown

here:

[Click here to view code image](#)

```
Set-AzVM -Redeploy -ResourceGroupName ExamRefRG -Name ExamRefVM
```

To redeploy with the Azure CLI tools, use the `az vm redeploy` command:

[Click here to view code image](#)

```
az vm redeploy --resource-group ExamRefRG --name ExamRefVM
```

Skill 3.4: Create and configure containers

Now that you have a clear understanding of virtual machines and their place in today's technological world, let's discuss another type of virtualization: Containers.

Containers allow you to package an application and all its dependencies into a compressed package called an *image*. The image can then be uploaded to an image repository. You can then install a container runtime on your computer (or a VM) and point it to the image in the repository. The container runtime will download the image, extract it, and it will then create a container that hosts the application in an isolated environment.

In addition to the application and its dependencies, a container also contains operating system services that the application requires. These operating system services are typically included using a lightweight version of the OS, such as Windows Server Core or Alpine Linux. The container doesn't need the entire operating system because it uses the kernel of the host operating system. For that reason, you can't run a Linux-based container on a Windows computer or vice versa.

The most popular container runtime is Docker. You can install Docker on Windows, MacOS, or Linux. Docker also operates a popular image repository called Docker Hub.

There are many advantages that companies may realize by using containers. Containers require fewer IT resources to deploy, run, and manage. Also, a single computer can run more containers than VMs, so hardware costs can be reduced. Perhaps the greatest advantage of containers is their flexibility. You can easily lift and shift an application between environments using containers, so moving

an application from on-premises to the cloud is simplified.

Azure has many services that utilize containers. In this chapter, we're going to cover two of them: Azure Container Instances and Azure Kubernetes Service. Azure Container Instances makes it easy to create containers in Azure. Simply point Azure Container Instances to a repository and it creates the container for you. It doesn't require that you pay for a VM. Instead, Azure Container Instances is serverless. Once the container is running, you can access it using a public IP address or a DNS name label using the format *label.azure_region.azurecontainer.io*.

Azure Kubernetes Service is a cloud-based implementation of the popular container orchestration service Kubernetes. Kubernetes runs on top of the container runtime, and it can help you to scale and manage a containerized deployment. Configuring Kubernetes on your own is complex and requires specialized knowledge. Using Azure Kubernetes Service is much simpler because Microsoft has done the heavy lifting for you.

This section covers how to:

- [Configure sizing and scaling for Azure Container Instances](#)
- [Configure container groups for Azure Container Instances](#)
- [Configure storage for Azure Kubernetes Service \(AKS\)](#)
- [Configure scaling for AKS](#)
- [Configure network connections for AKS](#)
- [Upgrade an AKS cluster](#)

Configure sizing and scaling for Azure Container Instances

You can create a container instance using the Azure portal or from the command line. One convenient way to create a container using ACI is Azure Cloud Shell. You can do this from within the Azure portal (as shown in [Figure 3-48](#)) or using the Azure mobile app. Cloud Shell allows you to use PowerShell or Bash, and you can use both the Azure CLI and the Az PowerShell module to create and manage Azure resources.

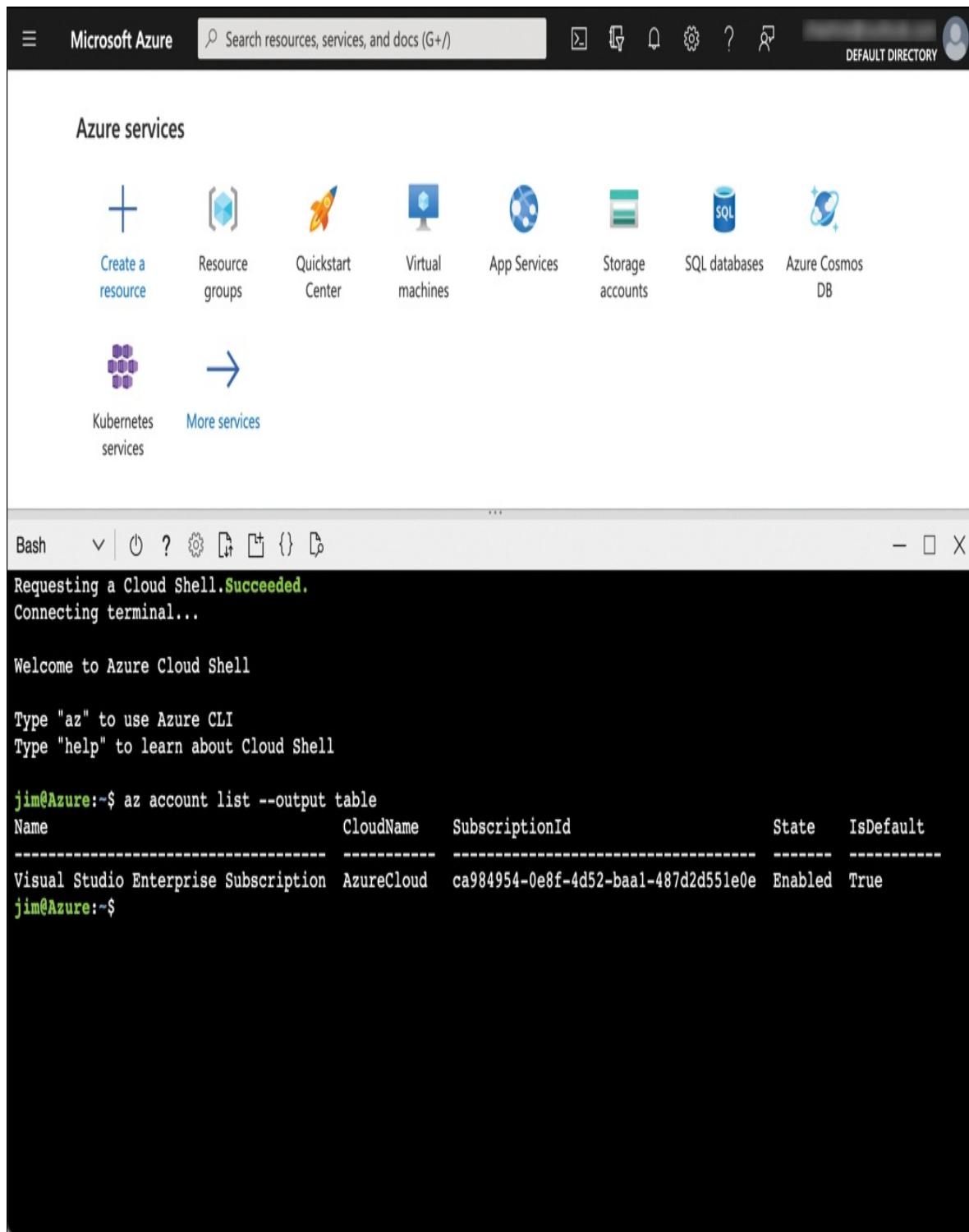


FIGURE 3-48 Azure Cloud Shell in the Azure portal

To create a container in ACI using the CLI, you'll first need to know where

the image you want to use is hosted. If you don't have an image, Microsoft has a sample image you can use at <https://mcr.microsoft.com/azuredocs/aci-helloworld>. The CLI command shown below creates a new container in the AZ104 resource group in the East US region using Microsoft's sample image. The container name is az104container.

[Click here to view code image](#)

```
az container create -g AZ104 --name az104container --image  
mcr.microsoft.com/azuredocs/  
aci-helloworld --dns-name-label az104-test --ports 80
```

Once the container is up and running, it can be accessed with the URL <http://az104-test.eastus.azurecontainer.io> on port 80.

To create the same container using PowerShell, you would run the following command.

[Click here to view code image](#)

```
New-AzContainerGroup -ResourceGroupName AZ104 -Name az104container  
-Image mcr.microsoft.  
com/azuredocs/aci-helloworld -DnsNameLabel az104-test
```

When these commands run, you are actually creating a container group and a container to run inside of that container group. A container group is the top-level object in ACI, and it represents all the containers running on a particular computer. You can, in fact, deploy a container group that runs multiple containers, but doing so requires that you create a deployment in an ARM template. When multiple containers are in a container group, they share the same URL, so you'll need to specify a separate port for each container.



Exam Tip

Multi-container container groups are currently only supported on Linux. A container group that hosts a Windows container can only contain that single container. Microsoft is working on feature parity between Linux and Windows containers, so this will likely change in the future.

By default, a container instance will run on a machine with 1 CPU core and 1.5GB of memory. If you want to change the number of cores or amount of memory, you have to delete the existing container and deploy a new one of the desired size. However, you can scale out by creating additional containers, and you can even use a workflow in Logic Apps to automate that based on conditions that you choose.



Exam Tip

The largest image you can host in ACI is 15GB. You also cannot create a container that uses more than 4 CPU cores and 16GB of memory.

Configure container groups for Azure Container Instances

As stated earlier, a container runs inside of a container group. There may be cases where you need to update a container group. For example, you might need to change the DNS label for a container group. While you can always delete the container group and recreate it, doing so takes longer than simply updating the container group.

To update a container group, use the same command you used to create the group, along with the new property value. If you specify an existing container group when you run your command, Azure will modify the property of the existing container group, and then it will restart the containers within your container group.

More Info Container group properties

Not all properties of a container group can be modified. Some properties

require that you delete and redeploy the container group in order to change them. For more information, see <https://docs.microsoft.com/en-us/azure/container-instances/container-instances-update#properties-that-require-container-delete>.

Configure storage for Azure Kubernetes Service (AKS)

As applications grow, they may span multiple services running in multiple containers, and these containers may be running on multiple servers. This complexity may become difficult to manage with Azure Container Instances. AKS includes an orchestrator that can help you to manage the complexity of a multi-container deployment.

More Info K8s

Kubernetes is often shortened to “K8s,” with the “8” representing the 10 letters between the “K” and the “s.” You’ll hear K8s pronounced different ways, but the most common pronunciations are “kates,” “k-eights,” and “k 8 s.”

AKS deployments run in a cluster, and each computer in the cluster is referred to as a *node*. There is a single node that’s responsible for the other nodes in the cluster, and that node is commonly referred to as the *control plane*. The other computers in the cluster are most commonly referred to as nodes.

Inside of a cluster, you’ll find one or more containers. These containers run inside of a *pod*. A pod can run a single container, but it can also run multiple containers. When multiple containers are running in a pod, they share storage and a single IP address. Therefore, if multiple containers require interoperability and the ability to share resources, it makes the most sense to deploy them all to the same pod.

Before we can discuss scaling, sizing, or networking capabilities of AKS, we need to understand that any applications running in AKS may need to read and write data. Therefore, they will need storage.

Azure Disks can be used for storage with AKS. To create and configure an Azure disk using the Azure CLI, use the code below. This command will create the disk and then display the ID of the disk in the output.

[Click here to view code image](#)

```
az disk create --resource-group AZ-104 --name 104disk --size-gb 50  
--query id --output tsv
```



Exam Tip

The resource group you specify when creating your disk is the resource group for the AKS cluster. If you don't know the resource group of your AKS cluster, you can use the `az aks show` command in the Azure CLI.

Azure Disks can only be used by a single pod. If you need to access storage across multiple pods, you should use Azure Files instead. When you use Azure Files, you'll be able to access your storage using an SMB path that any of your pods can use. Whether you use Azure Disks or Azure Files, if you create your storage as part of your pod, that storage will not exist when the pod is deleted. This is more impactful than you might think at first. Consider a situation where AKS determines your pod needs to be rescheduled on another host, a situation that's almost certain as part of orchestration. In such a situation, the data you're storing will not persist. If you require that the lifecycle of your stored data not be tied to the lifecycle of the pod, you should use *persistent volumes* instead.

Persistent volumes can also use Azure Files or Azure Disks, and they can either be created by the AKS cluster administrator or by the Kubernetes API. Unlike non-persistent volumes created along with a pod, the persistent volume exists within the cluster, but outside of the pod. Kubernetes connects the persistent volume to the pod using a *persistent volume claim*. The persistent volume claim contains information about what type, tier, and size of storage is required, and it's used when creating the Azure Files share or the Azure disk.

Configure scaling for AKS

Applications can demand varying workloads. In order to keep up with demand, you may need to add additional AKS nodes, increase resources of existing nodes, or create more instances of your application.

AKS provides the ability to manually scale or to automatically scale, and you can scale at the cluster level when you need to scale nodes and at the pod level when you need to scale pods. You can even incorporate ACI into your AKS cluster in order to handle situations where you need to add additional nodes quickly.

To manually scale your pods, you can use Kubectl, a command-line tool provided by Kubernetes. You can access Kubectl from within the Azure CLI. Once you've opened the CLI and connected to your Azure subscription, run the following command to install Kubectl.

```
az aks install-cli
```

You'll then need to authenticate to your cluster by using the following command. Note that you specify the resource group and name of your AKS cluster in this command.

[Click here to view code image](#)

```
az aks get-credentials --resource-group aks_rg --name aks_cluster
```

Once you've done that, you can use Kubectl to scale your pod. The following command increases the number of pods in my `az104` deployment to 3.

[Click here to view code image](#)

```
kubectl scale --replicas=3 deployment/az104
```

To manually scale your cluster and add additional nodes, you can use the Azure CLI or PowerShell. The following CLI command scales my cluster to three nodes.

[Click here to view code image](#)

```
az aks scale --resource-group aks_rg --name aks_cluster --node-count 3
```

You can perform the same operation with PowerShell using the following

command.

[Click here to view code image](#)

```
Get-AzAksCluter -ResourceGroupName aks_rg -Name aks_cluster | Set-AzAksCluster  
-NodeCount 3
```

Manually scaling can be convenient, but it's much more likely that you'll want to scale based on the dynamic needs of your deployment. Kubernetes provides two autoscaler components to make it easy to configure auto-scaling; the horizontal pod autoscaler (HPA) and the cluster autoscaler.

To use the HPA to scale your pods, use Kubectl to run the autoscale command on your deployment. The following example sets the az104 deployment to autoscale when CPU usage exceeds 60 percent of the CPU that it was configured for when it was deployed. The HPA will scale to a maximum of five pods and a minimum of two pods.

[Click here to view code image](#)

```
kubectl autoscale deployment az104 --cpu-percent=60 --max=5 --min=2
```

To use the cluster autoscaler to autoscale the number of nodes in your cluster, you can use the Azure CLI. The following command configures my AKS cluster for a maximum of five nodes and a minimum of one.

[Click here to view code image](#)

```
az aks update --resource-group aks_rg --name aks_cluster --enable-cluster-autoscaler  
--max-count 5 --min-count 1
```

Note that you can also include these settings when you create your cluster.



Exam Tip

Under the hood, AKS uses Virtual Machine Scale Sets (VMSS) to implement cluster autoscaling. However, you should never attempt to interact with the VMSS directly.

Configure network connections for AKS

When creating an AKS cluster, you have two options for networking: kubenet and Azure Container Networking Interface (CNI). By default, AKS will use kubenet networking, also called *basic* networking. However, you can specify to use CNI (or *advanced*) networking if desired.

When you use kubenet networking, each node in the cluster gets an IP address from the VNet subnet where the cluster is deployed. However, each pod within the cluster gets an internal IP address from an address space explicitly set aside for the pods. When you use CNI networking, both the nodes and the pods receive an IP address from the subnet.

Kubenet does reduce the number of IP addresses you need for your cluster, but because all pods are using an IP address internal to the cluster, network address translation (NAT) is required in order for the pods to establish a network connection to other Azure resources. It also means that other VMs in Azure or on-premises can't directly establish communication with those pods.

Whether you're using kubenet or CNI, there are challenges with networking in Kubernetes. Kubernetes is designed to orchestrate pods, and that means that it spins them up and tears them down as needed. Because of that, the IP addresses for your pods are constantly changing. For that reason, Kubernetes implements the concept of a *service* that sits between incoming network traffic and one or more identical pods. The service gets an IP address from a specific IP address pool set aside for services, and because the service is always running, it's not affected by pod lifecycle.

When network traffic needs to reach a particular pod (for example, a pod running a website that needs to process an HTTP request), the traffic is received by the service. The service will then balance the traffic to the pods using a round robin algorithm.

There are multiple service types.

- **Cluster IP** Provides an internal IP address that can only be used within the AKS cluster.
- **NodePort** Provides a port mapping on the node, allowing network traffic to reach the node using the specified port. (Note that a different port can

then be used from the service to the actual pod.)

- **LoadBalancer** Provides an Azure Load Balancer and an external IP address to allow access to the node as per load balancing rules that are created. (Internal load balancers can be created to restrict access from the Internet.)
- **ExternalName** Provides a DNS entry for AKS nodes.

When using a LoadBalancer service type, the load balancer distributes traffic based on the originating port. However, there may be situations where you need more control over where traffic is routed. For example, suppose your cluster is hosting a web service, and you need traffic to be distributed to specific pods based on the incoming URL. A LoadBalancer service is unable to implement rules to deal with that, so in a case such as this, you can use an *ingress controller* to handle the traffic.

You can use NGINX for an ingress controller in AKS, but you can also use other methods such as the AKS HTTP application routing feature or the Application Gateway Ingress Controller (AGIC) add-on. AGIC uses Application Gateway in Azure to make services in your cluster available over the Internet.

More Info

Network security in an AKS cluster is handled using NSGs and network policy. Azure creates NSG rules for you as you create resources. Network policy is a feature in Kubernetes that enables you to control network traffic between pods.

Upgrade an AKS cluster

Kubernetes is constantly being updated, and whether it's because you want to access new features or because you want to ensure you have all the latest bug fixes, you are likely going to need to upgrade your AKS cluster at some point.

The first step in upgrading a cluster is to determine whether updates are even available. You can do this easily using the Azure CLI with the following command.

[Click here to view code image](#)

```
az aks get-upgrades --resource-group aks_rg --name aks_cluster
```

The output of this command will be in JSON, and the `upgrades` section will show you any upgrades that are available. You can then upgrade your cluster using the following command.

[Click here to view code image](#)

```
az aks upgrade --resource-group aks_rg --name aks_cluster --  
kubernetes-version 1.21.1
```

AKS doesn't upgrade all the nodes in the cluster at one time because that can cause problems with the applications running in the cluster. Instead, Kubernetes upgrades one node at a time. It first stops any pods from being scheduled on the node it's about to upgrade, and any pods that are currently running on that node are scheduled for other nodes. A new node is then created using the version of Kubernetes you've specified to upgrade to, and only when that node is ready and added to the cluster will new pods be scheduled to run on it. Once that's done, Kubernetes deletes the node running the older version and begins the upgrade process on the next node in the cluster. This continues until all nodes are upgraded.



Exam Tip

When upgrading an AKS cluster, you cannot skip minor versions. For example, you can't upgrade from version 1.19.3 to version 1.21.1. You would first have to upgrade to a 1.20 build and then upgrade again to version 1.21.1.

Skill 3.5: Create and configure Azure App Service

Azure App Service is a PaaS offering that makes it easy to host a web app in the cloud. However, App Service isn't just for web apps. Any application that is designed to process HTTP requests can benefit from App Service. This makes

App Service the ideal hosting platform for web apps, apps that expose REST APIs, and much more.

App Service consists of a front-end load balancer that uses a round robin algorithm to distribute requests to web servers. These web servers are called *workers*, and they are responsible for processing HTTP requests. You can run your app on Linux workers or Windows workers, and you can also choose the VM size used by your app. App Service offers both shared workers (shared with other App Service users) or dedicated workers that host only your apps. These configuration choices are part of an *App Service plan* that is used to host your apps.

This section covers how to:

- [Create an App Service Plan](#)
- [Configure scaling settings in an App Service plan](#)
- [Create an App Service](#)
- [Secure an App Service](#)
- [Configure custom domain names](#)
- [Configure backup for an App Service](#)
- [Configure networking settings](#)
- [Configure deployment settings](#)

Create an App Service Plan

Before you create a web app in App Service, you'll need to create your App Service plan. You can explicitly create an App Service plan, or you can opt for Azure to create it for you when you create your web app.

To create an App Service plan in the Azure portal, select **App Service Plan** in the Azure Marketplace. Once you specify the resource group for the App Service plan, you can choose your operating system, region, and the pricing tier (VM SKU and size) for the App Service plan, as shown in [Figure 3-49](#).

Home > Create a resource > Marketplace >

Create App Service Plan ...

Basics Tags Review + create

App Service plans give you the flexibility to allocate specific apps to a given set of resources and further optimize your Azure resource utilization. This way, if you want to save money on your testing environment you can share a plan across multiple apps. [Learn more](#)

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource Group * ⓘ

▼

[Create new](#)

App Service Plan details

Name *

✓

Operating System *

Linux Windows

Region *

▼

Pricing Tier

App Service plan pricing tier determines the location, features, cost and compute resources associated with your app.

[Learn more](#)

Sku and size *

Premium V2 P1v2

210 total ACU, 3.5 GB memory

[Change size](#)

[Review + create](#)

< Previous

Next : Tags >

FIGURE 3-49 Creating an App Service plan

Configure scaling settings in an App Service plan

One of the greatest benefits of App Service is the flexibility it provides for scaling applications. You can scale vertically (scale to a more or less powerful VM) or horizontally (add and remove workers) easily and quickly. Options for scaling are available on the menu when you open your App Service plan in the portal.

More Info Scaling is always the same

You'll also see scaling options in the menu after opening a web app in the portal. However, when you select these menu options, you're actually scaling the App Service plan the web app is running in. In other words, scaling is exactly the same whether you do it from within the App Service plan or the web app running in the App Service plan.

To scale an App Service plan vertically, click the **Scale up (App Service Plan)** menu option in the portal, as shown in [Figure 3-50](#). You can select either **Dev/Test**, **Production**, or **Isolated**, depending on your specific needs and Azure will show you recommended pricing tiers, along with other pricing tiers you can select. After you select the desired pricing tier, click **Apply** and your App Service plan will be immediately scaled to the selected tier.

MyAppServicePlan | Scale up (App Service plan)

Linux App

Search (Cmd+/)

Dev / Test For less demanding workloads

Production For most production workloads

Isolated Advanced networking and scale

Recommended pricing tiers

P1V2	210 total ACU 3.5 GB memory Dv2-Series compute equivalent 81.03 USD/Month (Estimated)	P2V2	420 total ACU 7 GB memory Dv2-Series compute equivalent 161.33 USD/Month (Estimated)	P3V2	840 total ACU 14 GB memory Dv2-Series compute equivalent 322.66 USD/Month (Estimated)
P1V3	195 minimum ACU/vCPU 8 GB memory 2 vCPU 93.44 USD/Month (Estimated)	P2V3	195 minimum ACU/vCPU 16 GB memory 4 vCPU 186.88 USD/Month (Estimated)	P3V3	195 minimum ACU/vCPU 32 GB memory 8 vCPU 373.76 USD/Month (Estimated)

[See only recommended options](#)

Additional pricing tiers

S1	100 total ACU 1.75 GB memory A-Series compute equivalent 69.35 USD/Month (Estimated)	S2	200 total ACU 3.5 GB memory A-Series compute equivalent 138.70 USD/Month (Estimated)	S3	400 total ACU 7 GB memory A-Series compute equivalent 277.40 USD/Month (Estimated)
----	---	----	---	----	---

Included features

Every app hosted on this App Service plan will have access to these features:

- Custom domains / SSL
- Auto scale

Included hardware

Every instance of your App Service plan will include the following hardware configuration:

- Azure Compute Units (ACU)
- Memory

Apply

Support + troubleshooting

FIGURE 3-50 Scaling up an App Service plan

When App Service scales your app to a new tier, it will take steps to ensure your application remains available during the scaling process. It will first allocate one or more VMs of the selected size and it will then copy any apps you have in the App Service plan to the new VMs. While this happens, the App Service front-end will continue to send any incoming requests to your current VMs. Only when your apps are on the new VMs and ready to process requests will the front-end start sending new requests to the new VMs. That might seem like it would take a bit of time, but in fact, App Service can perform all of these steps quickly.

To scale an App Service plan horizontally, click the **Scale Out (App Service Plan)** menu option in the portal. You can then choose the number of instances (an instance is a VM) that you want to run. In [Figure 3-51](#), an App Service plan is being scaled to three instances. To complete the scaling operation, click the **Save** button at the top of the screen.

Home > Microsoft.Web-ASP-Portal-2fe70ace-9218 > MyAppServicePlan

MyAppServicePlan | Scale out (App Service plan)

Linux App

Search (Cmd+ /) Save Discard Refresh Logs Feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Events (preview)

Configure Run history JSON Notify Diagnostic settings

Autoscale is a built-in feature that helps applications perform their best when demand changes. You can choose to scale your resource manually to a specific instance count, or via a custom Autoscale policy that scales based on metric(s) thresholds, or schedule instance count which scales during designated time windows. Autoscale enables your resource to be performant and cost effective by adding and removing instances based on demand. [Learn more about Azure Autoscale](#) or [view the how-to video](#).

Choose how to scale your resource

Manual scale

Maintain a fixed instance count

Custom autoscale

Scale on any schedule, based on any metrics

Manual scale

Override condition

Instance count 3

FIGURE 3-51 Scaling out an App Service plan



Exam Tip

Remember that when an App Service plan runs on more than one instance, the front-end load balancer will use a round robin algorithm to load balance between all instances.

In addition to manually scaling an App Service plan horizontally, you can also configure automatic scaling. Automatic scaling uses the autoscale service in Azure to automatically scale your App Service plan based upon specific metrics or specific dates and times.

To configure autoscale, click the **Custom Autoscale** option in the **Scale Out** screen shown previously in [Figure 3-51](#). After selecting the autoscale option, you can choose to scale based on a metric or scale to a specific instance count. These options are available in the **Default** scale condition, as shown in [Figure 3-52](#). The **Default** scale condition applies only when no other scale conditions are matched.

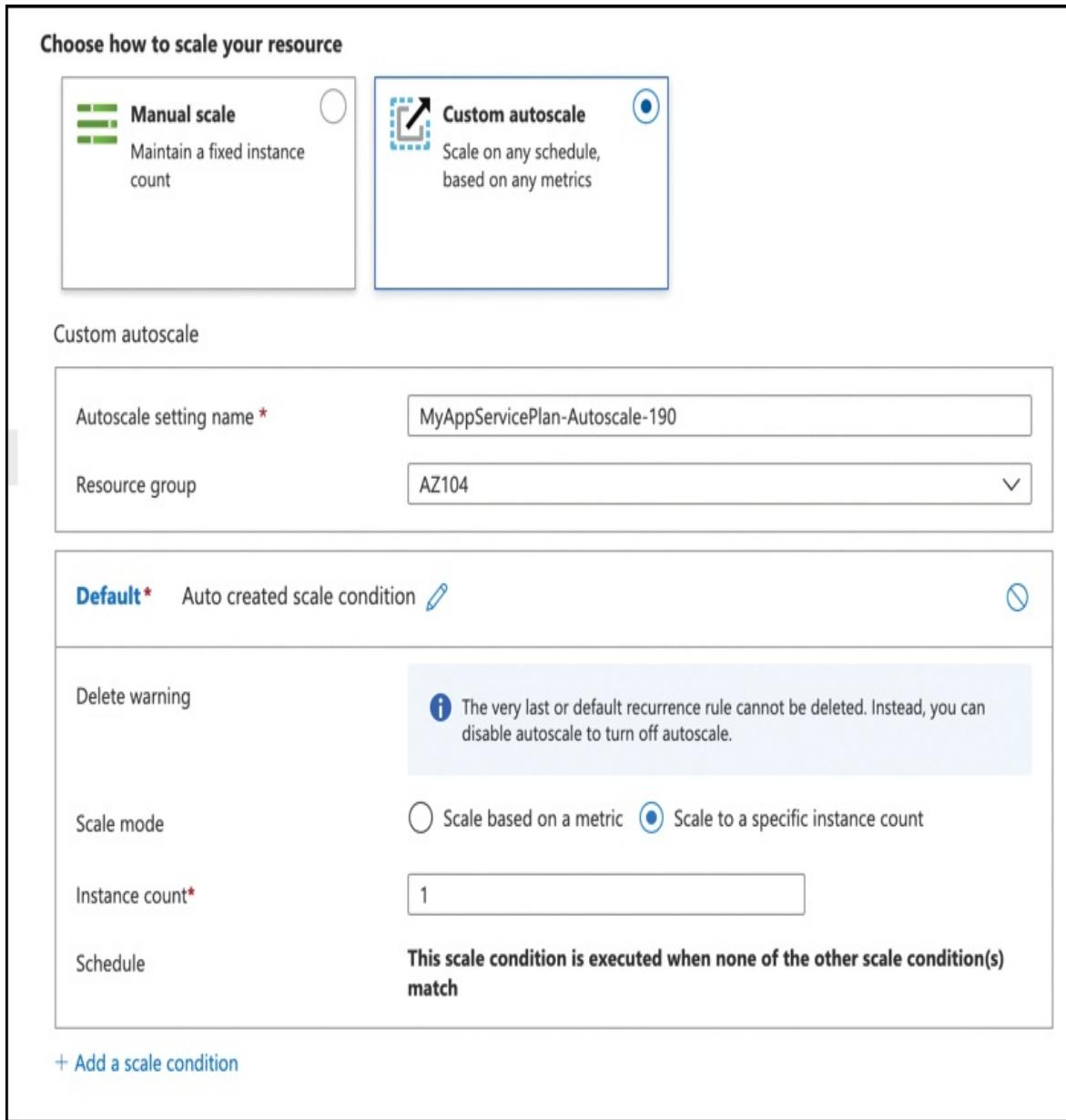


FIGURE 3-52 Configuring autoscale for an App Service plan

To add a new scale condition, click the **Add A Scale Condition** link at the bottom of the screen. Just as with the default rule, you can specify to scale based on a metric or to scale to a specific instance count. When you choose to scale based on a metric, you must configure a rule for the metric you want to use. You do that by clicking the **Add Rule** link shown in Figure 3-53.

Default* Auto created scale condition [Edit](#) [Delete](#)

Scale mode Scale based on a metric Scale to a specific instance count

Instance count*

Schedule **This scale condition is executed when none of the other scale condition(s) match**

Auto created scale condition 1 [Edit](#) [Delete](#)

Scale mode Scale based on a metric Scale to a specific instance count

Rules
No metric rules defined; click [Add a rule](#) to scale out and scale in your instances based on rules.
For example: 'Add a rule that increases instance count by 1 when CPU percentage is above 70%'. If you save the setting without any rules defined, no scaling will occur. [+ Add a rule](#)

Instance limits
 Minimum [?](#) ✓ Maximum [?](#) ✓

Default [?](#) ✓

Schedule Specify start/end dates Repeat specific days

Timezone [▼](#)

Start date [Select](#)

End date [Select](#)

[+ Add a scale condition](#)

FIGURE 3-53 Configuring an autoscale condition

In [Figure 3-54](#), a new scale rule is being created based on CPU utilization. An historical graph is displayed for the selected metric to make it easier to see the pattern for the metric. This particular rule is configured so that if CPU usage is greater than 70 percent for 10 minutes, autoscale will increase the number of

instances by 1. After the App Service plan is scaled, autoscale will wait for 5 minutes (the cool down time) before it will scale the app again. After those 5 minutes, if CPU usage is still high enough to activate the rule, autoscale will increase the instance count by 1 again until it reaches the maximum number of instances, as shown previously in [Figure 3-53](#).

Scale rule

X

Time aggregation * ⓘ

Average

Metric namespace *

App Service plans standard metrics

Metric name

CPU Percentage

1 minute time grain

Dimension Name

Operator

Dimension Values

Add

Instance

=

All values

▼



If you select multiple values for a dimension, autoscale will aggregate the metric across the selected values, not evaluate the metric for each values individually.



CpuPercentage (Average)

3.45 %

Enable metric divide by instance count ⓘ

Operator *

Metric threshold to trigger scale action * ⓘ

Greater than

▼

70

%

Duration (in minutes) *

10

Time grain (in mins) ⓘ

Time grain statistic * ⓘ

1

Average

▼

Action

Operation *

Cool down (minutes) * ⓘ

Increase count by

▼

5

Instance count *

1



Add

FIGURE 3-54 Configuring an autoscale rule



Exam Tip

When you configure an autoscale rule to scale out for a specific metric, you should also create another rule to scale in when that metric drops below your desired threshold.

Create an App Service

So far, you've seen how to create an App Service plan. Now let's look at how you can create an App Service app that runs in that App Service plan. To create a new web app, search for Web App in the Azure Marketplace. When creating your web app, you'll specify a name for the app, as shown in [Figure 3-55](#).

Create Web App

Basics Deployment (Preview) Monitoring Tags Review + create

App Service Web Apps lets you quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. Meet rigorous performance, scalability, security and compliance requirements while using a fully managed platform to perform infrastructure maintenance. [Learn more](#)

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource Group * ⓘ

 ▼

[Create new](#)

Instance Details

Name *

 ✓

.azurewebsites.net

Publish *

Code Docker Container

Runtime stack *

 ▼

Operating System *

Linux Windows

Region *

 ▼

[ⓘ Not finding your App Service Plan? Try a different region.](#)

App Service Plan

App Service plan pricing tier determines the location, features, cost and compute resources associated with your app.

[Learn more](#)

Linux Plan (Central US) * ⓘ

 ▼

[Create new](#)

Sku and size *

Premium V2 P1v2

210 total ACU, 3.5 GB memory

[Review + create](#)

[< Previous](#)

[Next : Deployment \(Preview\) >](#)

FIGURE 3-55 Creating a web app

You have the option of creating your app using your own code or to run a Docker container in your App Service plan. If you choose **Code**, you can select from a wide range of runtime stacks, including .NET, Java, Node, PHP, Ruby, and Python. In [Figure 3-55](#), I've selected my existing App Service plan for this web app, but you can also create a new App Service plan by clicking **Create New**. Once everything is configured the way you want it, click **Review + Create** to create the web app. Once the web app has been created, you can access it by browsing to https://web_app_name.azurewebsites.net.



Exam Tip

If you don't see an existing App Service plan when creating a new web app, make sure you have selected the OS that matches the App Service plan's OS. You also need to ensure the region you select is the region where the App Service plan is deployed.

Secure an App Service

As is typical with a PaaS service, App Service makes it simple to implement security for your web app. You can secure your app with Azure Active Directory, and you can also easily implement security using Facebook, Google, and Twitter so that users can authenticate to your app using their existing logins.on-premises

To configure authentication, click the Authentication menu item after opening your web app in the Azure portal. When you first do this, you'll need to add an identity provider. In [Figure 3-56](#), Facebook has been selected as the identity provider. The App ID and App Secret are from the Facebook developer's site. The **Authentication** setting has been set to **Require Authentication**, thereby preventing users from browsing my web app without logging in. Directly under that setting, unauthenticated requests are configured to redirect to Facebook so that users can log in. Once the settings are configured,

click **Add** to add the provider.

Home > Microsoft.Web-WebApp-Portal-c651902f-8b37 > MyAz104App >

Add an identity provider

Basics Scopes

Identity provider * Facebook

App registration

An app registration associates your identity provider with your app. Enter the app registration information here, or go to your provider to create a new one. [Learn more](#)

App ID * az104FB

App secret * 771829875092802938409238493

App Service authentication settings

Requiring authentication ensures all users of your app will need to authenticate. If you allow unauthenticated requests, you'll need your own code for specific authentication requirements. [Learn more](#)

Authentication * Require authentication Allow unauthenticated access

Unauthenticated requests * HTTP 302 Found redirect: recommended for websites HTTP 401 Unauthorized: recommended for APIs HTTP 403 Forbidden

Redirect to Facebook

Token store

Add < Previous Next: Scopes >

FIGURE 3-56 Configuring an identity provider to secure a web app



Exam Tip

App Service uses OAUTH authentication when configuring a third-party identity provider. Secrets that you provide to configure the provider are securely stored in Azure Key Vault.

You can configure multiple identity providers for your app. However, doing so will require you to create a page that presents users with a list of authentication providers and links to sign in. For more information on doing this, see <https://docs.microsoft.com/en-us/azure/app-service/app-service-authentication-how-to#use-multiple-sign-in-providers>.

Configure custom domain names

As I mentioned earlier, you can browse to your app in App Service using `https://app_name.azurewebsites.net`, but it's likely you'll want to use your own custom domain name. App Service makes using a custom domain easy.

In order to map an existing custom domain to your web app, you'll need to go through a series of steps, some of which are carried out at your domain registrar's website. Therefore, some of the steps may differ slightly depending on your provider.

Open your web app in the Azure portal and click Custom Domains in the menu. Once you do, you'll see the IP address and custom domain verification ID as shown in [Figure 3-57](#). You'll use these to configure the DNS records at your domain registrar.

Home > Microsoft.Web-WebApp-Portal-c651902f-8b37 > MyAz104App

MyAz104App | Custom domains

App Service

Search (Cmd+/) Refresh Troubleshoot FAQs

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Security Events (preview)

Deployment Quickstart Deployment slots Deployment Center

Settings Configuration Authentication Authentication (classic) Application Insights Identity Backups Custom domains TLS/SSL settings Networking

Custom Domains

Configure and manage custom domains assigned to your app [Learn more](#)

IP address: 20.40. [REDACTED]

Custom Domain Verification ID: 5E6EF8320DA477F6BF646C79CB2F9B1744 [REDACTED]

HTTPS Only: Off

Add custom domain

Status Filter: All (1) Not Secure (0) Secure (1)

SSL STATE	ASSIGNED CUSTOM DOMAINS	SSL Binding
Secure	myaz104app.azurewebsites.net	

App Service Domains

Manage domains for your Azure services with auto-renew and privacy protection. [Learn more](#)

Buy App Service domain

DOMAINS	EXPIRES	STATUS
No data found		

FIGURE 3-57 Configuring a custom domain in App Service

You'll need to create several DNS records at your domain registrar.

- **A record** To map your root domain (*mydomain.com*), create an A record that maps a host of @ to the IP address shown in the Azure portal.
- **txt record** To verify ownership of your root domain, create a TXT record named `asuid` with a value of the custom domain verification ID shown in the Azure portal.
- **CNAME record** To map a subdomain (www.mydomain.com), create a CNAME record that maps the desired subdomain (such as `www`) to the `azurewebsites.net` URL for your app.
- **txt record** To verify ownership of your domain, create a TXT record named `asuid.subdomain` (such as `asuid.www`) with a value of the custom domain verification ID shown in the Azure portal.

Once you've created these records, you can proceed to add your custom domain in the Azure portal by clicking the **Add Custom Domain** link shown previously in [Figure 3-57](#). After clicking the link, enter the custom domain you want to add and click the **Validate** button shown in [Figure 3-58](#).

Add custom domain

X

myaz104app

Custom domain *

www.mycustomdomainname.com



Validate

Hostname record type

CNAME (www.example.com or any subdomain)



CNAME configuration

A CNAME record is used to specify that a domain name is an alias for another domain. In your scenario, that would be mapping www.mycustomdomainname.com to custom domain verification id below. [Learn More](#)

Custom Domain Verification ID: ⓘ

5E6EF8320DA477F6BF646C79CB2F9B1744



CNAME

myaz104app.azurewebsites.net

Add custom domain



DNS propagation

Please be aware that depending on your DNS provider it can take up to 48 hours for the DNS entry changes to propagate. You can verify that the DNS propagation is working as expected by using <https://digwebinterface.com/>. [Learn more](#)



Hostname availability



Domain ownership

FIGURE 3-58 Adding a custom domain

As you can see in [Figure 3-58](#), **Domain Ownership** shows an error icon. This is because the necessary DNS records have not yet been configured for the custom domain specified. Once those records have been created and propagated, I can click **Validate** again to retry. Once domain ownership is verified by the DNS records, the **Add Custom Domain** button will become enabled.



Exam Tip

If you don't already own a custom domain name, you can use App Service domains to buy and configure a custom domain name. App Service domains are fully managed in Azure, and they're the easiest way to configure a custom domain.

To create an App Service domain, search for App Service Domains in the Azure portal or click the link on the **Custom Domains** screen for your web app.

Configure backup for an App Service

App Service provides easy backup and restore of your apps. These backups can be created manually, or they can be scheduled on a regular basis. Backups can be retained for an indefinite amount of time.

When you back up your app, App Service can back up not only the app's content and configuration, but it can also back up SQL Database, Azure Database for MySQL, Azure Database for PostgreSQL, and MySQL in-app databases.

More Info MySQL in-app

MySQL in-app is a feature of App Service that provides a file-based

MySQL database for use in your app.

App Service backups are stored in Azure storage, and each backup is a complete copy of the app. Backups are not incremental.

You can manually back up your app by clicking on the Backups menu option and clicking on Backup as shown in [Figure 3-59](#). Note that if you haven't already configured the storage account for your backups, you'll first need to do that.

Once you've created a backup, you can restore the backup by clicking **Restore**. You can choose to overwrite your existing app or restore the backup to a new app. If you'd prefer, you can create a scheduled backup so that App Service backs up your app automatically at configured times. To configure a backup schedule, click the banner shown previously in [Figure 3-59](#). When you do, you'll see the backup configuration screen where you can click **Scheduled Backup** to define a backup schedule as shown in [Figure 3-60](#).

Home > Microsoft.Web-WebApp-Portal-c651902f-8b37 > MyAz104App

MyAz104App | Backups

App Service

Search (Cmd+ /) Configure Refresh Reset FAQs

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Security Events (preview)

Deployment Quickstart Deployment slots Deployment Center

Settings Configuration Authentication Authentication (classic) Application Insights Identity Backups Custom domains

Snapshot (Preview)

Snapshots automatically create periodic restore points of your app when hosted in a Premium App Service plan. [Learn more](#)

Restore

Backup

Configure backup to create restorable archive copies of your apps content, configuration and database. [Learn more](#)

⚠ Backup configured, backup schedule is not configured, configure scheduled backup to automatically take backups.

↓ Backup **Restore**

Status	Backup time	Size (MB)
No backup history is available		

Figure 3-59 Backing up a web app

Backup Configuration

X



Backup Schedule

Configure the schedule for your app backup.

Scheduled backup

On Off

Backup Every *

1

Days Hours

Start backup schedule from *

06/06/2021



10:26:07 AM

(UTC-06:00) Central Time (US & Canada)



Retention (Days) * ⓘ

30

Keep at least one backup

No Yes



Backup Database

Select the databases to include with your backup. The backup database list is based on the app's configured connection strings. Note: The maximum size of content + database backup cannot exceed 10GB. If your database is large and growing, use Azure Backup for database backup instead.

Include in Backup

Connection String Name

Database Type

No supported connection strings of type SQL Database or MySQL found configured in app.

Save

Discard

FIGURE 3-60 Configuring a backup schedule

Once you configure your desired settings, click **Save** to save the schedule.



Exam Tip

If you want to keep your scheduled backup for an indefinite amount of time, set the **Retention Days** to 0.

App Service backups cannot exceed 10 GB, and that includes the app's content and any databases. It's also important to remember that you have to be running your App Service plan in the Standard tier or higher. Backup is not available for lower tiers.

Configure networking settings

There are several features in App Service that enable certain networking scenarios. These features include:

- **VNet integration** Enables outgoing communication from your app into your Azure virtual network.
- **Private Endpoint connections** Enables connectivity to your app from private endpoints using Azure Private Link.
- **Hybrid connections** Enables outgoing communication from your app to an endpoint using a TCP connection. The host can be located practically anywhere.

It's relatively common for an app in App Service to need access to other Azure resources running inside of a VNet. By configuring VNet integration with your web app, you can easily connect your app to your existing VNet.

To configure VNet integration, click the **Networking** menu for your web app

in the portal and then click the link to configure VNet integration. Click **Add VNet** and select your existing VNet and subnet as shown in [Figure 3-61](#). Click **OK** to add the VNet.

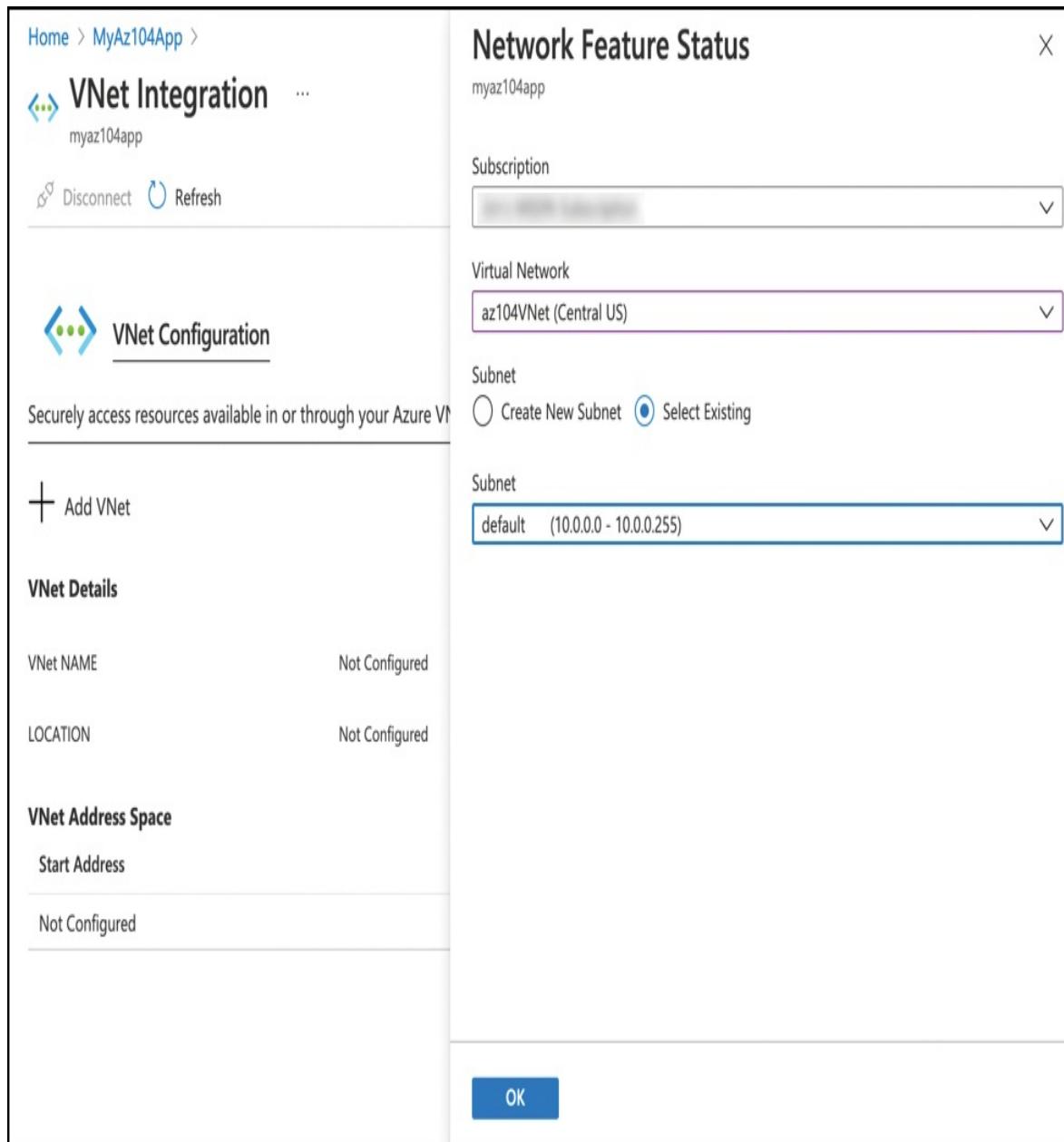


FIGURE 3-61 Configuring VNet integration with a web app

Once you've added your VNet, your app can then make outgoing connections to the VNet.

If you want to provide a secure connection to your app, you can configure private endpoints. Private endpoints enable you to connect securely to resources running in an Azure VNet or on-premises resources using either VPN or ExpressRoute.

To configure private endpoint connections, click the link to configure private endpoint connections in the Networking screen of your web app. You can then click **Add** to add a new private endpoint, as shown in [Figure 3-62](#).

The screenshot shows the 'Add Private Endpoint' dialog box overlaid on the 'Private Endpoint connections' blade of the Azure portal. The dialog box contains the following fields:

- Name ***: 104PE
- Subscription ***: A dropdown menu showing a single option.
- Virtual network ***: az104VNet
- Subnet ***: PESubnet

Below the form, there is a note: **i** If you have a network security group (NSG) enabled for the subnet above, it will be disabled for private endpoints on this subnet only. Other resources on the subnet will still have NSG enforcement.

There is also a section for integrating with a private DNS zone:

- Integrate with private DNS zone**: A radio button labeled **Yes** is selected.
- i** Your private endpoint will be integrated with the private DNS zone 'privatelink.azurewebsites.net' in the resource group of the selected subnet. If the private DNS zone does not exist, it will be created automatically. [Learn more](#)

At the bottom right of the dialog box is a blue **OK** button.

FIGURE 3-62 Configuring a private endpoint

After you enter the endpoint name, select your VNet and subnet before clicking **OK** to add the endpoint. After doing that, a network device on your VNet can access your web app using Private Link, and that connection will be sent over the Azure backbone infrastructure rather than the public Internet.

Hybrid connections enable your web app to access a resource on another network without complicated configuration. They're often used to access an on-premises resource from a web app. Hybrid connections rely on the installation of the Hybrid Connection Manager (HCM) on the host you're attempting to access. The HCM handles the communication between the remote host and the web app, and because this communication happens over standard web ports, a hybrid connection usually doesn't require any ports to be opened on a firewall.

To configure a hybrid connection, click the link to configure hybrid connections in the Networking screen of your web app. As shown in [Figure 3-63](#), from this screen, you can download the HCM and you can also add a new hybrid connection.

Home > AZ104 > MyAz104App >

Hybrid connections

myaz104app

Refresh

Hybrid connections

App Service integration with hybrid connections enables your app to access a single TCP endpoint per hybrid connection. Here you can manage the new and classic hybrid connections used by your app. [Learn more](#)

App service plan (pricing tier): **MyAppServicePlan (PremiumV2)**

Location: Central US

Connections used: 0 / Connections quota 220

Download connection manager

Add hybrid connection ⓘ

Name	Status	Endpoint	Namespace
No results			

FIGURE 3-63 Hybrid connections with a web app

After clicking the **Add Hybrid Connection** link, click **Create New Hybrid Connection** to display the screen to create a hybrid connection as shown in Figure 3-64.

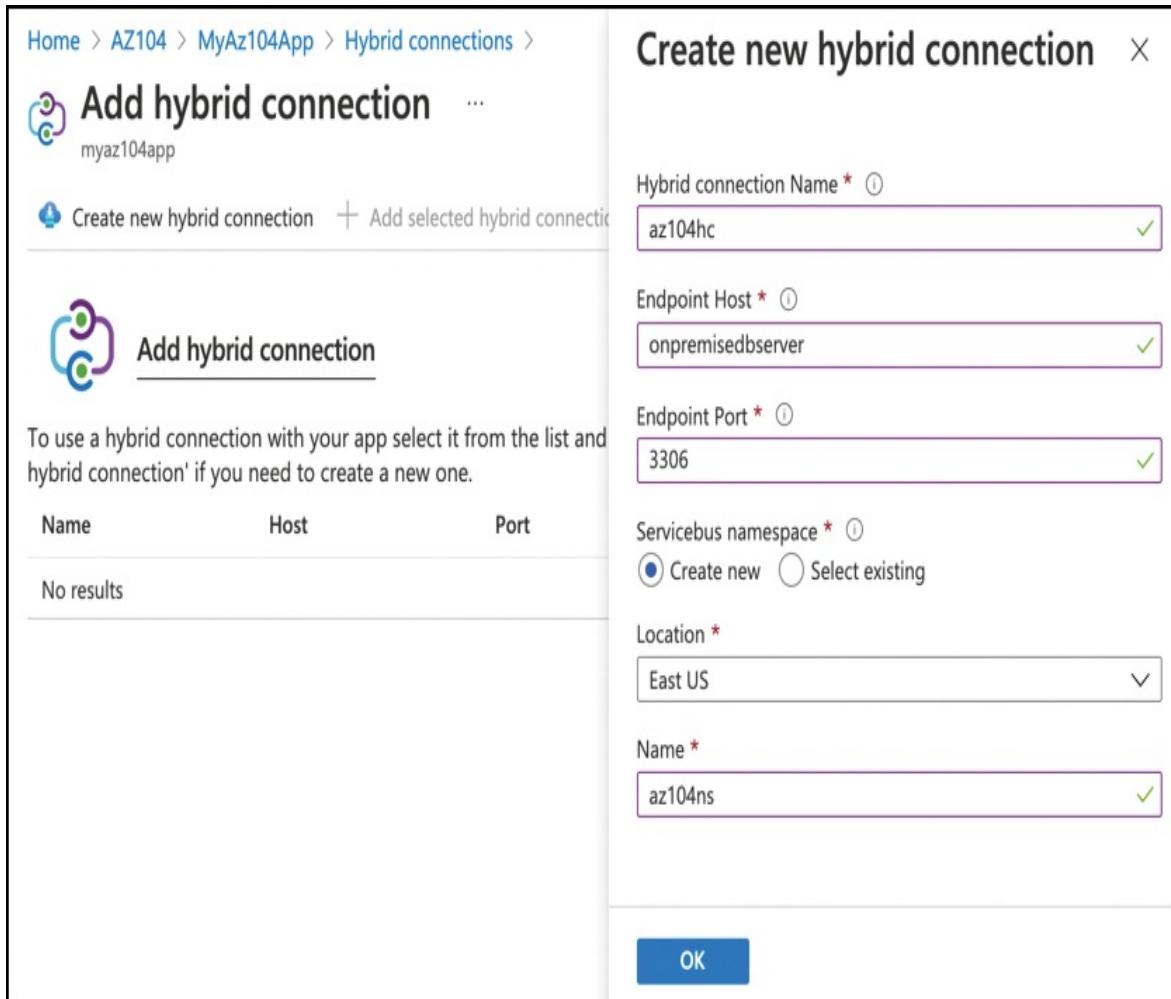


FIGURE 3-64 Adding a new hybrid connection

Give your hybrid connection a name that you can easily identify. The endpoint host you specify is usually going to be the NetBIOS name of the server you're connecting to. For example, in Figure 3-64, I'm connecting to an on-premises database server with a name of onpremisedbserver. The endpoint port is configured to 3306 because my database server is running MySQL, and that's the port that MySQL uses for incoming connections.

Note that a Service Bus namespace has also been configured. This is because hybrid connections use Service Bus for communication. It's important that your Service Bus namespace be in the same region as your web app in order to avoid latency.



Exam Tip

The HCM doesn't have to be installed on the host you specify for your hybrid connection. It does, however, need to be installed on a server that can successfully resolve the endpoint host. This means that you can use hybrid connections to connect to on-premises servers that aren't even connected to the Internet as long as they can connect to the server where the HCM is installed.

Configure deployment settings

App Service provides numerous options for deploying your app, but before we get into those options, we first need to discuss a feature called deployment slots.

Deployment slots allow you to create another app with its own hostname in your App Service plan. You can use a deployment slot to test a new version of an app, and once you're satisfied with the new version, you can easily swap the test deployment slot into production. You can even configure App Service to send a percentage of live traffic to a deployment slot for testing.

To create a new deployment slot, click the **Deployment Slots** menu option for your web app. Click **Add Slot** to add a new deployment slot, as shown in Figure 3-65.

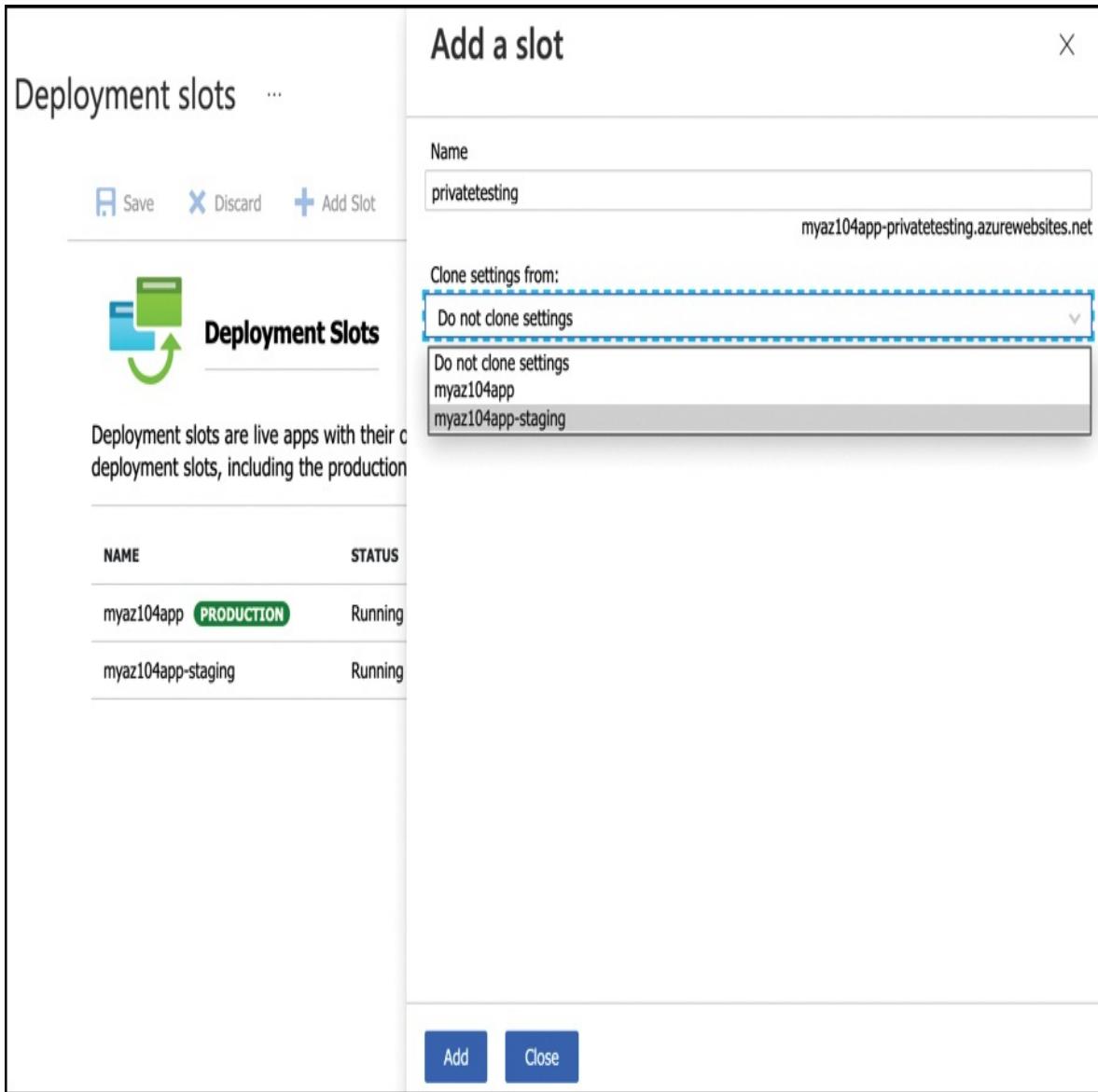
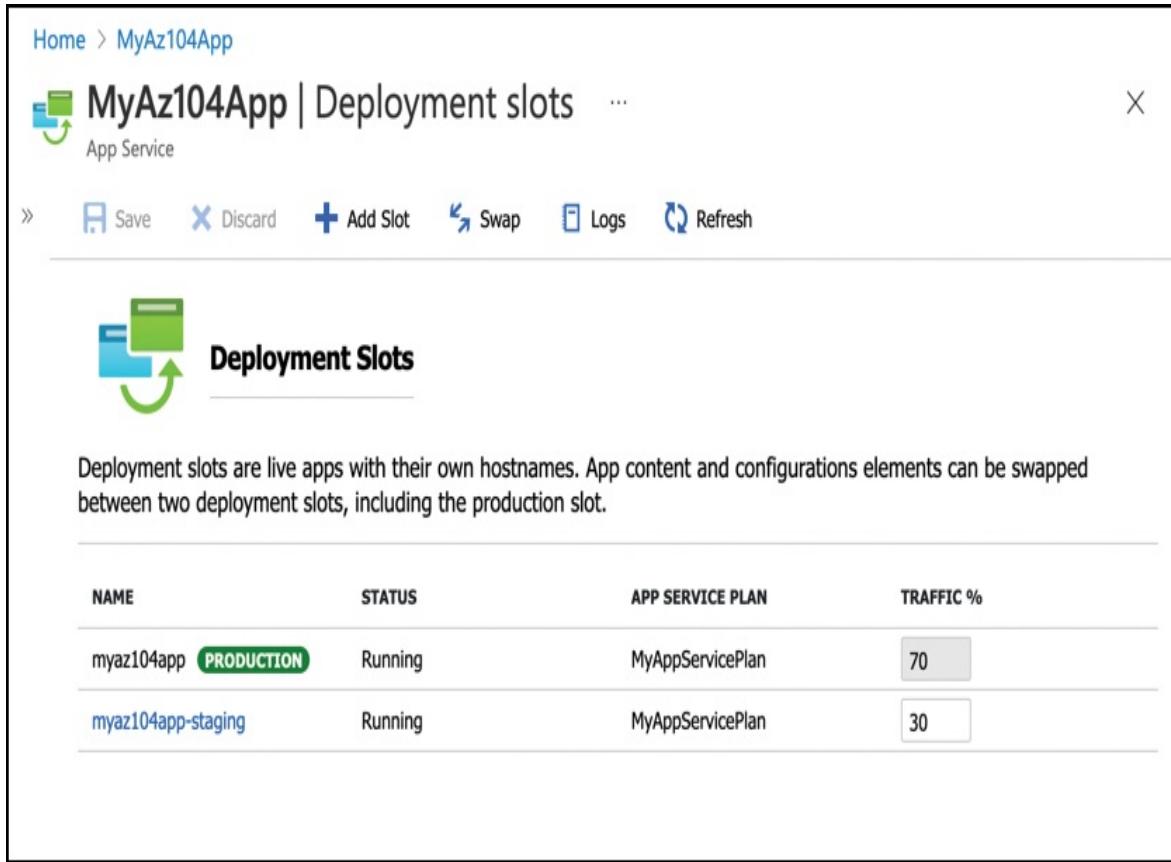


FIGURE 3-65 Adding a deployment slot

Enter the name you want to use for your deployment slot. Once you've added the slot, you can browse to it using this URL: https://app_name-slot_name.azurewebsites.net. You can also choose to clone settings from another deployment slot. Once you've entered the necessary information, click **Add** to add the slot and click **Close** to close the **Add A Slot** screen.

Once you've added your deployment slot, you can deploy a test version of your app to that slot. You can then easily configure a certain portion of live traffic to be distributed to that slot. In [Figure 3-66](#), 30 percent of my live traffic

is configured to go to the staging deployment slot.



The screenshot shows the Azure portal interface for managing deployment slots. At the top, there's a breadcrumb navigation 'Home > MyAz104App'. Below it, the title 'MyAz104App | Deployment slots' is displayed, along with a small green icon representing an app service. On the right side of the title bar are buttons for 'Save', 'Discard', 'Add Slot', 'Swap', 'Logs', and 'Refresh'. A large green circular icon with a white arrow pointing upwards and to the right is positioned above the 'Deployment Slots' section. The section is titled 'Deployment Slots' and contains a descriptive text: 'Deployment slots are live apps with their own hostnames. App content and configurations elements can be swapped between two deployment slots, including the production slot.' Below this text is a table showing two deployment slots:

NAME	STATUS	APP SERVICE PLAN	TRAFFIC %
myaz104app	PRODUCTION	MyAppServicePlan	70
myaz104app-staging	Running	MyAppServicePlan	30

FIGURE 3-66 Configuring deployment slot traffic

It's important to understand that each deployment slot is its own web app. If I click the staging slot shown in [Figure 3-66](#), it will open that web app in the Azure portal, and any changes I make apply only to the staging slot. This concept will be important to understand when we cover deployment options later in this section.

After testing the staging slot, I can easily move it into production by clicking the **Swap** button at the top of the screen. I can then select the source slot, the target slot, and review changes that will take place based on my settings, as shown in [Figure 3-67](#).

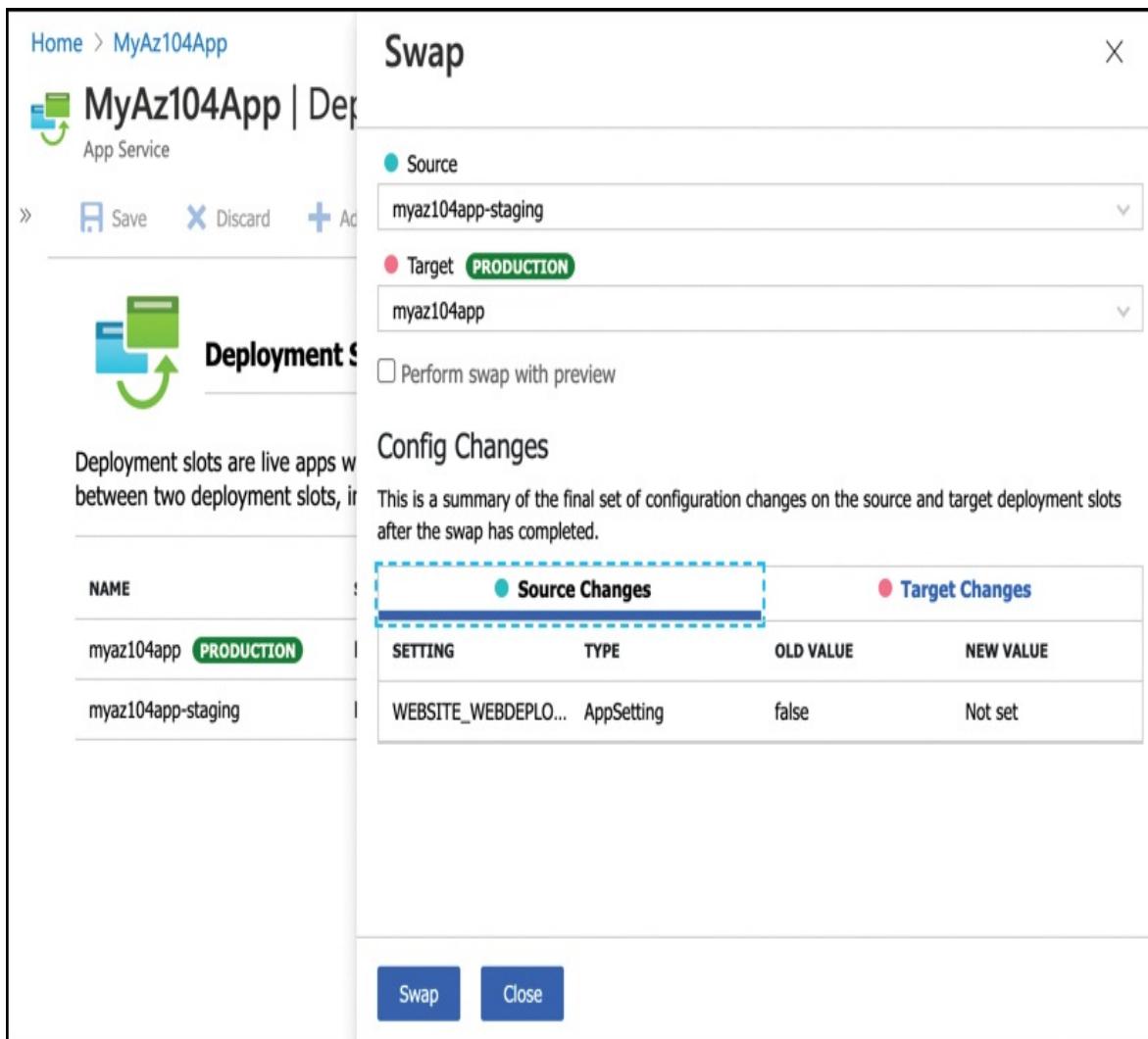


FIGURE 3-67 Swapping a slot

When I'm ready to complete the swap, I click the **Swap** button at the bottom of the screen, and App Service takes care of the rest.

Once you configure deployment slots, you'll typically always deploy new code into the staging slot. To configure deployment for that slot, click the staging slot in the **Deployment Slots** screen shown previously in [Figure 3-66](#). This will open the staging slot's web app in the Azure portal, as I mentioned previously.

After opening the staging slot, click **Deployment Center** in the menu to display the Deployment Center, as shown in [Figure 3-68](#).

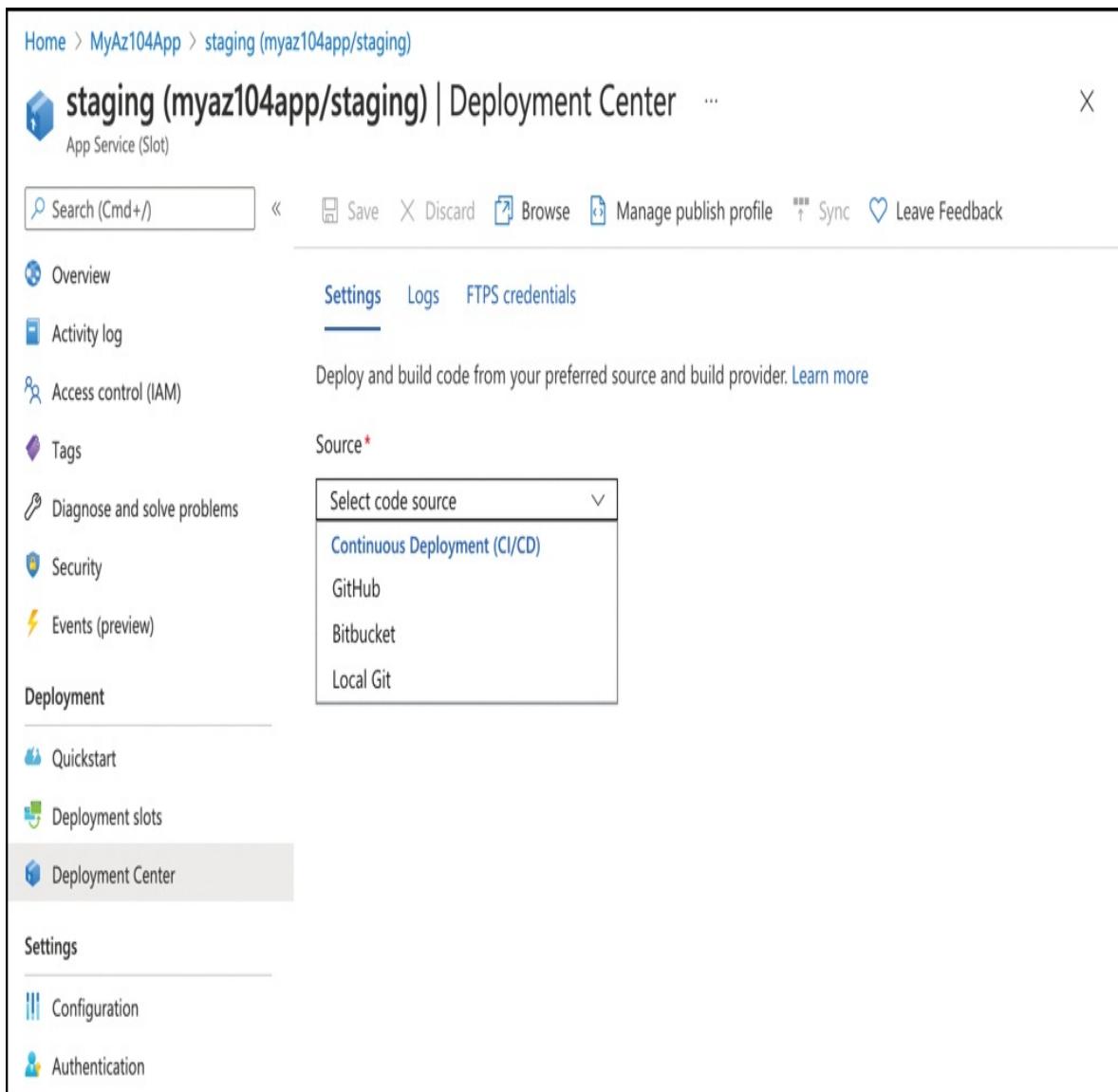


FIGURE 3-68 The Deployment Center

One of the most common deployment options is to use continuous integration and continuous deployment (CI/CD). When using this option, you use a source repository such as GitHub, Bitbucket, or a local Git repository. Whenever you push new code to the source repository, it will automatically publish that new code to your web app. To configure CI/CD, select your source code repository from the drop-down menu and follow the prompts.

App Service also supports deployment using FTPS, and you can find the credentials and other information necessary to use FTPS deployment by clicking the **FTPS Credentials** tab shown previously in [Figure 3-68](#).

Finally, you can deploy to your web app using tools such as Visual Studio, Visual Studio Code, and others by using the publish profile. The publish profile is a configuration file that lists all the methods available for deployment, as well as the credentials necessary for connecting. Visual Studio and Visual Studio Code can both access this publish profile automatically for easy deployment, but you can also click **Manage Publish Profile** shown previously in [Figure 3-68](#) if you want to download the file for use in other applications that support it.

More Info Keep the publish profile secure

It's important to keep the publish profile secured. It contains all the usernames and passwords possible to deploy to your web app using various deployment methods. If this file has become compromised, you can click **Manage Publish Profile** and reset these credentials.

Thought experiment

In this thought experiment, apply what you have learned in this chapter. You can find answers to these questions in the next section, "[Thought experiment answers](#)."

Scenario 1

You are the IT administrator for Contoso, and you are tasked with migrating an existing web farm and database to Microsoft Azure. The web application is written in PHP and is deployed across 20 physical servers running RedHat for the operating system and Apache for the web server. The backend consists of two physical servers running MySQL in an active/passive configuration.

The solution must provide the ability to scale to at least as many web servers as the existing solution and ideally the number of web server instances should automatically adjust based on the demand. All the servers must be reachable on the same network, so the administrator can easily connect to them using SSH from a jumpbox (a VM which is exposed to public IP and used to connect to other VMs in the network using private IPs internally) to administer the VMs.

Answer the following questions for your manager:

1. Which compute option would be ideal for the web servers?
2. How should the servers be configured for high availability?
3. What would be the recommended storage configuration for the web servers? What about the database servers?
4. What feature could be used to ensure that traffic to the VMs only goes to the appropriate services (Apache, MySQL, and SSH)?

Scenario 2

You are the solution architect for Contoso, and you must design a Python-based solution for hosting a web application in Microsoft Azure. Users must be able to access this application from multiple locations, and the application must be available around the clock. Also, the application should be implemented with DevOps capabilities, such as continuous deployment, package management, and the like.

Moreover, you don't want to manage the infrastructure, and you want to avoid the administration as much as possible. You do not want to manage the Windows and software updates on your own.

Answer the following questions about your solution:

1. Which compute option would be ideal for hosting this web application?
2. How will you avoid managing the infrastructure?
3. How will you make sure your application is highly available?

Thought experiment answers

This section contains the solution to the thought experiment for the chapter.

Scenario 1

1. The web servers would be best served by deploying them into a virtual machine scale set (VMSS). Scaling should be configured on the VMSS to address the requirement of automatically scaling up/down the number of instances based on the demand (CPU) used on the web servers.

2. The web servers should be deployed into their own availability set or availability zone if it is available within the region. The database tier should also be deployed into its own availability set or availability zone.
3. The web servers will likely not be I/O intensive so Standard SSD should be appropriate. The database servers will likely be I/O intensive so Premium SSD is the recommended approach. To minimize management overhead and to ensure that storage capacity planning is done correctly managed disks should be used in both cases.
4. Use Network Security Groups (NSGs) to ensure that only traffic destined for allowed services can communicate to the VMs.

Scenario 2

1. Azure App Service will be best suited for this set of requirements. You should deploy a Python web app to App Service on Linux.
2. Azure App Service is a PaaS solution, so a managed infrastructure will host your application. You do not need to worry about the VMs on which the app is deployed. No additional administration efforts are required to manage Windows and software updates.
3. Azure App Service provides 99.95 percent availability for the Basic tier (or higher).

Chapter summary

This chapter focused heavily on creating and configuring virtual machines in Azure as well as automated deployments using Azure Resource Manager templates and even the command line tools. The chapter wrapped up focusing on container services, such as AKS and ACI, followed by Azure App Service and App Service Plans. Let's review some of the key takeaways.

- Each compute family is optimized for either general or specific workloads. You should optimize your VM by choosing the most appropriate size.
- You can create VMs from the Azure portal, PowerShell, the CLI, and Azure Resource Manager templates. You should understand when to use which tool and how to configure the virtual machine resource during provisioning and after provisioning. For example, availability sets can

only be set at provisioning time, but data disks can be added at any time.

- You can connect to Azure VMs using a public IP address or a private IP address with RDP, SSH, or even PowerShell. To connect to a VM using a private IP, you must also enable connectivity such as site-to-site, point-to-site, or ExpressRoute.
- The Custom Script Extension is commonly used to execute scripts on Windows or Linux-based VMs. The PowerShell DSC extension is used to apply desired state configurations to Windows-based VMs.
- A common method of troubleshooting virtual machines with RDP/SSH connectivity or unexplained application issues is to redeploy the virtual machine. Redeploy moves the virtual machine to a different Azure node.
- VM storage comes in Standard HDD, Standard SSD, Premium SSD, with Ultimate SSD in preview. Understanding which tier to choose for capacity and performance planning is important.
- There are unmanaged and managed disks and images. The key difference between the two is with unmanaged disks or images it is up to you to manage the storage account. With managed disks, Azure takes care of this for you, so it greatly simplifies managing images and disks.
- The Azure Diagnostics agent can be enabled on Windows and Linux virtual machines to capture diagnostic, performance, logs and boot diagnostic data.
- Availability Zones provide high availability at the data center level. availability sets provide high availability within a data center.
- Managed disks provide additional availability over unmanaged disks by aligning with availability sets and providing storage in redundant storage units.
- Virtual machine scale sets (VMSS), can scale up to 1,000 instances. You need to ensure that you create the VMSS configured for large scale sets if you intend to go above 100 instances. There are several other limits to consider too. Using a custom image, you can only create up to 300 instances. To scale above 100 instances, you must use the Standard SKU of the Azure Load Balancer or the Azure App Gateway.
- Azure Resource Manager templates are authored using JSON and allow you to define the configuration of resources, such as virtual machines,

storage accounts, and so on in a declarative manner.

- Kubernetes is an open-source container management and orchestration system.
- Azure Kubernetes Service is a PaaS offering of Kubernetes running in Azure. It reduces the configuration and operational overhead of the cluster.
- Azure Container Instances are a way to quickly deploy isolated containers without worrying about backend infrastructure.
- App Service is a PaaS-hosting service that makes it easy to build applications that can process HTTP requests.
- An App Service plan offers computer resources to the web application for its execution. This App Service plan can be shared with multiple web apps, too.

Chapter 4

Configure and manage virtual networking

An Azure virtual network (or VNet) provides the foundation of the Azure networking infrastructure. Virtual machines are connected to virtual networks. This connection provides inbound and outbound connectivity to other virtual machines, to on-premises networks, and to the Internet. Azure provides many networking features that will be familiar to those already experienced in networking, such as the abilities to control which network flows are permitted and to control network routing. These features allow Azure deployments to implement familiar network architectures, such as network segmentation between layers of an N-tier application.

This chapter focuses on the core capabilities that allow you to connect your Azure virtual machines—flexibly and securely.

Skills in this chapter:

- [Skill 4.1: Implement and manage virtual networking](#)
- [Skill 4.2: Secure access to virtual networks](#)
- [Skill 4.3: Configure load balancing](#)
- [Skill 4.4: Monitor and troubleshoot virtual networking](#)
- [Skill 4.5: Integrate an on-premises network with an Azure virtual network](#)

Skill 4.1: Implement and manage virtual networking

Azure Virtual Networks (VNets) form the foundation of the Azure Networking infrastructure. Each virtual network allows you to define a network space, comprising one or more IP address ranges. This network space is then carved into subnets. IP addresses for virtual machines, as well as some other services such as an internal Azure Load Balancer, are assigned from these subnets. Each subnet allows you to define which network flows are permitted (using Network

Security Groups), and what network routes should be taken (using user-defined routes). Together, these features allow you to implement many common network topologies, such as a DMZ containing a network security appliance or a multi-tier application architecture with restricted communications between application tiers.

This section covers how to:

- Create and configure virtual networks, including peering
- Configure private and public IP addresses
- Configure user-defined network routes
- Implement subnets
- Configure Azure DNS, including custom DNS settings and private or public DNS zones

Create and configure a virtual networks and subnets

A virtual network (VNet) is an Azure resource. When creating a VNet, the most important setting is the IP range (or ranges) the VNet will use.

IP ranges are defined using classless inter-domain routing (CIDR) notation. For example, the range 10.5.0.0/16 represents all IP ranges starting with 10.5. (The /16 represents the bitmask and indicates that the first 16 bits are the same for every IP in the address range. Each virtual network can use either a single IP range or multiple disjointed IP ranges.)

Note CIDR Notation

You will need to understand CIDR notation to work effectively with virtual networks in Azure. There are many good explanations to be found online.

For example, see <https://devblogs.microsoft.com/premier-developer/understanding-cidr-notation-when-designing-azure-virtual-networks-and-subnets/>.

Note Virtual network IP ranges

It is normally a good idea to plan your network space in advance. Typically, you will want to avoid creating overlaps with other virtual networks or with on-premises environments because any overlap will prevent you from connecting these networks later.

Your VNet IP ranges must be taken from the private address ranges defined in RFC 1918:

- 10.0.0.0–10.255.255.255 (10.0.0.0/8)
- 172.16.0.0–172.31.255.255 (172.16.0.0/12)
- 192.168.0.0– 192.168.255.255 (192.168.0.0/16)

You can also use public, Internet-addressable IP ranges in your VNet. However, this is not recommended because the addresses within your VNet will take priority, and virtual machines in your VNet will no longer be able to access the corresponding Internet addresses.

In addition, there are a small number of IP ranges reserved by the Azure platform and therefore, cannot be used:

- 169.254.0.0/16 (Link-local)
- 168.63.129.16/32 (Azure-provided DNS)

Subnets

Any Azure resource in a virtual network is deployed into a subnet. Subnets are used to divide the VNet IP space. Different subnets can have different network security and routing rules, enabling applications and application tiers to be isolated and network flows between them to be controlled. For example, consider a typical three-tier application architecture comprised of a web tier, an application tier, and a database tier. By implementing each tier as a separate subnet, you can control precisely which network flows are permitted between tiers and from the Internet.

The name of a subnet must be unique within that VNet. You cannot change the subnet name after it has been created.

Each subnet must also define a single network range (in CIDR format). This range must be contained within the IP ranges defined by the VNet. Only IP addresses from within the subnets can be assigned to virtual machines and other

resources. Subnets do not have to span the entire VNet address space; subnets can be a subset, leaving unused space for future expansion.

Azure will reserve couple of IP addresses from each subnet. Like standard IP networks, Azure reserves the first and last IP addresses in each subnet for network identification and broadcast, respectively.

You are required to define one subnet when creating a VNet using the Azure portal. VNets typically can have multiple subnets, and you can add new subnets to your VNet at any time.

You can't change the address range if there are resources already deployed to the subnet. If you want to make a change to a subnet's address range, you first must delete all the objects in that subnet. If the subnet is empty, you can change the range of addresses to any range that is within the address space of the VNet not assigned to any other subnets.

Subnets can be only be deleted from VNets if they are empty. Once a subnet is deleted, the addresses that were part of that address range are released and available again for use within new subnets that you can create.

Additional virtual network settings

So far, we have focused on the most important settings of each VNet and subnet: the IP address ranges. There are some additional settings and features of VNets and subnets to also be aware of. [Table 4-1](#) provides a summary of few settings supported by virtual networks.

TABLE 4-1 Properties of a virtual Network

Property	Description
Name	The VNet name must be unique within the resource group, is between 2 and 64 characters, and may contain letters (case insensitive), numbers, underscores, periods, or hyphens. It must start with a letter or number and end with a letter, number, or underscore.
Location	Each VNet is tied to a single Azure region and can only be used by resources (such as Virtual Machines) in the same region.

n	
Ad	An array of IP address ranges available for use by subnets.
dre	
ss	
Spa	
ce	
DN	Contains an array of DNS servers. If specified, these DNS servers are configured on virtual machines in the virtual network in place of the Azure-provided DNS servers.
ing	
s	
Su	The list of subnets configured for this VNet.
bne	
ts	
Pe	The list of peerings configured for this VNet. Peerings are used to create network connectivity between separate VNets.
rin	
gs	

[Table 4-2](#) provides a summary of the settings supported by virtual network subnets.

TABLE 4-2 Settings of a virtual network subnet

Prop	Description
erty	
Name	The subnet name must be unique within the VNet. It is between 2-80 characters and may contain letters (case insensitive), numbers, underscores, periods, or hyphens. It must start with a letter or number and must end with a letter, number, or underscore.
Address	The IP address range for a subnet, specified in CIDR notation. All subnets must sit within the VNet address space and cannot overlap.
range	
e	

Network security group	Reference to the network security group (NSG) for the subnet. NSGs can be associated to a subnet and are used to control which inbound and outbound traffic flows are permitted.
Route table	Route table applied to the subnet and used to override the default system routes. These are used to send traffic to destination networks that are different than the routes that Azure uses by default.
Service endpoints (and policies)	An array of Service Endpoints for this subnet. Service Endpoints provide a direct route to various Azure PaaS services (such as Azure storage), without requiring an Internet-facing endpoint. Service Endpoint Policies provide further control over which instances of those services may be accessed.
Delegations	An array of references to delegations on the subnet. Delegations allow subnets to be used by certain Azure services, which will then deploy managed resources (such as an Azure SQL Database Managed Instance) into the subnet. Access to these resources is private and can be controlled using NSGs. Delegations also support access to and from on-premises networks when hybrid networking is used.

Creating a virtual network and subnets using the Azure portal

To create a new VNet by using the Azure portal, first click **Create A Resource** on the home page and then select **Networking**. Next, click **Virtual Network** from the list.

The Create Virtual Network blade opens. Here you can provide configuration information about the virtual network. This blade requires the following inputs, as shown in [Figure 4-1](#).

- Subscription in which the VNet is created
- The resource group where the VNet is created
- Name of the virtual network

- The location for VNet

The following values are set automatically, though you can override them as needed:

- Address space to be used for the VNet using CIDR notation
- Subnet name for the first subnet in the VNet
- The Address Range of the first Subnet

Create virtual network

Basics IP Addresses Security Tags Review + create

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation. [Learn more about virtual network](#)

Project details

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Resource group * ⓘ

(New) ExamRef-RG



[Create new](#)

Instance details

Name *

ExamRef-VNet



Region *

(Canada) Canada Central



FIGURE 4-1 Basics blade while creating virtual network

On the next screen, you can supply address spaces to be used for the VNet using CIDR notation. When creating a VNet using the Azure portal, you can specify multiple IP address ranges, and you can specify one or more subnets (see [Figure 4-2](#)). While creating subnet, you can also create the service endpoints if you desire to use any of the Azure Services.

Home > New >

Create virtual network

Basics IP Addresses Security Tags Review + create

The virtual network's address space, specified as one or more address prefixes in CIDR notation (e.g. 192.168.1.0/24).

IPv4 address space

10.1.0.0/16 10.1.0.0 - 10.1.255.255 (65536 addresses)



Add IPv6 address space ⓘ

The subnet's address range in CIDR notation (e.g. 192.168.1.0/24). It must be contained by the address space of the virtual network.

+ Add subnet Remove subnet

Subnet name

Subnet address range

Apps

10.1.0.0/24

Edit subnet

Subnet name *

Apps

Subnet address range * ⓘ

10.1.0.0/24

10.1.0.0 - 10.1.0.255 (251 + 5 Azure reserved addresses)

SERVICE ENDPOINTS

Create service endpoint policies to allow traffic to specific Azure resources from your virtual network over service endpoints. [Learn more](#)

Services ⓘ

Microsoft.Storage



[Review + create](#)

< Previous

Next : Security >

[Download a template for automation](#)

Save

Cancel

FIGURE 4-2 Define subnets while creating virtual network

The following blade also allows you to enable some additional settings, related to Bastion Host, DDoS protection, and the Azure Firewall service, as shown in [Figure 4-3](#).

The screenshot shows the 'Create virtual network' blade in the Azure portal. At the top, there's a breadcrumb navigation: Home > New > Virtual Network >. The main title is 'Create virtual network'. Below the title, there are tabs: Basics, IP Addresses, Security (which is underlined, indicating it's the active tab), Tags, and Review + create. The Security tab contains three sections with toggle buttons:

- BastionHost**: A radio button labeled 'Disable' is selected. There is also an 'Enable' option.
- DDoS Protection Standard**: A radio button labeled 'Disable' is selected. There is also an 'Enable' option.
- Firewall**: A radio button labeled 'Disable' is selected. There is also an 'Enable' option.

FIGURE 4-3 Security blade while creating virtual network

Once the VNet has completed provisioning, you can review the settings using the Azure portal. Notice the Apps subnet has been created as part of the inputs

shown in [Figure 4-4](#).

To create another subnet in the VNet, click **+Subnet** on this blade and provide the following inputs, as shown in [Figure 4-5](#):

- The **Name** of the subnet
- The **IP Address Range (CIDR Block)**
- The **NAT Gateway**
- The **Network Security Group** (if any)
- The **Route Table** (if any)
- The **Service Endpoints**
- The **Subnet Delegation**

The screenshot shows the Azure portal interface for managing subnets. The left sidebar lists various resources under the 'Virtual network' category, with 'Subnets' currently selected. The main content area displays a table of subnets for the 'ExamRef-VNet'. The table has columns for Name, IPv4, IPv6 (many available), Delegated to, and Security group. A single row is visible, representing an subnet named 'Apps' with the IPv4 range 10.1.0.0/24 (251 available). The top navigation bar includes a search bar, a 'Subnet' button, a 'Gateway subnet' button, and a 'Refresh' button.

Name ↑↓	IPv4 ↑↓	IPv6 (many available) ↑↓	Delegated to ↑↓	Security group ↑↓
Apps	10.1.0.0/24 (251 available)	-	-	-

FIGURE 4-4 Subnets for ExamRef-VNet virtual network

Add subnet

X

Name *

Data



Subnet address range * ⓘ

10.0.1.0/24



10.0.1.0 - 10.0.1.255 (251 + 5 Azure reserved addresses)

Add IPv6 address space ⓘ

NAT gateway ⓘ

None



Network security group

None



Route table

None



SERVICE ENDPOINTS

Create service endpoint policies to allow traffic to specific azure resources from your virtual network over service endpoints. [Learn more](#)

Services ⓘ

0 selected



SUBNET DELEGATION

Delegate subnet to a service ⓘ

None



FIGURE 4-5 Add Subnet blade, which is used to add a new subnet to an existing virtual network

Create and configure VNet peering

VNet peering allows virtual machines in two separate virtual networks to communicate directly by using their private IP addresses. The VNets can either be in the same Azure region or in separate Azure regions. Peering between VNets in different regions is called Global VNet peering. In all cases, traffic between peered VNets travels over the Microsoft backbone infrastructure, not the public Internet.

Note VNet Peering

You can peer VNets in different subscriptions, even if those subscriptions are under different Azure Active Directory tenants.

You can also use VNet peering to connect Resource Manager VNets to the older “classic” VNets. However, peering between two classic VNets is not supported. (A VNet-to-VNet VPN can be used in this case.)

The peered VNets must have non-overlapping IP address spaces. In addition, the VNet address space cannot be modified once the VNet is peered with another VNet.

More Info Peering Requirements and Constraints

There are a few requirements and constraints to keep in mind while peering the VNets, which are found here: <https://docs.microsoft.com/azure/virtual-network/virtual-network-manage-peering#requirements-and-constraints>.

VNet peering provides the similar network performance between VMs as if they were placed in a single large VNet within the same region. There is no bandwidth cap imposed on peered VNets. The only limits are those on the VMs themselves, based on VM series and size.

Note Peering Limits

Be aware of the limit of 500 peering connections per VNet. This is a hard limit.

No VNet gateways are required by VNet peering. This avoids the cost, throughput limitations, additional latency, and additional incurred complexity associated with using VNet gateways, though you can use VNet gateways to connect to on-premises networks using gateway transit.

Note Global Peering Limitations with the Load Balancer's Basic TIER

Global peering cannot be used to access the front-end IP of a basic internal Azure load-balancer in the remote virtual network. In these cases, a VNet-to-VNet VPN should be used instead. This limitation doesn't apply with the standard tier of the Load Balancer.

There are no restrictions on connectivity between the peered VNets, so virtual machines in peered VNets can communicate with each other as if they were in the same VNet. In addition, the `virtualNetwork` service tag (described in [Skill 4.3](#)) spans the address space of both peered networks.

Alternatively, you also have the option to limit connectivity by using the **Allow Virtual Network Access** option, there is no automatic outbound connectivity between peered VNets, and the `virtualNetwork` service tag does not include the address space of the peered VNet. In this case, you control the connectivity between peered virtual networks using network security groups.

A simple example of VNet peering is shown in [Figure 4-6](#). This shows two VNets which have been connected using VNet peering. This allows (for example), the WEB1 virtual machine in VNetA to connect to the MYSQL1 database in VNetB.

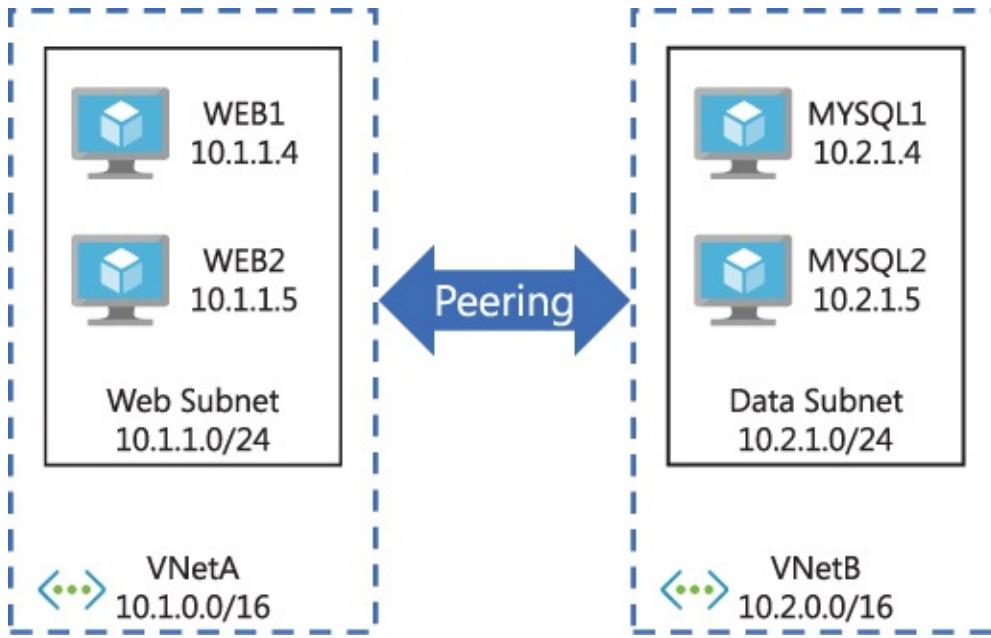


FIGURE 4-6 VNet Peering between two virtual networks

Once peered, traffic between VMs is routed through the Microsoft backbone infrastructure. Traffic does not pass over the public Internet, even when using global VNet peering to connect VNets in different Azure regions.

While global VNet peering allows for open connectivity between virtual machines across VNets in different Azure regions, a limitation is that a VM can only connect to the front-end IP address of a basic internal Azure Load Balancer in the same region.

It is important to understand that VNet peering is a pairwise relationship between two virtual networks. To create connectivity across three virtual networks (VNetA, VNetB, and VNetC), all three pairs must be peered (VNetA to VNetB; VNetB to VNetC; and VNetA to VNetC). This is illustrated in [Figure 4-7](#).

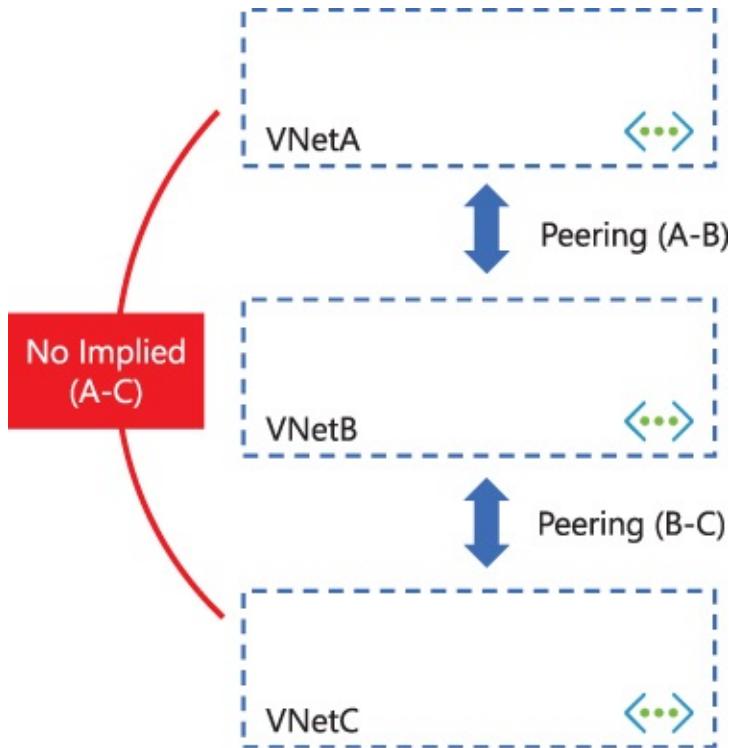


FIGURE 4-7 VNet peerings do not have a transitive relationship

Service chaining and hub-and-spoke networks

A common way to reduce duplication of resources is to use a hub-and-spoke network topology. In this approach, shared resources (such as domain controllers, DNS servers, monitoring systems, and so on) are deployed into a dedicated hub VNet. These services are accessed from multiple applications, each deployed to their own separate spoke VNets.

As you have just seen, VNet peering is not transitive. This means there is no automatic connectivity between spokes in a hub-and-spoke topology. Where such connectivity is required, one approach is to deploy additional VNet peerings between spokes. However, with a large number of spokes, this can quickly become unwieldy.

An alternative approach is to deploy a network virtual appliance (NVA) into the hub through user-defined routes (UDRs) to route inter-spoke traffic through the NVA. This is known as *service chaining*, and it enables spoke-to-spoke communication without requiring additional VNet peerings, as illustrated in Figure 4-8.

To transit traffic from one spoke VNet to another spoke VNet via an NVA in

the hub VNet, the VNet peerings must be configured correctly. By default, a peering connection will only accept traffic originating from the VNet to which it is connected. This will not be the case for traffic forwarded between spoke VNets via an NVA in a hub VNet. To permit such traffic, the **Allow Forwarded Traffic** setting must be enabled for those VNet peerings.

Sharing virtual network gateways

Suppose you want two peered VNets, say VNet-A and VNet-B, want to send traffic to an external network via a virtual network gateway. Rather than deploy two virtual network gateways, it is much simpler and more cost-efficient for both VNets to share a single gateway. This can be achieved with local or global peering.

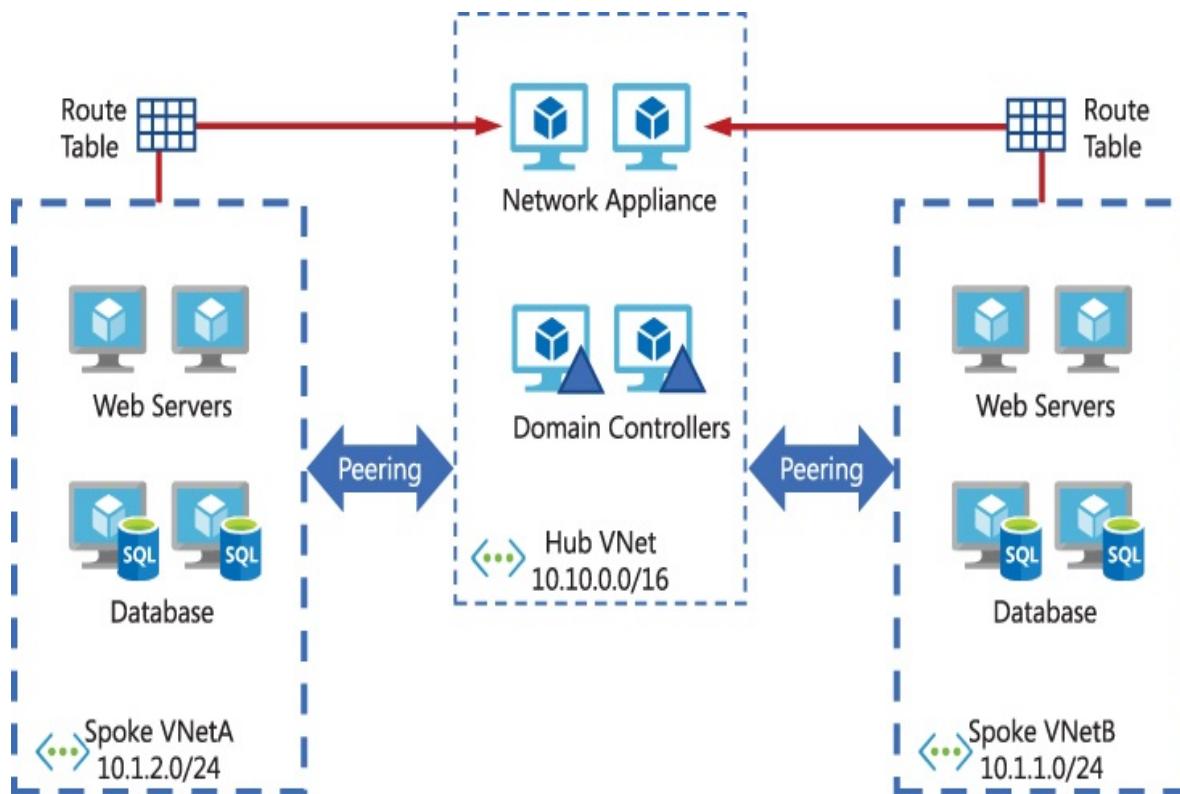


FIGURE 4-8 Service chaining allows for the use of common services across VNet Peerings

Suppose the virtual network gateway is deployed to VNet-A, allowing VNet-A to communicate with the external network. By default, only traffic originating

in VNet-A is permitted to use this gateway, and the external network is only able to connect to VMs in VNet-A. To allow connectivity between VNet-B and the external network, the following settings must be configured:

- **Use Remote Gateways.** This setting must be enabled on the peering connection from VNET-B to VNET-A. This informs VNET-B of the availability of the gateway in VNET-A. Note that to enable this setting, VNET-B cannot have its own virtual network gateway.
- **Allow Gateway Transit.** This option must be enabled on the peering connection from VNET-A to VNET-B. This permits traffic from VNET-B to use VNET-A's gateway to send traffic to the external network. Gateway transit can be used for S2S, P2S, and VNet to VNet.

Note that in this case, the **Allow Forwarded Traffic** peering option is not required.

Creating a VNet peering using the Azure portal

To create a peering connection between two VNets, the VNets must already have been created and must not have overlapping address spaces.

To create a new VNet peering from VNet1 to VNet2, connect to the Azure portal and locate VNet1. Under **Settings**, click **Peering**, and then select **+Add** to open the **Add Peering** blade. Use the following steps to set up a standard peering connection, as shown in [Figure 4-9](#).

1. Choose a name for the peering from VNet1 to VNet2. In this example, we will use `VNet1 - to - VNet2`.
2. Enter the peer details. You can choose **Resource Manager** or **Use Classic**. In this example, we will choose **Resource Manager**.
3. Select the subscription for VNet2 from the **Subscription** drop-down menu.
4. From the **Virtual Network** dropdown, choose **VNet2**.
5. Set the **Name Of The Peering From Vnet2 To Vnet1** option to **VNet2-To-VNet1**.



Tip: Known resource ID

Alternatively, instead of entering a name, you can specify the peer VNet by selecting the **I Know My Resource ID** checkbox and entering the peer VNet resource ID.

6. Select the following virtual network access settings:
 - **Allow Virtual Network Access From VNet1 To VNet2.** Enabled
 - **Allow Virtual Network Access From VNet2 To VNet1.** Enabled
7. Select the forwarded traffic settings:
 - **Allow Forwarded Traffic From VNet2 To VNet1.** Disabled
 - **Allow Forwarded Traffic From VNet1 To VNet2.** Disabled
8. Under **Configure Gateway Transit Settings**, leave **Allow Gateway Transit** unselected.

Add peering

VNet1

i For peering to work, a peering link must be created from VNet1 to VNet2 as well as from VNet2 to VNet1.

Name of the peering from VNet1 to VNet2 *

VNet1-to-VNet2



Peer details

Virtual network deployment model **i**

Resource manager Classic

I know my resource ID **i**

Subscription * **i**

Visual Studio Ultimate with MSDN



Virtual network *

VNet2 (ExamRef-RG)



Name of the peering from VNet2 to VNet1 *

VNet2-to-VNet1



Configuration

Configure virtual network access settings

Allow virtual network access from VNet1 to VNet2 **i**

Disabled Enabled

Allow virtual network access from VNet2 to VNet1 **i**

Disabled Enabled

Configure forwarded traffic settings

Allow forwarded traffic from VNet2 to VNet1 **i**

Disabled Enabled

Allow forwarded traffic from VNet1 to VNet2 **i**

Disabled Enabled

Configure gateway transit settings

Allow gateway transit **i**

OK

FIGURE 4-9 Adding peering from VNet1 to VNet2 using the Azure portal

9. Click **OK** to create the peering between VNet1 and VNet2. Once the peering has completed provisioning, it will appear in Azure portal with the peering status as **Connected** to peer network VNet2, as shown in [Figure 4-10](#).

Name	Peering status	Peer	Gateway transit
VNet1-to-VNet2	Connected	VNet2	Disabled

FIGURE 4-10 VNet1-to-VNet2 peering showing as Connected in the Azure portal

10. If you return to the peering blade of VNet2, you will see that the **Peering Status** of VNet2 to VNet1 shows as **Connected** (see [Figure 4-11](#)).

Name	Peering status	Peer	Gateway transit
VNet2-to-VNet1	Connected	VNet1	Disabled

FIGURE 4-11 VNet2-to-VNet1 peering showing as Connected in the Azure portal

Now, VNet1 and VNet2 are peers, and VMs on these networks can communicate with each other as if this was a single virtual network.

Configure private and public IP addresses and network interfaces

VMs in Azure use TCP/IP to communicate with: services in Azure, other VMs you have deployed in Azure, on-premises networks, and the Internet. Just as a physical server uses a network interface card (NIC) to connect to a physical network, virtual machines use a network interface resource (also referred to as a NIC) to connect to a virtual network or the Internet.

There are two types of IP addresses you can use in Azure:

- **Public IP addresses.** Used for communication with the Internet
- **Private IP addresses.** Used for communication within Azure virtual networks and connected on-premises networks

This section focuses on how to deploy and manage private IP addresses and network interfaces. Public IP addresses are discussed in the next section.

Network interfaces

Both public and private IP addresses are configured on virtual machines using network interface resources. Therefore, to understand how to use public and private IP addresses with your virtual machine, you first must understand network interfaces. A network interface is a standalone Azure resource. Because its only purpose is to provide network connectivity for virtual machines, it is typically provisioned and deleted with its corresponding virtual machine.

Just as a physical server can have more than one network card, you can associate multiple network interfaces with a single virtual machine. This is a common practice when configuring virtual machines to act as network virtual appliances. These appliances provide network security as well as routing and other features similar to physical network devices in a traditional network.

[Table 4-3](#) details the settings of each network interface resource in Azure.

TABLE 4-3 Network interface settings

Setting	Description
Name	This is the network interface name, which must be unique within the

e	resource group. It can be between 1-80 characters, and it may contain letters (case insensitive), numbers, underscores, periods, or hyphens. The name must start with a letter or number and end with a letter, number, or underscore.
Locat ion	This is the location of the resource, and it must be the same as the location of any virtual network or any virtual machine to which the network interface will be connected.
DNS Settings	If specified, these DNS servers are configured on virtual machines in the virtual network in place of the Azure-provided DNS servers. This setting will override the VNet-level DNS settings, if both are specified.
IP Forwarding	Used to enable IP forwarding on this network interface. It allows the VM using the interface to receive traffic that is not sent to one of the IPs in the IP configurations. Also, it allows the VM to send traffic using an IP address that is not in the IP configurations.
IP Configurations	This is a list of IP configurations for the network interface. These are the most important settings, which contain the public and private IP addresses.
Netw ork Security Group Properties	This setting will display the name of any NSGs associated with this interface.
Accelerate Networking	This setting is used to enable accelerated networking, though it is only supported on certain VM sizes.

The setting of the network interface is the IP configuration. This is where the public and private IP address settings are configured. Each network interface

supports an array of IP configurations, which enables each network interface to support multiple IP addresses.

Private IP addresses

Private IP addresses are configured within the IP configurations of the network interface. They are not a separate resource. Each IP configuration specifies a single subnet, and the private IP address is allocated from the address space of that subnet.

There are two methods used to assign private IP addresses: dynamic or static. The default allocation method is dynamic, where the IP address is automatically allocated from the resource's subnet (using an Azure DHCP server).

Dynamic allocation assigns private IP addresses from each subnet in order, starting with the lowest available IP in the subnet IP range. Remember that the first four IP addresses in each subnet are reserved by the Azure platform. For example, if the subnet is 10.10.0.0/24, the first private IP to be allocated will be 10.10.0.4 (because 10.10.0.0 to 10.10.0.3 are reserved).

A Dynamically allocated IP address can change when you stop and start the associated virtual machine. To avoid this, private IP addresses can also be allocated statically. This is used where you want to control which IP address is assigned to a specific server and for that IP address to remain fixed.

Static private IP addresses are commonly used for:

- Virtual machines that act as domain controllers or DNS servers
- Resources that require firewall rules using IP addresses
- Resources accessed by other apps/resources through an IP address explicitly, rather than a domain name

To configure a static private IP address, simply specify the static IP allocation within the network interface IP configuration, together with the desired IP address. By default, when you change to static, Azure will assign the previously assigned dynamic IP address.

When changing a private IP address, you might need to manually review and update the VM's network settings. For this reason, it is preferable to plan and specify static private IP addresses in advance when first provisioning the virtual machine.

Note Configuring Static Private IP addresses

Static private IP addresses should only be configured in the Azure network interface resource. They will be assigned to the virtual machine using DHCP, just like with dynamic private IP addresses.

Both IPv4 and IPv6 private IP addresses are supported. Each network interface must have one private IPv4 address assigned as the primary IP configuration. You can add one or more IPv4 address as secondary IP configurations. Each network interface can be assigned zero or a maximum of one private IPv6 address as a secondary IP configuration.

Note Dynamic and Static Private IP assignment

Private IPv4 address assignments can be either dynamic or static. Private IPv6 addresses can only be assigned dynamically.

Enabling static private IP addresses on VMs with the Azure portal

The network interface of a VM holds the configurations of the private IP address. This is known as the IP configuration. Using the Azure portal, you can modify the private IP address allocation method for the IP configuration from dynamic to static. You can also use the Azure portal to manage other network interface settings, such as assigning network security groups, public IP addresses, and adding new IP configurations.

Using the Azure portal, locate the network interface for the VM to be assigned a static IP address. Once the blade loads for the NIC, click **IP Configurations** and then select the IP configuration you want to update. The IP Configuration blade is shown in [Figure 4-12](#). Here, you can update the private IP address allocation method to **Static** and specify the **IP Address**.

ipconfig1

examref913

 Save  Discard

! The virtual machine associated with this network interface will be restarted to utilize the new private IP address. The network interface will be re-provisioned and network configuration settings, including secondary IP addresses, subnet masks, and default gateway, will need to be manually reconfigured within the virtual machine. [Learn more](#)

Public IP address settings

Public IP address

[Disassociate](#) [Associate](#)

Private IP address settings

Virtual network/subnet

VNet1/default

Assignment

[Dynamic](#) [Static](#)

IP address *

10.0.0.5



FIGURE 4-12 Assigning a Static Private IP Address to a NIC

Create and configure public IP addresses

Associating a public IP address with a network interface creates an Internet-facing endpoint, allowing your virtual machine to receive network traffic directly from the Internet.

A public IP address is a standalone Azure resource. This contrasts with a private IP address that exists only as a collection of settings on another resource, such as a network interface or a Load Balancer.

To associate a public IP address with a virtual machine, the IP configuration of the network interface must be updated to contain a reference to the public IP address resource. As a standalone resource, public IP addresses can be created and deleted independently as well as moved from one virtual machine to another.

Basic vs Standard Pricing Tiers

Public IP addresses are available at two pricing tiers (or SKUs): Basic or Standard. All Public IP Addresses created before the introduction of these tiers are mapped to the Basic tier.

The Standard tier Public IP Addresses support zone-redundant deployment, allowing you to use availability zones to protect your deployments against potential outages caused by data center-level failures (such as fire, power failure, or cooling failure). There are a number of other important differences between the two tiers, as summarized in [Table 4-4](#).

TABLE 4-4 Comparison of public IP Address Basic and Standard Tiers

Feature	Basic Tier	Standard Tier
Allocation method	Supports both static and dynamic allocation methods.	Supports static allocation only.

Traffic restrictions	Open by default for inbound traffic. Use NSGs to restrict inbound or outbound traffic.	Closed by default for inbound traffic. Use NSGs to allow inbound traffic and restrict outbound traffic.
Redundancy	Not zone redundant and doesn't support availability zone.	Zone redundant by default, or it can instead be assigned to a specific availability zone
Public IP prefixes	Does not support public IP prefixes (discussed later).	Supports public IP prefixes, allowing IP addresses to be assigned from a contiguous IP address block.

Public IP address allocation

As with private IP addresses, public IP addresses support both dynamic and static IP allocation. For the Basic tier, both static and dynamic allocation are supported, the default being dynamic. For the Standard tier, only static allocation is supported.

Under dynamic allocation, an actual IP address is only allocated to the public IP address resource when the resource is in use—that is, when it is associated with a resource such as a running virtual machine. If the virtual machine is stopped (deallocated) or deleted, the IP address assigned to the public IP address resource is released and returned to the pool of available IP addresses managed by Azure. When you restart the virtual machine, a different IP address will most likely be assigned.

If you want to retain the IP address, the public IP address resource should be configured to use static IP allocation. An IP address will be assigned immediately (if one was not already dynamically assigned). This IP address will never change, regardless of whether the associated virtual machine is stopped or deleted.

Typically, static public IP addresses are used in scenarios where a dependency is defined by a particular IP address. For example, static IP addresses are commonly used in the following scenarios:

- Where firewall rules specify an IP address
- Where a DNS record would need to be updated when an IP address

changes

- Where the source IP address is used as a (weak) form of authentication of the traffic source
- Where an SSL certificate specifies an explicit IP address rather than a domain name

With private IP addresses, static allocation allows you to specify the IP address to use from the available subnet address range. In contrast, static allocation of public IP addresses does not allow you to specify which public IP address to use. Azure assigns the IP address from a pool of IP addresses in the Azure region where the resource is located.

Public IP address prefixes

When using multiple public IP addresses, it can be convenient to have all of the IP addresses allocated from a single IP range or prefix. For example, when configuring firewall rules, this allows you to configure a single rule for the prefix, rather than separate rules for each IP address.

To support this scenario, Azure allows you to reserve a public IP address prefix. Public IP address resources associated with that prefix will have their IP addresses assigned from that range, rather than from the general-purpose Azure pool.

When creating a prefix, specify the prefix resource name, subnet size (for example, /28 for 16 IP addresses), and the Azure region where the IP addresses will be allocated.

Once the prefix is created, individual public IP addresses can be created that are associated with this prefix. Note that only standard-tier public IP addresses support allocation from a prefix, and thus only static allocation is supported. The IP address assigned to these resources will be taken from the prefix range—you cannot specify a specific IP address from the range.

Note Prefixes benefits and constraints

- See the following links to the benefits and constraints of public IP address prefixes:
Benefits: <https://docs.microsoft.com/azure/virtual-network/public-ip-address-prefix#benefitsConstraints>:
<https://docs.microsoft.com/azure/virtual-network/public-ip-address-prefix#constraints>

DNS Labels

The Domain Name System (DNS) can be used to create a mapping from a domain name to an IP address. This allows you to reference IP address endpoints using a domain name, rather than using the assigned IP address directly.

There are four ways to configure a DNS label for an Azure public IP address:

1. By specifying the DNS name label property of the public IP address resource
2. By creating a `DNS A` record in Azure DNS or a third-party DNS service hosting a DNS domain
3. By creating a `DNS CNAME` record in Azure DNS or a third-party DNS service hosting a DNS domain
4. By creating an alias record in Azure DNS

Specifying DNS name label property

With this option, you specify the left-most part of the DNS label as a property in the public IP address resource. Azure provides the DNS suffix, which will be of the form `<region>.cloudapp.azure.com`. The DNS label you provide is concatenated with this suffix to form the fully qualified domain name (FQDN), which can be used to look up the IP address via a DNS query.

For example, if your public IP address is deployed to the Central US region, and you specify the DNS label `contoso-app`, then the FQDN will be `contoso-app.centralus.cloudapp.azure.com`.

The major limitation of this approach is that the DNS suffix is taken from an Azure-provided DNS domain. It does not support the use of your own vanity domain, such as `contoso.com`. To address this, you will need to use one of the other approaches.

Creating a `DNS A` record

In this approach, you will have already hosted your vanity domain either in Azure DNS or a third-party DNS service. Using your hosting service, you can create a DNS entry in your vanity domain mapping to your public IP address

resource. If you use a DNS `A` record, which maps directly to an IP address, you will need to update the DNS record if the assigned IP address changes. To avoid this, you will probably prefer using static rather than dynamic IP allocation.

Creating a `DNS CNAME` record

In this approach, you start by creating a DNS label for your public IP address. You then create a `CNAME` record in your vanity domain, which maps your chosen domain name to the Azure-provided DNS name. For example, you might map www.contoso.com to `contoso-app.centralus.cloudapp.azure.com`. This approach has the advantage of avoiding the need for static IP allocation because the Azure-provided DNS entry updates automatically if the assigned IP address changes. However, the downside of this approach is that the Domain Name System does not support `CNAME` records at the apex (or root) of a DNS domain, which means while you can create a `CNAME` record for www.contoso.com, you cannot create one for `contoso.com` (without the `www`).

Creating an alias record

In this approach, your vanity domain must be hosted in Azure DNS. You can then create an alias record, which works the same as an `A` record, except that rather than specifying the assigned IP address value explicitly in the DNS record, you simply reference the public IP address resource. The assigned IP address is taken from this resource and automatically configured in your DNS alias record. With alias records, the DNS record is automatically updated if the assigned IP address changes, avoiding the need for static IP allocation.

Outbound Internet connections

When a public IP address is assigned to a virtual machine's network interface, outbound traffic to the Internet will be routed through that IP address. The recipient will see your public IP address as the source IP address for the connection.

However, the virtual machine itself does not see the public IP address in its network settings—it only sees the private IP address. Traffic leaves the virtual machine via the private IP address, and Source Network Address Translation (SNAT) is used to map the outbound traffic from the private IP address to the public IP address.

Note that a public IP address is not required for outbound Internet traffic.

Even without a public IP address assigned, virtual machines can still make outbound Internet connections. In this case, SNAT is used to map the private IP address to the Internet-facing IP address.

IPv4 and IPv6

Public IP address resources can use either an IPv4 or IPv6 address (but not both). Note that IPv6 support is limited as follows:

- Only the Basic tier is supported.
- Only dynamic allocation is supported.

Creating a public IP address using the Azure portal

Creating a new public IP address is a simple process when using Azure portal. Click **New**, and then search for **public IP address** in the marketplace. Like all resources in Azure, some details will be required, including the name of the resource, the SKU (or pricing tier), the DNS name label, idle time-out, subscription, resource group, and location/region. For the Basic SKU, you also specify the IP version and static or dynamic assignment. For the Standard SKU, choose between zone-redundant deployment or a specific availability zone.

The location is critical, as an IP address must be in the same location/region as the virtual machine or other resource that will use it. [Figure 4-13](#) shows the **Azure Create Public IP Address** blade.

Configure network routes

Network routes control how traffic is routed in your network. Azure provides default routing for common scenarios, with the ability to configure your own network routes where necessary.

System routes

Azure VMs that are in the same VNet can communicate automatically with each other and with the Internet without any explicit configuration changes, even when they are in different subnets. This is also the case for communication from the VMs to your on-premises network when a hybrid connection from Azure to your data center has been established.

This ease of setup is made possible by what is known as system routes, which define how IP traffic flows in Azure VNets. The following are the default system routes that Azure will use and provide for you:

- Within the same subnet
- From one subnet to another within a VNet
- VMs to the Internet
- A VNet to another VNet through a VPN gateway (optional)
- A VNet to another VNet through VNet peering (optional)
- A VNet to your on-premises network through a VPN gateway or ExpressRoute (optional)
- VirtualNetworkServiceEndpoint (optional)

Home > New > Public IP address >

Create public IP address

IP Version * (i)

IPv4 IPv6 Both

SKU * (i)

Basic Standard

IPv4 IP Address Configuration

Name *

Exam-Ref-IP



IP address assignment

Dynamic Static

Idle timeout (minutes) * (i)

4

DNS name label (i)

examref-ip



.canadacentral.cloudapp.azure.com

Subscription *

Visual Studio Ultimate with MSDN



Resource group *

ExamRef-RG



[Create new](#)

Location *

(Canada) Canada Central



Availability zone * (i)

Zone-redundant



[Create](#)

[Automation options](#)

FIGURE 4-13 Creating a Public IP Address in the Azure portal

Figure 4-14 shows an example of how these system routes make it easy to get up and running. System routes provide for most typical scenarios by default, without you having to make any routing configuration.

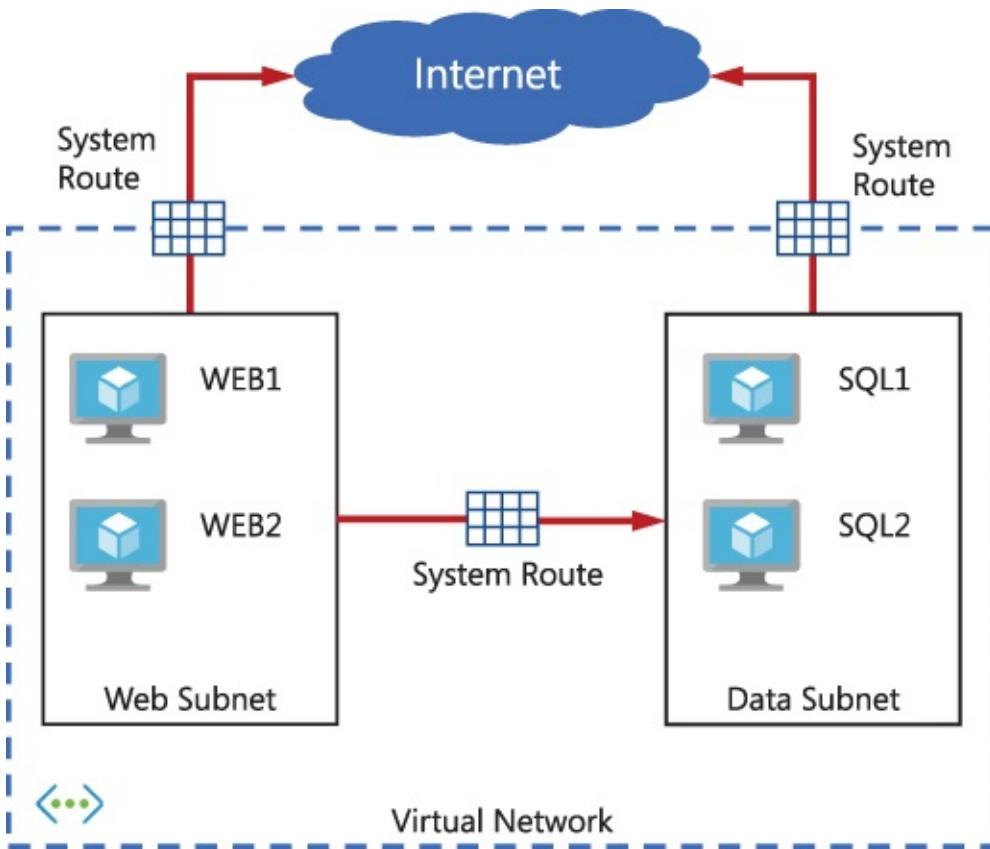


FIGURE 4-14 N-Tier application deployed to Azure VNet using System Routes

User-defined routes

There are some use cases where you will want to configure the routing of packets differently from what is provided by the default system routes. One of these scenarios is when you want to send traffic through a network virtual appliance, such as a third-party Load Balancer, firewall, or router deployed into your VNet from the Azure Marketplace.

To make this possible, you must create what are known as user-defined

routes (UDRs). The UDR is implemented by creating a route table resource. Within the route table, a number of routes are configured. Each route specifies the destination IP range (in CIDR notation) and the next hop IP address. A variety of different types of next hop are supported:

- **Virtual Appliance.** A virtual machine running a network application such as a load-balancer or firewall. With this next hop type, you also specify the IP address of the appliance, which can be a virtual machine or internal load-balancer for high-availability virtual appliances.
- **Virtual Network Gateway.** Used to route traffic to a VPN Gateway (but not an ExpressRoute Gateway, which uses BGP for custom routes). Because there can be only one VPN Gateway associated with a VNet, you are not prompted to specify the actual gateway resource.
- **Virtual Network.** Used to route traffic within the Virtual Network.
- **Internet.** Used to route a specific IP address or prefix to the Internet.
- **None.** Used to drop all traffic sent to a given IP address or prefix.

This route table is then associated with one or more subnets. Traffic originating in the subnet whose destination matches the destination IP range of a route table rule will instead be routed to the corresponding next hop IP address. The service running at this IP address is responsible for all onward routing.

Note Route Tables

You can have multiple route tables, and the same route table can be associated to one or more subnets. Each subnet can only be associated to a single route table. All VMs in a subnet use the route table associated to that subnet.

[Figure 4-15](#) shows a UDR that has been created to direct outbound traffic via a virtual appliance. In this case the appliance is a firewall running as a VM in Azure in the DMZ subnet. The same appliance can also be used to filter traffic between the Apps and Data subnets. An example route table implementing this design is shown in [Figure 4-16](#).

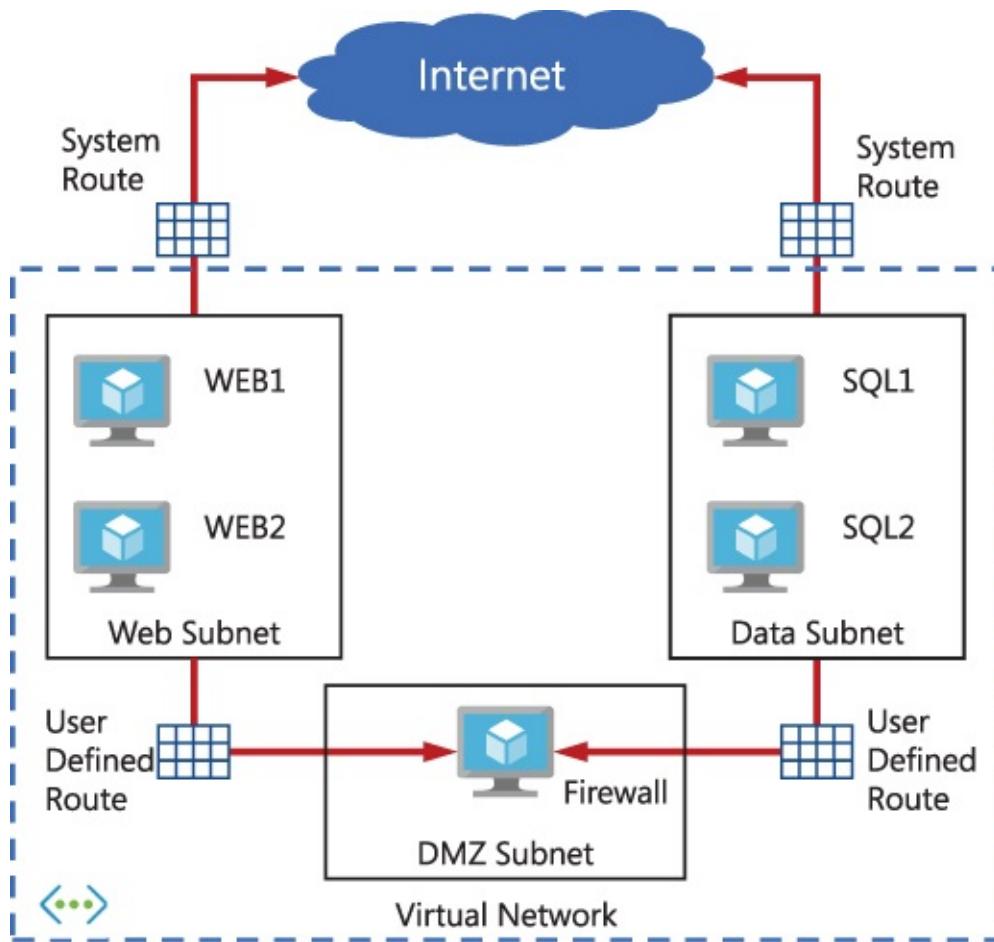


FIGURE 4-15 N-Tier application deployed with a firewall using user-defined routes

Home >



ExamRef-UDR | Routes

Route table

Search (Ctrl+ /)

«



Settings



Search routes

Name	↑↓ Address prefix	↑↓ Next hop type
AppsToData	10.0.0.0/24	10.10.99.4
DataToApps	10.0.1.0/24	10.10.99.4
ToInternet	0.0.0.0/24	10.10.99.4

FIGURE 4-16 Route table rules forcing network traffic through a firewall

Note Dedicated Subnets for Network Appliances

Do not apply a route table to a subnet if the route table contains a rule with a next hop address within that subnet. To do so could create a routing loop. For this reason, virtual network appliances should be deployed to dedicated subnets, separate from the resources that route through that appliance.

IP forwarding

User-defined routes (UDR) allow for changing the default system routes that Azure creates for you in an Azure VNet. In the virtual appliance scenario, the UDRs forward traffic to a virtual appliance such as a firewall, which is running as an Azure virtual machine.

By default, a virtual machine in Azure will not accept a network packet addressed to a different IP address. For that traffic to be allowed to pass into that virtual appliance, you must enable IP forwarding on the network interface of the virtual machine. This configuration doesn't typically involve any changes to the Azure UDR or VNet, but depending on the scenario, you might need to make some configuration changes in the VM's OS to enable this to work correctly.

IP forwarding can be enabled on a network interface by using the Azure portal, PowerShell, or the Azure CLI. In [Figure 4-17](#), you see that the network interface of the NGFW1 VM has the IP forwarding set as Enabled. This VM is now able to accept and send packets that were not originally intended for this VM.



examref913 | IP configurations

Network interface

« [Add](#) [Save](#) [Discard](#)



Overview [IP forwarding settings](#)



Activity log [IP forwarding](#)

[Disabled](#) [Enabled](#)



Access control (IAM) [Virtual network](#)

VNet1



Tags [IP configurations](#)

Settings



IP configurations

Subnet *

[default \(10.0.0.0/24\)](#)



DNS servers [Search IP configurations](#)



Name	IP Version	Type	Private IP address	Public IP address
------	------------	------	--------------------	-------------------



Properties	ipconfig1	IPv4	Primary	10.0.0.5 (Static)
------------	-----------	------	---------	-------------------



FIGURE 4-17 IP forwarding enabled on network interface

How routes are applied

A given network packet may match multiple route table rules. When designing and implementing custom routes, it's important to understand the precedence rules that Azure applies.

If multiple routes contain the same address prefix, Azure selects the route type, based on the following priorities:

1. User-defined routes
2. System routes for traffic in a virtual network, across a virtual network peering, or to a virtual network service endpoint
3. BGP routes
4. Other system routes

Within a single route table, a given network packet may match multiple routing rules. There is no explicit precedence order on the rules in a route table. Instead, precedence is given to the rule with the most specific match to the destination IP address. If an IP address matches two rules, the longest prefix match algorithm is used to select the route.

For example, if a route table contains one rule for prefix `10.10.0.0/16`, and another rule for `10.10.30.0/28`, then any traffic to IP address `10.10.30.4` will be matched against the second rule in preference to the first.

When troubleshooting networking issues, it can be useful to get a deeper insight into exactly which routes are being applied to a given network interface. The effective routes feature of each network interface allows you to see the full details of every network route applied to that network interface, giving you full insight into how each outbound connection will be routed based on the destination IP address.

Forced tunneling

A special case is when routes are configured with the destination IP prefix `0.0.0.0/0`. Given the precedence rules described above, this route controls traffic destined for any IP address is not covered by any other rules.

By default, Azure implements a system route directing all traffic matching

`0.0.0.0/0` (and not matching any other route) to the Internet. If you override this route, this traffic is instead directed to the next hop you specify. By using a VPN Gateway as the next hop, you can direct all Internet-bound traffic over your VPN connection to an on-premises network security appliance. This is known as forced tunneling.

Configure user-defined routes using the Azure portal

To configure user-defined routes, the first step is to create a route table resource. From the Azure portal, click **+Create A Resource > Networking > Route Table** to open the **Create Route Table** blade, as shown in [Figure 4-18](#). Select the **Subscription** and **Resource Group** from the drop-down menus, fill in the route table **Name** and specify the route table **Region**, which must be the same Azure region that the subnets use with this route table.

Having created the route table, the next step is to define the routes. Open the route table blade, and under **Settings** click **Routes** to open the list of routes in the route table. Then click **+Add** to open the **Add Route** blade, as shown in [Figure 4-19](#).

Create Route table

Basics Tags Review + create

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Resource group * ⓘ

ExamRef-RG



Create new

Instance details

Region * ⓘ

Canada Central



Name * ⓘ

ExamRef-RouteTable



FIGURE 4-18 The Create Route Table blade in the Azure portal

Add route

ExamRef-RouteTable

Route name *

 ✓

Address prefix * ⓘ

 ✓

Next hop type ⓘ

 ▾

Next hop address * ⓘ

 ✓

Info Ensure you have IP forwarding enabled on your virtual appliance. You can enable this by navigating to the respective network interface's IP address settings.

FIGURE 4-19 The Add Route Blade in the Azure portal

Repeat this process for each custom route in the route table. The list of routes in the route table will be shown in the route table blade, as shown in [Figure 4-20](#).

Name	↑↓ Address prefix	↑↓ Next hop type
VNet3-Route	10.3.0.0/16	10.2.20.4

FIGURE 4-20 The list of routes in the route table blade in the Azure portal

The final step is to specify which subnets this route table should be associated with. This can be configured either from the subnet, or from the route table. In the latter case, from the route table blade under **Settings**, click **Subnets** to open the list of subnets associated with the route table. Click **+Associate** to open the **Associate Subnet** blade, as shown in [Figure 4-21](#).

Home >

ExamRef-RouteTable | Subnets

Route table

Search (Ctrl+ /) Associate

Overview Search subnets

Activity log Name ↑ Address range ↑ Virtual network ↑

Access control (IAM) No results.

Tags

Diagnose and solve problems

Settings

Configuration

Routes

Subnets

Properties

Locks

Export template

Support + troubleshooting

Effective routes

New support request

Associate subnet

ExamRef-RouteTable

Virtual network: VNet1

Subnet: default

FIGURE 4-21 The Associate Subnet blade for a route table, in the Azure portal

After creating the subnet association, the route table blade will show a list of associated subnets as shown in [Figure 4-22](#).

Name ↑↓	Address range ↑↓	Virtual network ↑↓	Security group ↑↓
default	10.0.0.0/24	VNet1	.

FIGURE 4-22 The list of subnets in the route table blade in the Azure portal

To see the effective routes for a given network interface, navigate to the network interface blade in the Azure portal and then click **Effective Routes** to open the **Effective Routes** blade, as shown in [Figure 4-23](#).

examref913 | Effective routes

Network interface

Search (Ctrl+ /)

<<

Download Refresh



Overview



Activity log



Access control (IAM)



Tags

Scope

Network interface (examref913)

Associated route table: ①

ExamRef-RouteTable

Settings

Effective routes



IP configurations

Source ↑↓	State ↑↓	Address Prefixes ↑↓	Next Hop Type ↑↓	Next Hop Type IP Address ↑↓	User Defined Route Name ↑↓
-----------	----------	---------------------	------------------	-----------------------------	----------------------------



DNS servers

Default	Active	10.0.0.0/16	Virtual network	-	-
---------	--------	-------------	-----------------	---	---



Network security group

Default	Active	10.1.0.0/16	VNet peering	-	-
---------	--------	-------------	--------------	---	---



Properties

Default	Active	0.0.0.0/0	Internet	-	-
---------	--------	-----------	----------	---	---



Locks

Default	Active	10.0.0.0/8	None	-	-
---------	--------	------------	------	---	---



Export template

Default	Active	100.64.0.0/10	None	-	-
---------	--------	---------------	------	---	---

Support + troubleshooting

Default	Active	192.168.0.0/16	None	-	-
---------	--------	----------------	------	---	---



Effective security rules

Default	Active	25.33.80.0/20	None	-	-
---------	--------	---------------	------	---	---



Effective routes

Default	Active	25.41.3.0/25	None	-	-
---------	--------	--------------	------	---	---



New support request

User	Active	10.3.0.0/16	None	10.2.204	VNet3-Route
------	--------	-------------	------	----------	-------------

FIGURE 4-23 The list effective routes for the examref913 network interface

Configure endpoints on subnets

Service endpoints are a mechanism to integrate Azure PaaS services into your virtual network and access them through a Microsoft Azure backbone network instead of over the Internet. Service endpoints prevent the exposure of data and services to Internet. Service endpoints can be enabled on subnets, and you can also add service endpoints to multiple subnets from the virtual network settings. As of the time this book was published, service Endpoints are supported for the following services:

- Azure Storage
- Azure SQL Databases
- Azure Synapse Analytics
- Azure SQL Data Warehouse
- Azure Database for MySQL & MariaDB
- Azure Cosmos DB
- Azure Key Vault
- Azure Service Bus
- Azure Event Hubs
- Azure Data Lake Store (Gen 1 only)
- Azure App Service
- Azure Container Registry (Public Preview)

You can create service endpoints either using Azure Portal or using command-line tools such as PowerShell or Azure CLI. For example, let's assume the virtual network ExamRef-VNet and subnet Apps is already created. (See "[Create and configure virtual networks, including peering](#)" earlier in this chapter for detailed steps.) From the virtual network, click the **Service Endpoints** option on left under **Settings**, as shown in [Figure 4-24](#). Then click **+Add > Microsoft.Storage** or the select the appropriate Service from the drop-down menu. Select **Apps** from the **Subnets** drop-down menu. Finally, click **Add**.

ExamRef-VNet | Service endpoints

Virtual network | Directory: Microsoft

Search (Ctrl+ /) Add

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Address space Connected devices Subnets DDoS protection Firewall Security DNS servers Peerings Service endpoints (selected) Private endpoints Properties Locks

Monitoring

Alerts Metrics Diagnostic settings Logs

Add Filter service endpoints

Service	Subnet	Status
No service endpoints.		

Add service endpoints

Service * Microsoft.Storage

Service endpoint policies 0 selected

Subnets * Apps

Info With service endpoints, the source IP address for service traffic from this subnet will switch from using public IPv4 addresses to using private IPv4 address. Existing IP firewall rules using Azure public IP addresses will stop working with this switch. Please ensure IP firewall rules allow for this switch before setting up service endpoints. You may also experience temporary interruption to service traffic from this subnet while configuring service endpoints.

Add

FIGURE 4-24 Add service endpoints to subnet

You can create multiple service endpoints for the supported Azure services on a given subnet.

Configure private endpoints

A private endpoint establishes a private connection between any of the supported Azure services and your virtual network. For example, if you create a private endpoint for your storage account, then it provides a secure channel between your VNet and your storage account. The private endpoint gets a private IP address from your VNet address space. The connectivity for private endpoint takes place using private link. Once you create a private endpoint for a service, then a consent request is sent for approval by the service owner. Once it is approved, the private endpoint can be used to establish connectivity to the defined service (such as a storage account).

To configure private endpoint, let's assume you have created **ExamRef-VNet** and **ExamRef-VM**. To create a private endpoint for a particular PaaS service (such as a web app), we need to first deploy the web app with a minimum of the PremiumV2 tier. Let's assume you have created an examref web app. To create a private endpoint, go to the Azure homepage, click **+Create A Resource**, search for a private endpoint, and then click **Create**.

You need to select the **Subscription**, **Resource Group**, and **Region** for creating a private endpoint, as shown in [Figure 4-25](#). You also need to specify the private endpoint **Name**. Then click **Next: Resource**.

Create a private endpoint ...

1 Basics **2 Resource** **3 Configuration** **4 Tags** **5 Review + create**

Use private endpoints to privately connect to a service or resource. Your private endpoint must be in the same region as your virtual network, but can be in a different region from the private link resource that you are connecting to. [Learn more](#)

Project details

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Resource group * ⓘ

ExamRef-RG

[Create new](#)

Instance details

Name *

ExamRef-PE

Region *

(Canada) Canada Central

[< Previous](#)

[Next : Resource >](#)

FIGURE 4-25 Create a private endpoint—Basics blade

On the next screen, you need to select the resource details for the private endpoint **Subscription**, **Resource Type**, **Resource**, and **Target Sub-Resource**, as shown in [Figure 4-26](#). Then click **Next: Configuration >** at the bottom.

Create a private endpoint

✓ Basics

2 Resource

3 Configuration

4 Tags

5 Review + create

Private Link offers options to create private endpoints for different Azure resources, like your private link service, a SQL server, or an Azure storage account. Select which resource you would like to connect to using this private endpoint. [Learn more](#)

Connection method (i)

Connect to an Azure resource in my directory.

Connect to an Azure resource by resource ID or alias.

Subscription * (i)

Visual Studio Ultimate with MSDN

Resource type * (i)

Microsoft.Web/sites

Resource * (i)

examref01

Target sub-resource * (i)

sites

< Previous

Next : Configuration >

FIGURE 4-26 Create A Private Endpoint—Resource blade

On the next screen, you need to specify the virtual network details along with the **Subnet** and **Private DNS Zone** integration options, as shown in [Figure 4-27](#). Then click **Review + Create** at the bottom to create the private endpoint.

Create a private endpoint

✓ Basics ✓ Resource

3 Configuration

4 Tags

5 Review + create

Networking

To deploy the private endpoint, select a virtual network subnet. [Learn more](#)

Virtual network * ⓘ

ExamRef-VNet

Subnet * ⓘ

Apps (10.1.0.0/24)

i If you have a network security group (NSG) enabled for the subnet above, it will be disabled for private endpoints on this subnet only. Other resources on the subnet will still have NSG enforcement.

Private DNS integration

To connect privately with your private endpoint, you need a DNS record. We recommend that you integrate your private endpoint with a private DNS zone. You can also utilize your own DNS servers or create DNS records using the host files on your virtual machines. [Learn more](#)

Integrate with private DNS zone

Yes

No

Configuration name

Subscription

Private DNS zone

privatelink-azurewebs...

Visual Studio Ultimate with MS...

privatelink.azurewebsites.net

[Review + create](#)

[< Previous](#)

[Next : Tags >](#)

FIGURE 4-27 Create A Private Endpoint—Configuration blade

Configure Azure DNS

This section describes how Azure DNS is configured to host Internet-facing domains. We start with a summary of how the domain name system works because understanding DNS is a prerequisite to understanding Azure DNS.

How DNS Works

To properly understand the various DNS services and features available in Azure, it is first necessary to understand how the domain name system works. In particular, it is important to understand the different roles played by recursive and authoritative DNS servers, and how a DNS query is routed to the correct DNS name servers using DNS delegation.

First, it's important to understand the distinction between a domain name, and a DNS zone. The Internet-facing domain name system is a single global name hierarchy. A domain name is just a name within that hierarchy. Owning a domain name gives you the legal right to control the DNS records within that name, and any sub-domains of that name.

You purchase a domain name from a domain name registrar. The registrar then lets you control which name servers receive the DNS queries for that domain, by letting you configure the NS records for the domain.

A DNS zone is the representation of a domain name in an *authoritative DNS server*. It contains the collection of DNS records for a given domain name. The service hosting the DNS zone lets you manage the DNS records within the zone, and hosts the data on authoritative name servers, which answers DNS queries with DNS responses based on the configured DNS records.

In Azure, you can purchase domain names using the App Service Domains service. DNS zone hosting is provided by Azure DNS.

The DNS settings on the user's device point to a recursive DNS server, also sometimes known as a local DNS service (or LDNS), or simply as a DNS resolver. The recursive DNS service is typically hosted by your company (if you're at work) or by your ISP (if you're at home). There are also public recursive DNS services available, such as Google's 8.8.8.8 service. The recursive DNS service doesn't host any DNS records, but it allows your device

to off-load most of the work associated with resolving DNS queries.

To understand the role of recursive and authoritative DNS servers, consider [Figure 4-28](#), which describes the DNS resolution process for a single DNS query, www.contoso.com.

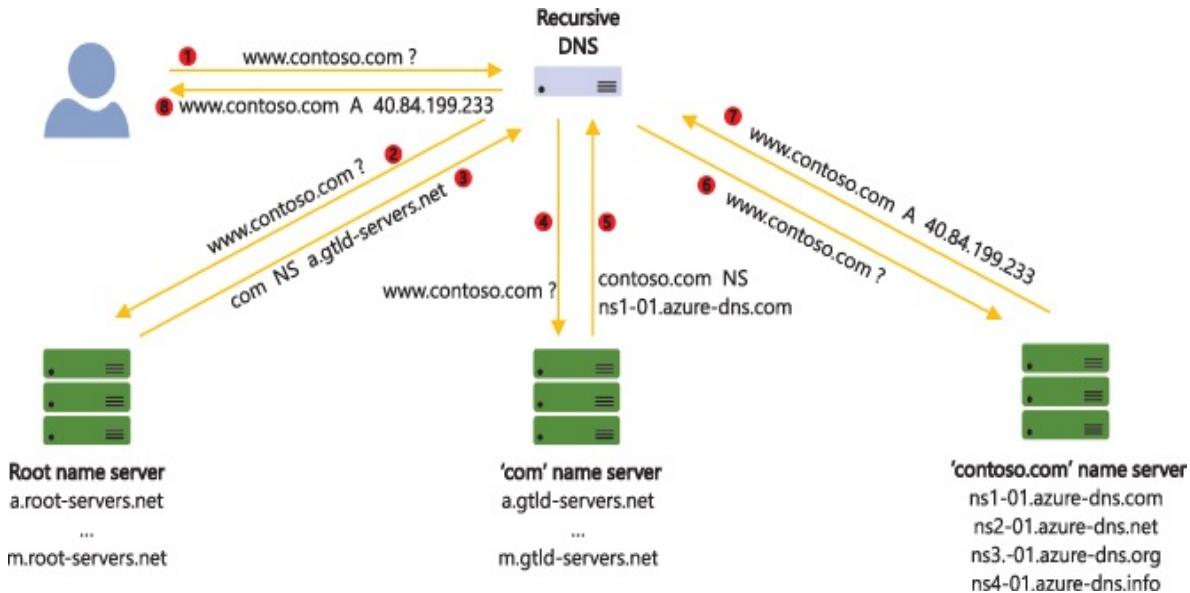


FIGURE 4-28 The DNS Resolution Process

This resolution process is described here:

1. Your PC makes a DNS query to its locally configured recursive DNS server. This query is simply a packet sent over UDP port 53, although TCP can also be used (typically when responses are too big to fit in a UDP packet).
2. Let's assume the recursive DNS server has just been switched on, so there is nothing in its cache. It passes the query to one of the root name servers (the addresses of the root name servers are pre-configured). The root name servers are authoritative name servers—they host the actual DNS records for the root zone. A zone is simply the data representing a node in the DNS hierarchy.
3. The root name servers don't know anything about the `contoso.com` DNS zone. They do, however, know where you can find the `.com` zone. So, they return a DNS record of type NS, which tells the recursive DNS server where to find the `.com` zone.

4. The recursive server tries again, this time calling the com name servers. Again, these are authoritative name servers, this time for the com zone.
5. These name servers don't recognize www.contoso.com, but they do have NS records that define where the *contoso.com* DNS zone can be found.
6. The recursive server tries again, this time calling the authoritative *contoso.com* name servers.
7. These servers *are* authoritative for the *contoso.com* DNS zone. And there is a record on these servers matching the www record name. The server does recognize the www.contoso.com query name and returns the A record response that maps this name to an IP address.
8. The recursive server then returns this result back to the client.

The recursive DNS server can also follow a chain of CNAME records (which map one DNS name to another name). And the recursive DNS server also caches the responses it receives, so that it can respond more quickly next time. The duration of the cache is determined by the TTL (time-to-live) property of each DNS record.

The domain name system is a distributed system, where one set of servers can refer queries to another set using NS records. The process we've just seen to map a query name to a result—perhaps via a long chain of authoritative DNS servers—is called “DNS name resolution.”

The NS records tell clients on the Internet where to find the name servers for a given DNS zone. The NS records for a DNS zone are configured in the parent zone, and a copy of the records is also present in the child zone. Setting up these NS records is called delegating a DNS domain.

A fully qualified domain name (FQDN) is a domain name containing all components all the way up to the root zone. Strictly speaking, a fully qualified name ends with a “.” (for example, *www-dot-contoso-dot-com-DOT*), which represents the root zone, although by convention, the trailing period is often omitted.

Reverse DNS is the ability to map an IP address to a name (as opposed to name to IP address, which is what normal DNS provides). Some applications use reverse DNS as a weak form of authentication. For example, it's commonly used in email spam-scoring algorithms.

Reverse DNS lookups use a DNS hierarchy that is completely independent of

the forward lookups. The reverse lookup for www.contoso.com does not sit in the *contoso.com* zone. Instead, it sits in a separate DNS zone hierarchy based on reversed IP addresses. For example, suppose *www.contoso.com* resolves to IP address 1.2.3.4. Typically, the reverse lookup for the IP address 1.2.3.4 will be a record named 4 in the DNS zone 3.2.1.in-addr.arpa, giving a FQDN of 4.3.2.1.in-addr.arpa (notice the reversed IP address.)

Reverse DNS lookup zones are controlled by whomever owns the IP subnet. The reverse DNS lookup zone for an IP block you own can be hosted in Azure DNS. Public IP addresses in Azure reside in Microsoft-owned IP blocks, which means the reverse DNS lookups use Microsoft-managed reverse DNS lookup zones.

There's nothing in the domain name system to ensure the reverse lookup maps to the same name as was used in the forward lookup. That's achieved simply by the correct configuration in both forward and reverse lookup zones.

DNS services in Azure

There are several DNS-related services and features in Azure—an overview of each is given below. The first three items are Azure services, which you consume by creating service-specific resources that you will be billed for. The remaining three items are Azure features, which you configure using settings on other resource types, such as a virtual network, public IP address, or network interface.

- **Azure DNS.** Allows you to host your DNS domains in Azure. It provides the ability to create and manage the DNS records for your domain and provides name servers, which answer DNS queries for your domain from other users on the Internet. Azure DNS also supports private DNS zones, which are used for intranet-based name resolution for VM to VM lookups, including support for some scenarios not supported by the Azure-provided DNS service, which we'll cover shortly. Private DNS zones are currently in preview.
- **Azure Traffic Manager.** An intelligent DNS service that uses DNS to implement global traffic management. Where Azure DNS always provides the same DNS response to any given DNS query, in Azure Traffic Manager the same query may result in one of several possible responses, depending on a number of factors which you control, such as where the end-user is located or which of your service endpoints is currently

available. This enables you to route traffic intelligently between Azure regions or between Azure deployments and on-premises deployments. Understanding Traffic Manager beyond the scope of the AZ-104 exam.

- **App service domains.** Allows purchasing of domain names, which can then be hosted in Azure DNS. This service is integrated with Azure App Service, but can be used for any domain registration, even if App Service is not being used.
- **Azure-provided DNS.** Sometimes called Internal DNS, it allows the VMs in your virtual network to find each other, using DNS queries based on the hostname of each VM. The DNS queries are internal (private) to the virtual network.
- **Recursive DNS.** A service provided by Azure for DNS name resolution from your Azure VMs or other Azure services. You can also configure your VMs to use your own DNS server instead. This is sometimes informally called bring your own DNS. This is common when joining your VMs to a domain controller.
- **Reverse DNS.** Provides the ability to configure the reverse DNS lookup for an Azure-assigned public IP address. (Reverse DNS lookup zones for IP blocks you own can be hosted in Azure DNS.)

Creating and delegating a DNS Zone to Azure DNS

A DNS zone is a resource in Azure DNS. Creating a DNS zone resource allocates authoritative DNS name servers to host the DNS records for that zone. Azure DNS can then be used to manage those DNS records. DNS queries directed to those DNS name servers receive a DNS response based on the DNS records configured at that time.

You do not have to own the corresponding domain name before creating a DNS zone in Azure DNS. You can create a DNS zone with any name, except for names on the public suffix list (see <https://publicsuffix.org/>). You can also create more than one DNS zone resource with the same DNS zone name, so long as they are in different resource groups. In this case, the DNS zones will be allocated to separate DNS name servers, so no conflict arises.

You can test your DNS records by directing DNS queries directly to the assigned DNS name servers for your zone. For general use, however, your DNS zone should be delegated from the parent zone. This requires you to own the

corresponding domain name.

Before you can delegate your DNS zone to Azure DNS, you first need to know the names of the name servers assigned to your zone. These can be obtained using the Azure portal, PowerShell, or CLI after the DNS zone resource has been created. You can't predict in advance which name server pool will be assigned to your DNS zone. You need to create the DNS zone, and then check.

The assigned name servers will vary between zones, so if you're setting up multiple zones in Azure DNS you need to check the name servers on each one. Don't assume that the name servers will be the same across all your zones.

Each domain name registrar has their own DNS management tool allowing you to set the name server (NS) records for a domain. In the registrar's DNS management page, edit the NS records and replace the NS records with the ones Azure DNS assigned.

When delegating a domain to Azure DNS, you must use the name server names provided by Azure DNS. You should always use all four name server names, regardless of the name of your domain. Domain delegation does not require the name server name to match your domain name.

Note Delegating DNS Zones to Azure DNS

When delegating a domain to Azure DNS, do not use DNS glue records to point to the Azure DNS name server IP addresses directly. A glue record is a DNS server record that is not authoritative for the zone and is used to avoid a condition of impossible dependencies for a DNS zone. These IP addresses might change in the future. Delegations using name server names in your own zone—sometimes called vanity name servers—are not currently supported in Azure DNS.

Azure DNS treats child zones as entirely separate zones. Therefore, delegating a child zone follows the same process as delegating the parent zone:

1. Create the child zone resource.
2. Identify the name servers for the child zone. These will be different to the name servers assigned to the parent zone.
3. Create NS records in the parent zone to delegate the child zone. The name

of the NS records should be the child zone name (excluding the parent zone name suffix), and the RDATA in the NS records should be the child zone name servers.

Note Delegating Child DNS Zones to Azure DNS

When you delegate a child zone, any existing name servers in the parent zone that match the child zone name will become hidden. You'll still see them in the Azure portal, but they won't resolve from the name servers because the delegation to the child zone will take precedence. To avoid this issue, before delegating the child zone, you should check for any records that will be hidden and replicate them into the child zone. This applies with any DNS service, not just Azure DNS.

Managing DNS records in Azure DNS

Each record in the domain name system includes the following properties:

- **Name.** The name of the DNS record is combined with the name of the DNS zone to form the fully qualified domain name (FQDN). For example, the record `www` in zone `contoso.com` corresponds to the FQDN www.contoso.com.
- **Type.** The type of DNS record determines what data is associated with the record and what purpose it is used for. A list of record types supported by Azure DNS is provided in [Table 4-5](#).
- **TTL.** The TTL (or Time-to-Live) tells recursive DNS servers how long a DNS record should be cached.
- **RDATA.** The data returned for each DNS record. The type of data returned depends on the DNS record type. For example, an `A` record will return an IPv4 address, whereas a `CNAME` record returns another domain name.

The collection of records in a DNS zone with the same name and the same type is called a resource record set. (These collections are also referred to as “RRSets” and as “record sets” in Azure DNS). Records in Azure DNS are managed using record sets. Record sets are a child resource of the DNS zone and can contain up to 20 individual DNS records. The name, type, and TTL are configured on the record set, and the RDATA is configured on each DNS record

within the record set.

To create a DNS record set at the root (or *apex*) of a DNS zone, use the record set name `@`. For example, the record set named `@` in the `contoso.com` zone will resolve against queries for `contoso.com`. You can also use an asterisk (`*`) in the record set name to create wildcard records (subject to DNS wildcard matching rules).

Azure DNS supports all commonly used DNS record types. The full list of supported record types—together with a description of each—is provided in [Table 4-5](#).

TABLE 4-5 DNS Record Types in Azure DNS

D Re co rd Ty pe	Remarks
NS	
Re	
co	
rd	
Ty	
pe	
A	Used to map a name to an IPv4 address.
AA	Used to map a name to an IPv6 address.
AA	
CA	Used to specify which certificate authorities can issue certificates for a domain. Note that <code>CAA</code> records are not currently available in the Azure portal, so they must be configured using the Azure CLI or Azure PowerShell.
CN	Provides a mapping from one DNS name to another. The DNS standards do not allow <code>CNAME</code> records at the zone apex. In addition, you cannot create a <code>CNAME</code> record with the same name as a record of any other record type, and <code>CNAME</code> record sets only support a single DNS record rather than a list of records. These are DNS RFC constraints, not Azure DNS limitations.
M	Used for mail server configuration.
X	

NS An NS record set at the zone apex containing the name servers for the DNS zone is required by the DNS standards. This is created for you when the DNS zone is created. It can be edited, for example to add additional records when co-hosting a DNS zone with more than one provider, but not deleted.

You can create additional NS record sets to delegate child zones.

PT Used for reverse DNS lookups in reverse lookup zones.

R

SO An SOA record is required at the apex of every zone. This is created and **A** deleted with the DNS zone resource.

SR SRV records are used for service discovery for a wide range of services, **V** from Kerberos to Minecraft to the Session Initiation Protocol used for Internet telephony.

Note that the `Service` and `Protocol` parameters are specified as part of the record set name, such as `_service._protocol.media.contoso.com`.

Some DNS services prompt you to enter these values separately and then merge them to form the record set name. With Azure DNS, you need to specify them as part of the record set name, but they are not entered separately.

TX Used for a wide range of applications, including email Sender Policy **T** Framework (SPF).

Note SPF Records

Sender Policy Framework (SPF) records are used to identify legitimate mail servers for a domain and help prevent spam. The SPF record type was deprecated by RFC7208, which states that the TXT record type should be used for SPF records.

Alias records

Azure DNS offers integration with other services hosted in Azure via Alias records.

With conventional DNS records, you explicitly specify the target, such as the IP address of an A record. If the IP address changes, you need to update the DNS record accordingly.

Alias records allow you to define the target of the DNS record implicitly by referencing another Azure resource. The value of the DNS record is populated automatically based on the resource it references and is updated automatically if that resource changes.

Alias records can reference three different resource types:

- **An A or AAAA.** These records can reference a public IP address, of type IPv4 or IPv6, respectively.
- **A, AAAA, or CNAME.** These records can reference a Traffic Manager profile. This exposes the dynamic, traffic-managed name resolution of the Traffic Manager directly within a record in your DNS domain. Prior to this feature, you had to create a CNAME record from your domain to a record in the trafficmanager.net domain provided by Azure Traffic Manager.
- **An A, AAAA or CNAME.** These records can also reference another record in the same DNS zone. This lets you create synchronized records with ease.

Alias records are a very useful way to address a number of scenarios.

- First, Alias records allow you to avoid orphaned DNS records. A common problem with DNS systems is that records are not cleaned up when the services they reference are deleted. The DNS record is left dangling. With Alias records, the DNS record no longer resolves once the underlying service is deleted.
- Second, as we have already discussed, by updating automatically when underlying resources change, Alias records reduce your management overhead and help you avoid accidental application downtime.
- Third, because Alias records enable you to avoid using a CNAME record when using a vanity domain name with Azure Traffic Manager, they enable you to implement a traffic-managed record at the apex of your domain.

Creating DNS zones and DNS records using the Azure portal

To create a DNS zone, click **+Create A Resource > Networking > DNS Zone** to open the **Create DNS Zone** blade. Fill in the blade by specifying the DNS domain **Name** as the DNS zone resource name, and selecting your **Resource Group**, as shown in [Figure 4-29](#).

Note DNS Zones and Azure Region

When creating a DNS zone, the location field only specifies the resource group location. It does not apply to the DNS zone resource itself, which is global rather than regional.

Once the DNS zone has been created, open the DNS zone blade. The Azure DNS name servers assigned to the zone are listed in the essentials panel, as highlighted in [Figure 4-30](#).

Create DNS zone

Basics Tags Review + create

A DNS zone is used to host the DNS records for a particular domain. For example, the domain 'contoso.com' may contain a number of DNS records such as 'mail.contoso.com' (for a mail server) and 'www.contoso.com' (for a web site). Azure DNS allows you to host your DNS zone and manage your DNS records, and provides name servers that will respond to DNS queries from end users with the DNS records that you create. [Learn more](#).

Project details

Subscription *

Visual Studio Ultimate with MSDN



Resource group *

ExamRef-RG



[Create new](#)

Instance details



This zone is a child of an existing zone already hosted in Azure DNS ⓘ

Name *

examref.com



Resource group location ⓘ

Canada Central



FIGURE 4-29 Creating a DNS zone using the Azure portal

Home >

examref.com ↗

DNS zone

Search (Ctrl+I) | [Record set](#) [Child zone](#) [Move](#) [Delete zone](#) [Refresh](#)

Overview	Resource group (change) : examref.rg	Name server 1 : ns1-05.azure-dns.com.			
Activity log	Subscription (change) : Visual Studio Ultimate with MSDN	Name server 2 : ns2-05.azure-dns.net			
Access control (IAM)	Subscription ID : 00b72028-9dce-4729-9b2a-a10e92054447	Name server 3 : ns3-05.azure-dns.org.			
Tags	Tags (change) : Click here to add tags	Name server 4 : ns4-05.azure-dns.info.			
^					
Settings	You can search for record sets that have been loaded on this page. If you don't see what you're looking for, you can try scrolling to allow more record sets to load.				
Properties	<input type="text"/> Search record sets				
Locks	Name	Type	TTL	Value	Alias resource type
Export template	@	NS	172800	ns1-05.azure-dns.com. ns2-05.azure-dns.net. ns3-05.azure-dns.org. ns4-05.azure-dns.info.	
Monitoring					
Alerts				Email: azuredns-hostmaster.micros...	
Metrics				Host: ns1-05.azure-dns.com. Refresh: 3600	
Support + troubleshooting				Retry: 300 Expire: 2419200 Minimum TTL: 300	
New support request				Serial number: 1	

FIGURE 4-30 The DNS zone blade, highlighting the Azure DNS name servers assigned to this zone

To set up DNS delegation for the DNS zone, these name servers must be listed in the corresponding NS records in the parent zone. If the domain name was purchased using the Azure App Service Domains service, this will be done automatically. Otherwise, this must be configured at the DNS registrar where the domain name was purchased.

To create a DNS record in a new record set, click **+Record Set** to open the **Add Record Set** blade. If there is an existing record with the same name and type as the record you want to create, you should instead click the existing record set and add the new record there. To create a pair of A records with name `www` (giving the fully qualified domain name `www.examref.com`), fill in the blade with the following values, as shown in [Figure 4-31](#).

- **Name.** `www`
- **Type.** A
- **Alias Record Set.** No
- **TTL.** 1 hour (or choose your own value)
- **IP Addresses.** Enter A record IP addresses, one for each DNS record in the record set.

Suppose now you want to create a DNS record at the zone apex (so the fully qualified domain name is simply the DNS zone name `examref.com`), pointing to a dynamically allocated public IP address. Click **+Add Record Set** again and complete the **Add Record Set** blade with the following settings, as shown in [Figure 4-32](#).

- **Name.** `@` (This is a DNS convention for records at the zone apex.)
- **Type.** A.
- **Alias Record Set.** Yes.
- **Choose Subscription.** Choose the subscription containing the public IP address.
- **Azure Resource.** Choose the public IP address resource.
- **TTL.** 1 hour (or choose your own value).

Add record set

X

examref.com

Name

WWW



.examref.com

Type

A



Alias record set ⓘ

Yes No

TTL *

1

TTL unit

Hours



IP address

23.34.45.56



123.134.145.156



0.0.0.0



FIGURE 4-31 The Add Record Set blade

Add record set

X

examref.com

Name

@



.examref.com

Type

A



Alias record set (i)

- Yes No

Alias type

- Azure resource Zone record set

Choose a subscription *

Visual Studio Ultimate with MSDN



Azure resource *

ExamRef-ip



TTL *

1

TTL unit

Hours



FIGURE 4-32 The Add Record Set blade for an Alias record set

Configure custom DNS settings

When a virtual machine connects to a virtual network, it receives its IP address via DHCP. As part of that DHCP exchange, DNS settings are also configured in the VM. By default, VMs are configured to use Azure's recursive DNS servers. These provide name resolution for Internet-hosted domains, plus private VM-to-VM name resolution within a virtual network.

The hostname of the VM is used to create a DNS record mapping to the private IP address of the VM. You specify the hostname—which is simply the VM name—when you create the virtual machine. Azure specifies the DNS suffix, using a value that is unique to the virtual network. These suffixes end with internal.cloudapp.net. The hostname and DNS suffix together form the unique fully qualified domain name.

Name resolution for these DNS records is private—they can only be resolved from within the virtual network. The DNS suffix is configured as a lookup suffix within each VM, so names can be resolved between VMs within the virtual network using the hostname only.

This built-in DNS service uses the IP address: 168.63.129.16. This is a special static IP address that is reserved by the platform for this purpose. This IP provides both the authoritative DNS service for Azure-provided DNS as well as Azure's recursive DNS service, which is used to resolve Internet DNS names from Azure VMs. This IP is used for other things as well, such as health problems from Azure Load Balancer, heartbeat messages for PaaS roles, and so on.

Bring your own DNS

Alternatively, you can configure your own DNS settings, which will be configured during the DHCP exchange on the VMs instead. This enables you to specify your own DNS servers, either in Azure or running on-premises. With your own DNS servers, you can support any DNS scenario, including scenarios not supported by the Azure-provided service. Example scenarios requiring you to use your own DNS servers include name resolution between VMs in different virtual networks, name resolution between on-premises resources and Azure virtual machines, reverse DNS lookup of internal IP addresses, and name

resolution for non-Internet-facing domains, such as domains associated with Active Directory.

You should not specify your own DNS settings within the VM itself because the platform is unaware of the settings you have chosen. Instead, Azure provides configuration options within the virtual network settings. These DNS server settings are at the virtual network level and apply to all VMs in the virtual network.

You can also specify VM-specific DNS server settings within each network interface. This takes precedence over settings at the virtual network level. Where multiple VMs are deployed in an availability set, setting DNS servers at the network interface, all VMs in the availability set are updated. The DNS servers applied are the union of the network interface-level DNS servers from across the availability set.

Note DNS Name Server Settings

Custom DNS settings can be configured at the VNet level, and the network interface level, but not at the subnet level. To use specific settings for an individual subnet, you must configure those settings on each network interface in the subnet.

You can use these DNS settings to direct your VMs' DNS queries to any DNS servers you choose. They can point to IP addresses of on-premises servers, such as an Active Directory Domain Controller or network appliance, a DNS service running in an Azure Virtual Machine, or anywhere else on the Internet.

If you use your own DNS servers, those servers will need to offer a recursive DNS service, otherwise name resolution for Internet domains from your virtual machines will break. If you point the DNS settings directly at an Internet-based recursive DNS service, such as Google 8.8.8.8, then you will not be able to perform VM-to-VM lookups.

Note Restart Virtual machines when changing DNS settings

If you make changes to the DNS settings at the virtual network level, any affected virtual machines must restart to pick up the new settings. If you make changes to DNS settings and the network interface level, the affected VM (or VMs across the availability set, if used) will restart automatically

to pick up the new settings.

One challenge when using your own DNS servers is that you will need to register each VM in your DNS service. To do this, you can configure the DNS service to accept Dynamic DNS queries, which the VM will send when it boots. This allows the VMs to register with the DNS server automatically. A problem with this approach is that the DNS suffix in the Dynamic DNS query must match the DNS zone name configured on the DNS server, and Azure does not support configuring the DNS suffix via the Azure platform settings. As a workaround, you can configure the correct DNS suffix within each VM yourself, using a start-up script.

Configure custom DNS settings using the Azure portal

To configure the DNS servers on a VNet, open the virtual network blade, and then click **DNS Servers** under **Settings**, as seen in [Figure 4-33](#). You can then enter the DNS servers you want this VM to use. After saving your changes, you need to restart the VMs in the VNet to pick up the changes.

The steps to configure the DNS servers on an individual VM are similar to previous step. Open the blade for the VM's network interface, and then click **DNS Servers** under **Settings**. You can then enter the DNS servers you want this VM to use. Note that VMs in an availability set will adopt the union of DNS servers from network interfaces across the availability set. After saving your changes, your VM (or VMs in the availability set) will automatically restart to pick up the changes.

Home >

VNet1 | DNS servers

Virtual network

Search (Ctrl+ /)



Save Discard

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Address space

Connected devices

Subnets

DDoS protection

Firewall

Security

DNS servers

Peerings

Virtual machines within this virtual network must be restarted to utilize the updated DNS server settings.

DNS servers (i)

Default (Azure-provided)

Custom

10.0.0.25

...

10.0.0.125

...

Add DNS server

...

FIGURE 4-33 Custom DNS servers for a virtual network configured using Azure portal

Configure private DNS zones

In addition to supporting Internet-facing DNS domains, Azure DNS also supports private DNS domains. This provides an alternative approach to name resolution within and between virtual networks.

By using private DNS zones, you can use your own custom domain names—including the DNS suffix, rather than the Azure-provided DNS suffix—without the overhead or complexity of running your own DNS servers.

The service supports automatic registration of VMs into the private zone, but only from a single virtual network, called the registration VNet. This must be registered with the DNS zone before any VMs are created.

If you want to resolve VM names from multiple virtual networks, the VMs in any other networks must be registered with the service manually (or via a custom automation). Name resolution between VNets is independent of connectivity between VNets, so peering your virtual networks or setting up a VNet-to-VNet connection is not required.

When name resolution is supported by virtual networks, these are called resolution VNets. The zone name is not registered with the VMs as a DNS search suffix, so you will need to register it yourself or use fully qualified domain names in your DNS queries.

Create private DNS zones using Azure portal

To create a private DNS zone, click **+Create A Resource > Networking > Private DNS Zone** to open the **Create Private DNS Zone** blade. Fill in the settings by specifying the DNS domain **Name** as the DNS zone resource name and selecting your **Resource Group**, as shown in [Figure 4-34](#).

Create Private DNS zone

Basics Tags Review + create

A Private DNS zone provides name resolution services within virtual networks. A Private DNS zone is accessible only from the virtual networks that it is linked to and can't be accessed over internet. For example you can create a Private DNS zone named contoso.com and then create DNS records like www.contoso.com in this zone. You can then link the zone to a one or more virtual networks. [Learn more.](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Visual Studio Ultimate with MSDN



Resource group *

ExamRef-RG



[Create new](#)

Instance details

Name * ⓘ

privateexamref.com



Resource group location ⓘ

Canada Central



i You can link virtual networks to this Private DNS zone after zone has been created.

FIGURE 4-34 Creating a private DNS zone using the Azure portal

With private DNS zone, you can create virtual network links by choosing **Virtual Network Links**, and then clicking **+ Add**, as shown in [Figure 4-35](#).

Home >

privateexamref.com | Virtual network links

Private DNS zone

Search (Ctrl+ /) Add Refresh

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

No results.

Virtual network links Properties Locks Export template

Link Name	Link status	Virtual network	Auto-Registration
No results.			

FIGURE 4-35 Virtual network links for a private DNS zone

You only need to supply the **Link Name**, **Subscription**, and **Virtual Network Name**, as shown in [Figure 4-36](#). You can also select the **Enable Auto Registration** checkbox, which will automate the creation of DNS records in the Private DNS zone for the virtual machines which are connected to the virtual network.

Add virtual network link

privateexamref.com

Link name *

ExamRef-Link



Virtual network details



Only virtual networks with Resource Manager deployment model are supported for linking with Private DNS zones.
Virtual networks with Classic deployment model are not supported.

I know the resource ID of virtual network (i)

Subscription * (i)

Visual Studio Ultimate with MSDN



Virtual network *

ExamRef-VNet (ExamRef-RG)



Configuration

Enable auto registration (i)

FIGURE 4-36 Add virtual network link for a private DNS zone

Once created, a **Virtual Network Link** appears on the right screen. Create private DNS zones using Azure PowerShell or the Azure CLI.

Skill 4.2: Secure access to virtual networks

Network security groups (NSGs) allow you to control which network flows are permitted into and out of your virtual networks and virtual machines. Each NSG contains lists of inbound and outbound rules, which give you fine-grained control over exactly which network flows are allowed or denied.

This section covers how to:

- [Create security rules](#)
- [Associate a network security group \(NSG\) to a subnet or network interface](#)
- [Evaluate effective security rules](#)
- [Implement Azure Firewall](#)
- [Implement Azure Bastion Service](#)

Create security rules

A network security group (NSG) is a standalone Azure resource, which acts as networking filter. Each NSG contains a list of security rules. These are used to allow or deny inbound or outbound network traffic, depending on the properties of that traffic such as protocol, IP address, and port. To apply the NSG, it is associated with either a subnet or with a specific VM's network interface.

NSG rules

NSG rules define which traffic flows are allowed or denied by the NSG. [Table 4-6](#) describes the properties of an NSG rule.

TABLE 4-6 NSG properties

Property	Description	Constraints	Considerations
Name	The name of the rule.	Must be unique within the region Must end with a letter, number, or underscore. Cannot exceed 80 characters.	You can have several rules within an NSG, so make sure you follow a naming convention that allows you to identify the purpose of each rule.
Protocol	The network protocol the rule applies to.	TCP, UDP, or *.	Using * as a protocol includes ICMP as well as TCP and UDP. In the Azure portal, select 'Any' instead of '*'.
Source port range(s)	Source port range(s) to match for the rule.	Single port number from 1 to 65535; a port range (example: 1-65535); a list of port or port ranges; or * (for all ports).	The source ports could be ephemeral, so unless your client program is using a specific port, use * in most cases. Try to reduce the number of rules by specifying multiple ports or port ranges in a single rule.
Destination port range	Destination port range(s) to match for the rule.	Single port number from 1 to 65535, port range (such as 1-65535), a list of port or port ranges, or * (for all ports).	Try to reduce the number of rules by specifying multiple ports or port ranges in a single rule.
Source address	Source address	Single IP address (such	Consider using ranges,

ce	prefix(es) or service tag(s) to match for the rule.	as 10.10.10.10), IP subnet (such as 192.168.1.0/24), a service tag, a list of the above, or * (for all addresses).	service tags, and lists to reduce the number of rules.
Destinatio	Destination address prefix(es) or service tag(s) to match for the rule.	Single IP address (such as 10.10.10.10); IP subnet (such as 192.168.1.0/24); a service tag; a list of the above; or * (for all addresses).	The IP addresses of Azure VMs can also be specified implicitly using application security groups.
Direction	Direction of traffic to match for the rule.	Inbound or outbound.	Consider using ranges, default tags, and lists to reduce the number of rules.
Priority	Rules are checked in the order of priority. Once a matching rule is found, no more rules are tested.	Unique Number between 100 and 4096. Uniqueness is only within this NSG.	The IP addresses of Azure VMs can also be specified implicitly using application security groups.
Action	Type of action to apply if the rule matches.	Allow or Deny.	Inbound and outbound rules are processed separately, based on traffic direction.

Note NSG Rule Priority

NSG Rules are enforced based on their priority. Priority values start from 100 and go to 4096 (and from 65001 to 65003 for default rules). Rules will be read and enforced starting with 100 and are followed by 101, 102, and so on. When a rule is found that matches the traffic under consideration, the rule is applied, and all further processing stops—subsequent rules are disregarded.

For example, suppose you had an inbound rule that allowed TCP traffic on any port with a priority of 250 and another that denied TCP traffic on Port 80 with a priority of 125. An inbound TCP connection on port 80 would be denied, since the deny rule has a lower priority value and would be applied before the allow rule is considered.

Service Tags

Many Azure services are accessed via Internet-facing endpoints. These endpoints can change over time, for example as new Azure regions are built. This makes it difficult to use NSG rules to control access to those services—it's hard to identify the list of IP ranges to use, and even harder to keep the list up-to-date.

To address this problem, Azure provides service tags. These are platform-defined shortcuts that map to the IP ranges of various Azure services. The IP ranges associated with each service tag are updated automatically whenever the IP addresses used by the service change.

Service tags are used in NSG rules as a quick and reliable way of creating rules that control traffic to each service. Typically, they are used in outbound rules to control which other Azure services the VMs in a VNet can or cannot access.

Note that service tags control access to the service, but not to a specific resource within that service. For example, a service tag might be used in an NSG rule allowing a VM to connect to Azure storage. This rule cannot control which account in Azure storage the VM will attempt to use.

Service tags are provided for more than 60 Azure services, and the list is growing. Here are some of the most commonly used service tags.

- **VirtualNetwork.** Controls access to the virtual network address space where the NSG is assigned. It refers to the entire virtual network (not just the subnet), plus all connected virtual networks and any on-premises address space connected via Site-to-Site VPN or ExpressRoute (which we discuss in the next Skill section of this course). Note that the network address space of peered virtual networks is only included if the `Allow Virtual Network Access` property is set to `Enabled`.
- **Internet.** Denotes the public Internet address space. This includes the Internet-facing Azure IP address ranges that are used for public IP addresses and Azure platform services.
- **AzureCloud.** Denotes the Azure data center public IP space. This service tag can be scoped to a specific Azure region, such as by specifying `AzureCloud.EastUs`.
- **AzureLoadBalancer.** Denotes the IPs where Azure Load Balancer health probes will originate. Traffic from these addresses should be allowed for any load-balanced VMs. Note that this service tag cannot be used to control traffic coming through the Load Balancer from elsewhere. This traffic can be filtered using the originating source IP, which is not modified as it passes through the Azure Load Balancer
- **AzureTrafficManager.** Performs a similar role for Azure Traffic Manager. It is used to allow traffic from the source IP addresses of Traffic Manager health probes.
- **Storage.** Represents the IP addresses used by the Azure Storage service. As with the Azure Cloud Service Tag, the Storage service tag can be region scoped. For example, you can specify `Storage.WestUS` to only allow access to Storage accounts in the West US region.
- **Sql.** Represents the IP addresses used by the Azure Database for MySQL, Azure Database for PostgreSQL, and Azure Synapse Analytics. This service tag can also be scoped to a specific region.

Default rules

All NSGs have a set of default rules. You cannot add to, edit, or delete these default rules. However, since they have the lowest possible priority, they can be overridden by other rules which you create.

The default rules allow and disallow traffic as follows:

- **Virtual network.** Traffic originating and ending in a virtual network is allowed both in inbound and outbound directions.
- **Internet.** Outbound traffic is allowed, but inbound traffic is blocked.
- **Load balancer.** Allows the Azure Load Balancer to probe the health of your VMs and role instances. If you are not using a load balanced set, you can override this rule.

Note Load Balancer Traffic

The Load Balancer default rule uses the `AzureLoadBalancer` service tag. This applies only to Azure Load Balancer health probes, which originate at the Load Balancer. It does not apply to traffic received through the Load Balancer, which retain their original source IP addresses and ports.

[Table 4-7](#) shows the default inbound rules for each NSG.

TABLE 4-7 Default Inbound Rules

Name	Priority	Source	Source Port	Destination	Destination Port	Protocol	Access
AllowVNetInBound	65000	VirtualNetwork	Any	VirtualNetwork	Any	Any	Allow
AllowAzureLoadBalancerInBound	65001	AzureLoadBalancer	Any	Any	Any	Any	Allow
DenyAllInBound	65500	Any	Any	Any	Any	Any	Deny

[Table 4-8](#) shows the default outbound rules for each NSG.

TABLE 4-8 Default Outbound Rules

Name	Priority	Source	Source	Destination	Destination	Protocol	Access
------	----------	--------	--------	-------------	-------------	----------	--------

			Port					
AllowVNet OutBound	65000	VirtualNet work	Any	VirtualNet work	Any	Any	Any	Allow
AllowInternet OutBound	65001	Any	Any	Internet	Any	Any	Any	Allow
DenyAllOutB ound	65500	Any	Any	Any	Any	Any	Any	Deny

Application security groups

As you have seen, NSG rules are like traditional firewall rules and are defined using source and destination IP blocks. They enable you to segment your network traffic into application tiers, which are segmented into separate subnets.

This creates some management challenges:

- The IP blocks for each subnet must be carefully planned in advance. To allow for additional servers to be added in future, each subnet must be bigger than you really need, which results in inefficient use of the IP space.
- If you make a subnet too small and run out of space, it can be time-consuming to reconfigure the network to free up additional space, especially without application downtime.
- Each subnet requires a separate NSG, making it difficult to get an overall picture of the permitted and blocked traffic at an application level.

Application security groups (ASGs) address these challenges by offering an alternative approach to network segmentation. They allow you to achieve the same goal of segmenting your application into separate tiers, and they strictly control the permitted network flows between tiers. However, ASGs do not require that you associate each tier with a separate subnet, so all the challenges associated with planning and managing subnets fall away. With ASGs, you

explicitly define which application tier each VM belongs to, rather than implicitly defining which application tier each VM belongs to, based on the subnet in which the VM has been placed. All VMs can be placed in a single subnet, and a single NSG is used to define all permitted network flows between application tiers. Because a single subnet is used, the IP space can be managed much more flexibly, and because there is a single NSG with rules referring to named application tiers, the network rules are easier to understand and can all be managed in one place.

[Figure 4-37](#) shows a standard three-tier application architecture with web servers, application servers, and database servers. These servers have been grouped by associating each server with the appropriate application security group. All servers are placed in the same subnet without having to think about how the network space is subdivided. A single network security group contains rules defining the permitted traffic flows between application tiers.

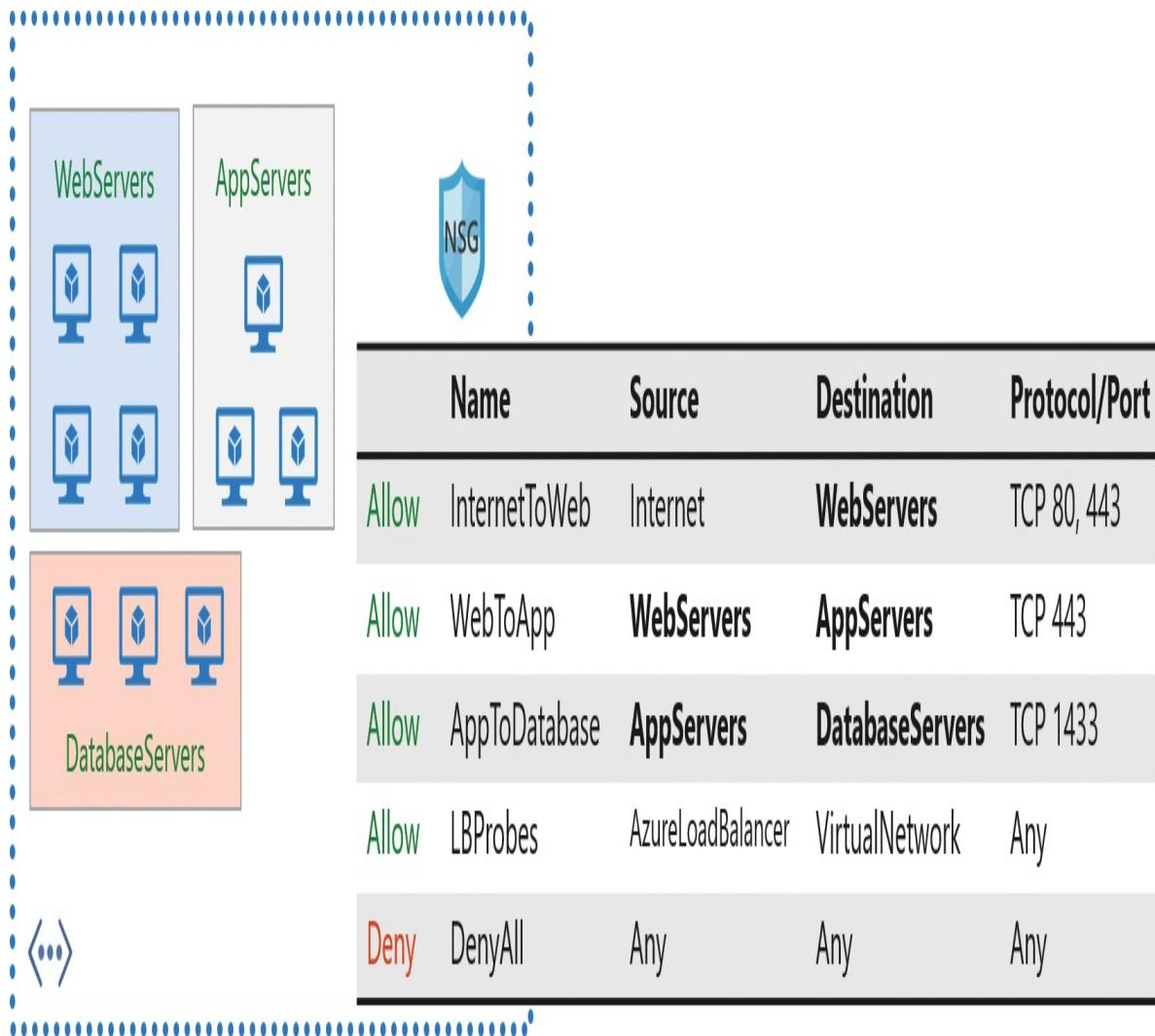


FIGURE 4-37 Using application security groups to simplify subnet and NSG management

Application security groups enable you to configure network security as a natural extension of an application's structure, which allows you to group virtual machines and define network security policies based on those groups. You can reuse your security policy at scale without the manual maintenance of explicit IP addresses. The platform handles the complexity of explicit IP addresses and multiple rule sets, which allows you to focus on your business logic.

Configuring application security groups is straightforward:

1. First, you create an application security group resource for each server group. This resource has no properties, other than its name, resource group,

and location.

2. Next, you associate the network interface from each VM with the appropriate application security group. This defines which group (or groups) each VM belongs to.
3. Finally, you define your network security group rules using application security group names instead of explicit IP ranges. This is similar to how rules are configured using named service tags.

Create an NSG using the Azure portal

To create an NSG using Azure portal, follow these steps:

1. First, click **Create A Resource** > **Networking** > **Network Security Group**.
2. Once the **Create Network Security Group** blade loads, you will need to provide a name, the subscription where your resources are located, the resource group for the NSG, and the location. (The location must be the same as the resources you want to apply the NSG.) In [Figure 4-38](#), the NSG will be created to allow HTTP traffic into the Apps subnet and be named `AppsNSG`.

Home > New >

Create network security group

Basics Tags Review + create

Project details

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

ExamRef-RG



[Create new](#)

Instance details

Name *

AppsNSG



Region *

(Canada) Canada Central



FIGURE 4-38 Creating a network security group using the Azure portal

3. After the NSG has been created, open the NSG Overview blade, as shown in [Figure 4-39](#). Here, you see that the NSG has been created, but there are no inbound or outbound security rules beyond the default rules.

AppsNSG   

Network security group

 **Overview** Resource group (change) : ExamRef-RG Custom security rules : 0 inbound, 0 outbound

 **Activity log** Location : Canada Central Associated with : 0 subnets, 0 network interfaces

 **Access control (IAM)** Subscription (change) : Visual Studio Ultimate with MSDN

 **Tags** Subscription ID : 00b72028-9dce-4729-9b2a-a10e90205447

 **Diagnose and solve problems** Tags (change) : Click here to add tags

Settings

 **Inbound security rules**

Priority	Name	Port	Protocol	Source	Destination	Action	...
65000	AllowInInBound	Any	Any	VirtualNetwork	VirtualNetwork	 Allow	
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	 Allow	
65500	DenyAllInBound	Any	Any	Any	Any	 Deny	

 **Outbound security rules**

Priority	Name	Port	Protocol	Source	Destination	Action	...
65000	AllowInetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	 Allow	
65001	AllowInternetOutBound	Any	Any	Any	Internet	 Allow	
65500	DenyAllOutBound	Any	Any	Any	Any	 Deny	

 **Properties**

 **Logs**

 **Export template**

Monitoring

 **Diagnostic settings**

 **NSG flow logs**

FIGURE 4-39 The NSG Overview blade, showing the inbound and outbound security rules

4. The next step is to create the inbound rule for HTTP and HTTPS traffic. In the **Settings** area, click **Inbound Security Rules**, and then click **+Add** to open the **Add Inbound Security Rule** panel. Notice how the panel has both **Basic** and **Advanced** modes, depending on the level of control required.
5. To allow HTTP/HTTPS traffic on Port 80 and 443, fill in the settings, as shown here and in [Figure 4-40](#):
 - **Source.** Any
 - **Source Port Ranges.** *
 - **Destination.** VirtualNetwork
 - **Destination Port Ranges.** 80,443
 - **Protocol.** TCP
 - **Action.** Allow
 - **Priority.** 100
 - **Name.** Allow_HTTP_HTTPS
 - **Description.** Allow HTTP and HTTPS inbound traffic on ports 80 and 443
6. Once all the settings have been filled in, click the **Add** button to create the NSG rule.

Note Applying NSGs to Virtual Networks

The destination IP ranges refer to the VNet, which allows the NSG to be applied to any subnet in any VNet and avoids coupling the NSG to a specific IP range. Traffic will only be permitted to those subnets where the NSG is applied.



Add inbound security rule

X

AppsNSG



Source * ⓘ

Any



Source port ranges * ⓘ

*

Destination * ⓘ

VirtualNetwork



Destination port ranges * ⓘ

80,443



Protocol *

Any

TCP

UDP

ICMP

Action *

Allow

Deny

Priority * ⓘ

100

Name *

Allow_HTTP_HTTPS



Description

Allow HTTP and HTTPS inbound traffic on ports 80 and 443



FIGURE 4-40 Adding an Inbound Rule to allow HTTP traffic

7. Once the inbound rule has been saved, it will appear in the Azure portal. Review your rule to ensure it has been created correctly.

Associate NSG to a subnet or network interface

NSGs are used to define the rules of how traffic is filtered for your IaaS deployments in Azure. We have seen how to create NSG resources and define the NSG rules. However, these NSGs, by themselves, are not effective until they are associated with a resource in Azure.

NSGs can be associated with network interfaces (NICs), which are associated to the VMs, or they can be associated with a subnet. Each NIC or subnet can only be associated with a single NSG. However, a single NSG can be associated with multiple NICs and/or subnets.

When associating an NSG with a NIC, it applies to all IP configurations in that NIC. All inbound and outbound traffic to and from the NIC must be allowed by the NSG. It is possible to have a multi-NIC VM, and you can associate the same or different NSG to each Network Interface.

Alternatively, NSGs can be associated with a subnet; in that case, they apply to all traffic to and from resources in that subnet. This approach is useful when applying the same rule across multiple VMs.

Note How NSGs are Applied

Microsoft does not recommend deploying NSGs to subnets and NICs within the same subnet. However, although Microsoft does not recommend it, this configuration is supported, and it's important to understand how NSGs are applied when deployed in this way.

For inbound traffic, first the NSG at the subnet is applied, followed by the NSG at the NIC. Traffic only flows if both NSGs allow the traffic to pass. For outbound traffic, the sequence is reversed. First, the NSG at the NIC is applied, followed by the NSG at the subnet. Again, traffic only flows if both NSGs allow the traffic to pass.

In all cases, rules within each NSG are applied in priority order, with the

first matching rule applicable first.

Associating an NSG with a subnet using the Azure portal

We have seen how to create an NSG and how to add an inbound rule for HTTP and HTTPS traffic. Yet, unless the NSG has been associated with subnets or NICs, that rule is not in effect.

The next task will be to associate a rule with the Apps subnet. You can use either the NSG blade or the virtual network subnet blade for this task. For this example, we'll use the former.

In the NSG blade of the Azure portal, click the **Subnets** link to show the list of subnets currently associated with the NSG, which should be empty at this stage. Click **+Associate** to open the **Associate Subnet** blade. Azure portal will ask for two configurations: the virtual network, and the subnet. Note that you can only select virtual networks in the same Azure region as the NSG. In [Figure 4-41](#), **ExamRef-VNet** has been selected from the **Virtual Network** drop-down menu, and **Apps** has been selected from the **Subnet** drop-down menu.

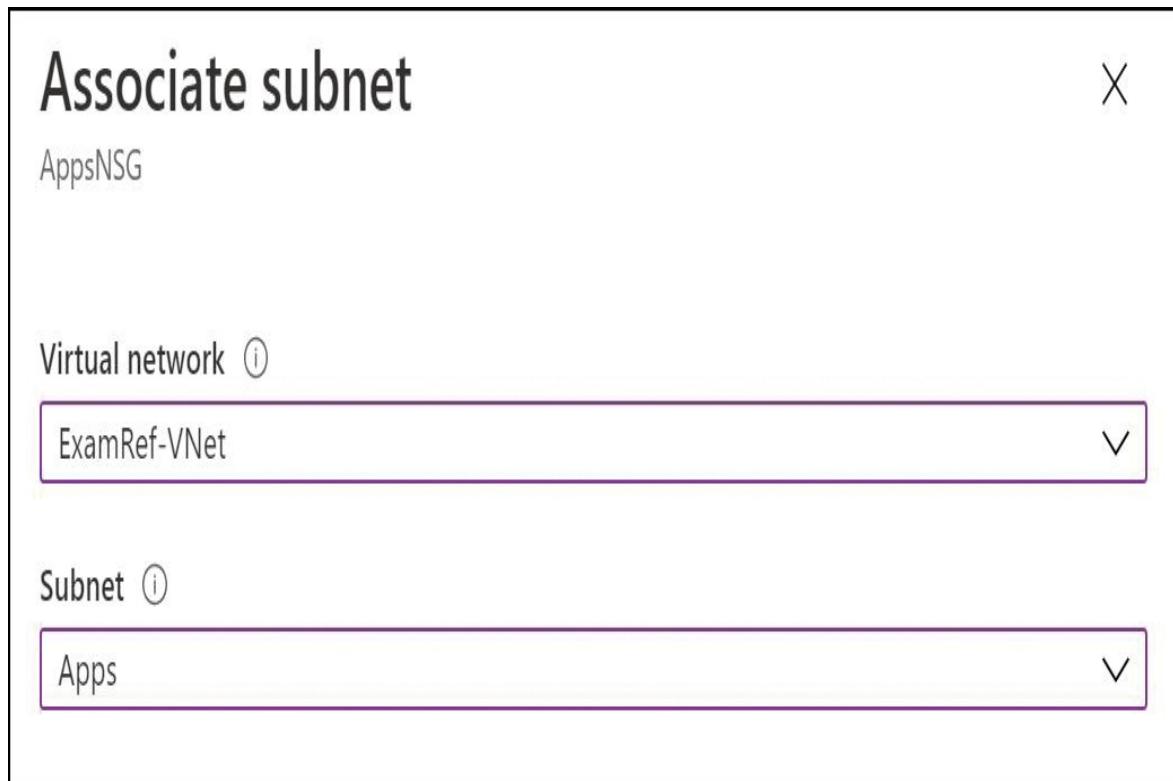


FIGURE 4-41 The ExamRef-VNet virtual network and Apps subnet have been selected

After being saved, the rules of the NSG are now being enforced for all network interfaces that are associated with this subnet. This will allow inbound TCP traffic on ports 80 and 443 for all VMs that are connected to this subnet. Of course, in order for it to respond, you need to have a webserver VM configured and listening on ports 80 or 443.

Evaluate effective security rules

When troubleshooting networking issues, it can be useful to get a deeper insight into exactly how NSGs are being applied. When NSG rules are defined using service tags and application security groups, instead of explicit IP addresses or prefixes, it sometimes isn't clear whether a particular flow matches a particular rule.

The **Effective Security Rules** view is designed to provide this insight. It allows you to drill into each NSG rule and see the exact list of source and destination IP prefixes that have been applied, regardless of how the NSG rule was defined.

To access the **Effective Security Rules** view, your virtual machine must be running because the data is taken directly from the configuration of the running VM.

View effective security rules using the Azure portal

Using the Azure portal, open the **Virtual Machine** blade, and then click **Networking**. This will show the networking settings, including the NSG rules and a convenient **Add Inbound Port Rule** button. At the top of this blade, click **Effective Security Rules**, as shown in Figure 4-42, to open the **Effective Security Rules** blade.

Home >

ExamRef | Networking

Virtual machine

Search (Ctrl+ /)

« ⚡ Attach network interface ⚡ Detach network interface

Overview

examref913

Activity log

IP configuration

ipconfig1 (Primary)

Tags

Network Interface: examref913 Effective security rules Topology

Diagnose and solve problems

Virtual network/subnet: VNet1/default NIC Public IP: - NIC Private IP: 10.0.0.5 Accelerated networking: Disabled

Settings

Inbound port rules Outbound port rules Application security groups Load balancing

Networking

Network security group ExamRef-nsgr attached to network interface: examref913

Add inbound port rule

Impacts 0 subnets, 1 network interfaces

Connect

Disks

Priority	Name	Port	Protocol	Source	Destination	Action	...
300	SSH	22	TCP	Any	Any	Allow	...
65000	AllowVnetInbound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowAzureLoadBalancerInbound	Any	Any	AzureLoadBalancer	Any	Allow	...
65500	DenyAllInbound	Any	Any	Any	Any	Deny	...

Size

Security

Advisor recommendations

Extensions

Continuous delivery

Figure 4-42 Azure Virtual Machine Networking blade

The **Effective Security Rules** blade (see [Figure 4-43](#)) looks very similar to the **Networking** blade shown in [Figure 4-42](#). It shows the name of the network interface and associated NSGs, along with a list of NSG rules.

The difference becomes clear when you click one of the NSG rules, which opens an additional pane that shows the exact source and destination IP address prefixes used by that rule. For example, in [Figure 4-44](#), you can see the exact list of 122 IP address prefixes used for outbound Internet traffic.

Having access to the exact list of address prefixes for each NSG rule allows you to investigate networking issues without fear of any ambiguity over how NSG rules are defined.

examref913 | Effective security rules

Network interface

Search (Ctrl+/) « [Download](#) [Refresh](#)

[Overview](#)
 [Activity log](#)
 [Access control \(IAM\)](#)
 [Tags](#)

Select a network interface below to see the effective security rules and network security groups associated with it.

Scope: Network interface (examref913)

Associated NSGs: [ExamRef-nsg \(Network interface\)](#)

[IP configurations](#)
 [DNS servers](#)
 [Network security group](#)

[Properties](#)
 [Locks](#)
 [Export template](#)

Support + troubleshooting

[Effective security rules](#) (selected)
 [Effective routes](#)
 [New support request](#)

ExamRef-nsg

Inbound rules

Name	Priority	Source	Source Ports	Destination	Destination Ports	Protocol	Access
SSH	300	0.0.0.0/0,0.0.0.0/0	0-65535	0.0.0.0/0,0.0.0.0/0	22-22	TCP	Allow
AllowVnetInBound	65000	Virtual network (3 prefixes)	0-65535	Virtual network (3 prefixes)	0-65535	All	Allow
AllowAzureLoadBalancerInBound	65001	Azure load balancer (2 prefixes)	0-65535	0.0.0.0/0,0.0.0.0/0	0-65535	All	Allow
DenyAllInBound	65500	0.0.0.0/0,0.0.0.0/0	0-65535	0.0.0.0/0,0.0.0.0/0	0-65535	All	Deny

Outbound rules

Name	Priority	Source	Source Ports	Destination	Destination Ports	Protocol	Access
AllowVnetOutBound	65000	Virtual network (3 prefixes)	0-65535	Virtual network (3 prefixes)	0-65535	All	Allow
AllowInternetOutBound	65001	0.0.0.0/0,0.0.0.0/0	0-65535	Internet (236 prefixes)	0-65535	All	Allow
DenyAllOutBound	65500	0.0.0.0/0,0.0.0.0/0	0-65535	0.0.0.0/0,0.0.0.0/0	0-65535	All	Deny

FIGURE 4-43 Azure virtual machine Effective Security Rules

examref913 | Effective security rules

Network interface

Search (Ctrl+I)

Download Refresh

Overview

Showing only top 50 security rules in each grid, click Download above to see all.

Activity log

Access control (IAM)

Tags

Scope

Network interface (examref913)

Settings

Associated NSGs: (0)

ExamRef-nsg (Network interface)

IP configurations

DNS servers

Network security group

Properties

Locks

Export template

Support + troubleshooting

Effective security rules

Effective routes

New support request

ExamRef-nsg

Inbound rules

Name	↑↓	Priority	↑↓	Source	Source Ports	↑↓	Destination	Destination Ports	↑↓	Protocol	↑↓	Access ↑↓
SSH		300		0.0.0.0/0.0.0.0/0	0-65535		0.0.0.0/0.0.0.0/0	22-22		TCP		Allow
AllowInnetInBound		65000		Virtual network (3 prefixes)	0-65535		Virtual network (3 prefixes)	0-65535		All		Allow
AllowAzureLoadBalancerInBound		65001		Azure load balancer (2 prefixes)	0-65535		0.0.0.0/0.0.0.0/0	0-65535		All		Allow
DenyAllInBound		65500		0.0.0.0/0.0.0.0/0	0-65535		0.0.0.0/0.0.0.0/0	0-65535		All		Deny

Outbound rules

Name	↑↓	Priority	↑↓	Source	Source Ports	↑↓	Destination	Destination Ports	↑↓	Protocol	↑↓	Access ↑↓
AllowInnetOutBound		65000		Virtual network (3 prefixes)	0-65535		Virtual network (3 prefixes)	0-65535		All		Allow
AllowInternetOutBound		65001		0.0.0.0/0.0.0.0/0	0-65535		Internet (236 prefixes)	0-65535		All		Allow
DenyAllOutBound		65500		0.0.0.0/0.0.0.0/0	0-65535		0.0.0.0/0.0.0.0/0	0-65535		All		Deny

Address prefixes

AllowInternetOutBound

Source Destination

1.0.0.0/8

2.0.0.0/7

4.0.0.0/6

8.0.0.0/7

11.0.0.0/8

12.0.0.0/8

13.0.0.0/10

13.64.0.0/11

13.104.0.0/13

13.112.0.0/12

13.128.0.0/9

14.0.0.0/7

16.0.0.0/6

20.0.0.0/9

20.129.0.0/16

20.131.0.0/16

20.132.0.0/14

20.136.0.0/13

20.148.0.0/14

20.152.0.0/13

20.160.0.0/12

20.184.0.0/13

20.192.0.0/10

FIGURE 4-44 Effective Security Rules showing Internet address prefixes

Implement Azure Firewall

Azure Firewall is a managed service that provides out-of-the-box network security for Azure resources. Because this is a managed service, its added advantages are that it offers high availability and scalability. In a typical enterprise environment, outbound network access restrictions are mandatory because they provide direct access to external web world. Azure Firewall provides an ability to limit the outbound IP addresses and ports that are allowed to communicate within an Azure subnet. It also provides additional benefits such as outbound SNAT support, Inbound DNAT support, and Azure Monitor logging.

Azure Firewall allows us to create and configure application and network rules. Application rules are created with the list of fully qualified names that are allowed to be accessed from a subnet. Network rules are a combination of source and destination IP addresses along with their ports and protocols. The network traffic solely depends on the firewall rules applied when the traffic is routed through the firewall.

Deploy and configure Azure Firewall using the Azure portal. Most of the enterprise deployments consider a hub-and-spoke model as a standard for Azure Firewall implementation where an Azure Firewall is hosted on its own VNet and the other resources are placed in peered VNets in the same region with one or more subnets. However, in small-scale deployments, Azure Firewall can be hosted in its dedicated subnet within a VNet.

Create required VNet and subnets

In the following example, it is assumed that you've already created the Exam-Ref-VNet, which has three subnets named AzureFirewallSubnet, WorkloadSubnet, and JumphostSubnet, as shown in [Figure 4-45](#). (See [Skill 4.1, “Implement and manage virtual networking,”](#) for detailed instructions for how to create a virtual network and the relevant subnets.)

The screenshot shows the Azure portal interface for managing a virtual network named "ExamRef-VNet". The main content area is titled "Subnets" and displays three entries:

Name	IPv4
AzureFirewallSubnet	10.1.0.0/24 (251 available)
WorkloadSubnet	10.1.1.0/24 (251 available)
JumphostSubnet	10.1.2.0/24 (251 available)

The sidebar on the left contains the following navigation links:

- Home > ExamRef-VNet | Subnets
- Virtual network
- Search (Ctrl+ /) (highlighted)
- Subnet
- Gateway subnet
- Refresh
- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings
- Address space
- Connected devices
- Subnets (highlighted)
- DDoS protection

FIGURE 4-45 The ExamRef-VNet with its subnets for Azure Firewall deployment

Note Azure Firewall Subnet Requirements

Azure Firewall must be hosted in a subnet named `AzureFirewallSubnet` with a minimum /26 address space in order for the Azure Firewall to provision more VMs to accommodate scaling.

Create required VMs

Now, you need to deploy two virtual machines—`ExamRef - Jump` and `ExamRef - Work`—in the same resource group as `ExamRef - VNet`. Remember to place the VMs in `JumpHostSubnet` and `WorkloadSubnet`, respectively. The networking blade for `ExamRef - Jump` is shown in [Figure 4-46](#).

Home > New >

Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.

[Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * ⓘ

ExamRef-VNet



[Create new](#)

Subnet * ⓘ

JumphostSubnet (10.1.2.0/24)



[Manage subnet configuration](#)

Public IP ⓘ

(new) Examref-Jump



[Create new](#)

NIC network security group ⓘ

None Basic Advanced

Public inbound ports * ⓘ

None Allow selected ports

Select inbound ports *

RDP (3389)



⚠️ This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Accelerated networking ⓘ

On Off

The selected VM size does not support accelerated networking.

Load balancing

You can place this virtual machine in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Place this virtual machine behind an existing load balancing solution?

Yes No

[Review + create](#)

< Previous

Next : Management >

FIGURE 4-46 The Networking blade while creating the Examref-Jump VM

The networking blade for `ExamRef-work` is shown in [Figure 4-47](#).

Home > New >

Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.

[Learn more](#)

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * ⓘ

ExamRef-VNet

[Create new](#)

Subnet * ⓘ

JumphostSubnet (10.1.2.0/24)

[Manage subnet configuration](#)

Public IP ⓘ

(new) Examref-Jump

[Create new](#)

NIC network security group ⓘ

None Basic Advanced

Public inbound ports * ⓘ

None Allow selected ports

Select inbound ports *

RDP (3389)

⚠ This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Accelerated networking ⓘ

On Off

The selected VM size does not support accelerated networking.

Load balancing

You can place this virtual machine in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Place this virtual machine behind an existing load balancing solution?

Yes No

[Review + create](#)

< Previous

Next : Management >

FIGURE 4-47 The Networking blade while creating the Examref-Work VM

Create an Azure Firewall

To deploy Azure Firewall, click **+Create A Resource > Networking > Firewall** to open the **Create A Firewall** blade. Fill in the blade by selecting the **Subscription**, **Resource Group**, and **Region** that are the same as what is used in the `ExamRef - VNet`. Also, specify the **Name** for the firewall, select the virtual network from the **Virtual Network** drop-down menu and create new public IP address, as shown in [Figure 4-48](#).

Home > New >

Create a firewall

Basics Tags Review + create

Azure Firewall is a managed cloud-based network security service that protects your Azure Virtual Network resources. It is a fully stateful firewall as a service with built-in high availability and unrestricted cloud scalability. You can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks. Azure Firewall uses a static public IP address for your virtual network resources allowing outside firewalls to identify traffic originating from your virtual network. The service is fully integrated with Azure Monitor for logging and analytics. [Learn more.](#)

Project details

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

ExamRef-RG

[Create new](#)

Instance details

Name *

ExamRef-Firewall

Region *

Canada Central

Availability zone ⓘ

None

Choose a virtual network

Create new Use existing

Virtual network

ExamRef-VNet (ExamRef-RG)

Public IP address *

(New) ExamRef-Firewall-ip

[Add new](#)

Forced tunneling ⓘ

Disabled

FIGURE 4-48 Basics blade while creating a firewall using Azure portal

Once created, the firewall overview blade will appear, as shown in [Figure 4-49](#).

ExamRef-Firewall



Firewall

Search (Ctrl+ /)

«

Delete Lock

Overview

This firewall can be managed by Azure Firewall Manager. →

Activity log

Access control (IAM)

Tags

Settings

DNS (preview)

Rules

Public IP configuration

Threat intelligence

Firewall Manager

Properties

Locks

Export template

Monitoring

Metrics

Diagnostic settings

Logs

Support + troubleshooting

New support request

Resource group ([change](#)) : ExamRef-RG Firewall subnet : AzureFirewallSubnet

Location : Canada Central Firewall public IP : ExamRef-Firewall-ip

Subscription ([change](#)) : Visual Studio Ultimate with MSDN Firewall private IP : 10.1.0.4

Subscription ID : 00b72028-9dce-4729-9b2a-a10e92054447 Management subnet :-

Virtual network : ExamRef-VNet Management public IP :-

Provisioning state : Succeeded Private IP Ranges : IANA RFC 1918

Tags ([change](#)) : Click here to add tags

▲

FIGURE 4-49 Overview of Azure Firewall

Create and associate a route table with a firewall

You will need to create a default route table to route the outbound requests through the firewall. To create route table, follow these steps:

1. Click **+Create A Resource > Networking, > Route Table** to open the **Create Route table** blade.
2. Select Choose the **Subscription**, **Resource Group**, and **Region** from the drop-down menus, and in the **Name** field, specify the route table name (see [Figure 4-50](#)).

Create Route table

Basics Tags Review + create

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Resource group * ⓘ

ExamRef-RG



[Create new](#)

Instance details

Region * ⓘ

Canada Central



Name * ⓘ

ExamRef-FW-RouteTable



Propagate gateway routes * ⓘ

Yes

No

FIGURE 4-50 Basics blade while creating a route table using Azure portal

3. Once created, associate `WorkloadSubnet` to this route table, as shown in [Figure 4-51](#).
4. Open the route table and then click **Subnets** to associate the `WorkloadSubnet`.

Home > Microsoft.RouteTable-20200803001453 | Overview >

ExamRef-FW-RouteTable | Subnets

Route table

Search (Ctrl+F) « + Associate

Overview Search subnets

Activity log Name ↑ Address range ↑ Virtual network ↑

Access control (IAM) No results.

Tags

Diagnose and solve problems

Settings

Configuration

Routes

Subnets

Properties

Locks

Export template

Support + troubleshooting

Effective routes

New support request

Associate subnet

X

ExamRef-FW-RouteTable

Virtual network (0)

ExamRef-VNet V

Subnet (0)

WorkloadSubnet V

FIGURE 4-51 Subnets page

5. Now you should add a default route to the firewall. From the **Routes** section in the left-side navigation menu, click **Add**. Set the **Route Name**, **Address Prefix**, **Next Hop Type**, and **Next Hop Address** options, as shown in [Figure 4-52](#).
6. To set a route as the default route, the **Address Prefix** must be set to 0.0.0.0/0. For the **Next Hop Address**, use the private IP address of the firewall created in the previous step.

Add route

ExamRef-FW-RouteTable

Route name *

Default-FW-Route



Address prefix * ⓘ

0.0.0.0/0



Next hop type ⓘ

Virtual appliance



Next hop address * ⓘ

10.1.0.4



Ensure you have IP forwarding enabled on your virtual appliance. You can enable this by navigating to the respective network interface's IP address settings.

FIGURE 4-52 Add default route to the route table

Add an Application rule collection

To filter outbound web traffic, you need to create an application rule. The following steps will allow you to create an application rule to allow outbound access to www.microsoftpressstore.com.

1. From the Azure portal, go to the **ExamRef-Firewall > Rules** section, and then select **Application Rule Collection > +Add Application Rule Collection**, as shown in [Figure 4-53](#).

The screenshot shows the Azure Firewall Rules blade for the resource 'ExamRef-Firewall'. The left sidebar includes links for Home, Firewall, Overview, Activity log, Access control (IAM), Tags, Settings, DNS (preview), Rules (selected), Public IP configuration, and Threat intelligence. The main area has a search bar and a refresh button. A note states: 'This firewall can be managed by Azure Firewall Manager.' Below are tabs for NAT rule collection, Network rule collection, and Application rule collection (underlined). A button '+ Add application rule collection' is visible. A table header row shows columns for Priority and Name. A message at the bottom notes: 'Azure infrastructure application rule collection is enabled by default. [Learn more](#)'.

Figure 4-53 Adding an application rule collection

2. Specify the application rule **Name**, **Priority**, and **Action**, as shown in Figure 4-54.

3. Under **Target FQDNs**, add an entry to allow the Microsoft Press Store website from the `WorkloadSubnet`.
4. Specify the **Source IP** address as `10.1.1.0/24`, the **Protocol** as `https`, and the **Target FQDN** as www.microsoftpressstore.com.

Add application rule collection

X

Name * AppRule1 ✓

Priority * 100 ✓

Action * Allow ✓

Rules

FQDN tags

name	Source type	Source	FQDN tags
	IP address ✓	* , 192.168.10.1, 192.168.10.0/24, 192.1...	0 selected ✓

 FQDN tags may require additional configuration. [Learn more.](#)

Target FQDNs

name	Source type	Source	Protocol:Port	Target FQDNs
Allow-MicrosoftPressStore ✓	IP address ✓	10.1.0.0/24 ✓	https ✓	www.microsoftpressstore.co... ✓
	IP address ✓	* , 192.168.10.1, 192.168.10.0/...	http, http:8080, https, mssql:1...	www.microsoft.com, *.micros...

 mssql: SQL should be enabled in proxy mode. This may require additional configuration. [Learn more.](#)

FIGURE 4-54 Add Application Rule Collection blade

Note Infrastructure FQDNs

By default, infrastructure FQDNs are allowed by Azure Firewall with a built-in rule collection. You can override infrastructure FQDNs by creating a deny all applications rule collection. The following services are included in the built-in rule collection:

- Compute access to storage Platform Image Repository (PIR)
- Managed disks status storage access
- Azure Diagnostics and Logging (MDS)

Add network rule collection

To resolve the FQDN, the network rule must be created to allow the DNS requests from the `WorkloadSubnet` to the public DNS servers (in our case, OpenDNS). To add a network rule, follow these steps:

1. From Azure portal, go to the **ExamRef-Firewall > Rules** section, choose **Network Rule Collection** and click **+ Add Network Rule Collection**, as shown in [Figure 4-55](#).

The screenshot shows the 'ExamRef-Firewall | Rules' interface. At the top left is a 'Home >' link. Below it is a logo with a red and blue shield-like icon and the text 'ExamRef-Firewall | Rules'. Underneath the logo is the word 'Firewall'. To the right of the logo is a search bar with the placeholder 'Search (Ctrl+ /)' and a 'Refresh' button. On the left side, there is a sidebar with icons and labels: 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Settings', 'DNS (preview)', 'Rules' (which is highlighted with a grey background), and 'Public IP configuration'. In the center, under the 'Network rule collection' tab, there is a button labeled '+Add network rule collection'. Below this, there are two columns: 'Priority' and 'Name', with the text 'No results' underneath.

FIGURE 4-55 Add network rule collection under Rules for ExamRef-Firewall

2. Specify the network rule **Name**, **Priority**, and **Action**, as shown in [Figure 4-56](#).
3. Choose **Rules -> IP Addresses**, and add an entry that allows DNS servers

from the WorkloadSubnet.

4. Set the **Protocol** to UDP; **Source IP Address** to 10.1.1.0/24; **Destination IP Address** to 208.67.222.222, 208.67.220.220 (OpenDNS servers); and the **Destination Port** to 53.

Add network rule collection

X

Name *

NetworkRule1

✓

Priority *

100

✓

Action *

Allow

✓

Rules

IP Addresses

name	Protocol	Source type	Source	Destination type	Destination Addr...	Destination Ports	⋮
Allow-DNS	UDP	IP address	10.1.1.0/24	IP address	208.67.222.222, ...	53	✓
	0 selected	IP address	* , 192.168.10.1, 192...	IP address	* , 192.168.10.1, 192...	8080, 8080-8090, *	

Service Tags

name	Protocol	Source type	Source	Service Tags	Destination Ports	⋮
	0 selected	IP address	* , 192.168.10.1, 192.168...	0 selected	8080, 8080-8090, *	

FQDNs (preview)

name	Protocol	Source type	Source	Destination FQDNs	Destination Ports	⋮
	0 selected	IP address	* , 192.168.10.1, 192.168...	time.windows.com	8080, 8080-8090, *	

FIGURE 4-56 Adding a network rule collection

You need to make sure that your `ExamRef - Work` VM uses OpenDNS server addresses to resolve FQDNs.

1. Go to the network interface for the `examref-work131` VM and add these custom DNS servers: `208.67.222.222` and `208.67.220.220` (see [Figure 4-57](#)).
2. Restart the VM to apply the changes.

examref-work131 | DNS servers

Network interface

 Search (Ctrl+ /)

«

 Save  Discard

 Overview

DNS servers

- Inherit from virtual network
 Custom

 Activity log

208.67.222.222 

 Access control (IAM)

208.67.220.220 

 Tags

Add DNS server 

Settings

 IP configurations

Applied DNS servers

208.67.222.222

208.67.220.220

 DNS servers

 Network security group

 Properties

 Locks

 Export template

Support + troubleshooting

 Effective security rules

 Effective routes

 New support request

FIGURE 4-57 Update the custom DNS servers for the ExamRef-Work VM

Test the Azure Firewall

To test the Azure Firewall rules, follow these steps:

1. Connect to the `ExamRef-Jump` using the public IP address `52.228.7.130`.
2. Then RDP from the `ExamRef-Jump` VM to connect to the **ExamRef-Work** machine using the private IP address `10.1.1.4`.

To test the application rule created earlier, open a web browser on the `ExamRef-Work` VM and type the website URL
<https://www.microsoftpressstore.com/>, as shown in Figure 4-58.

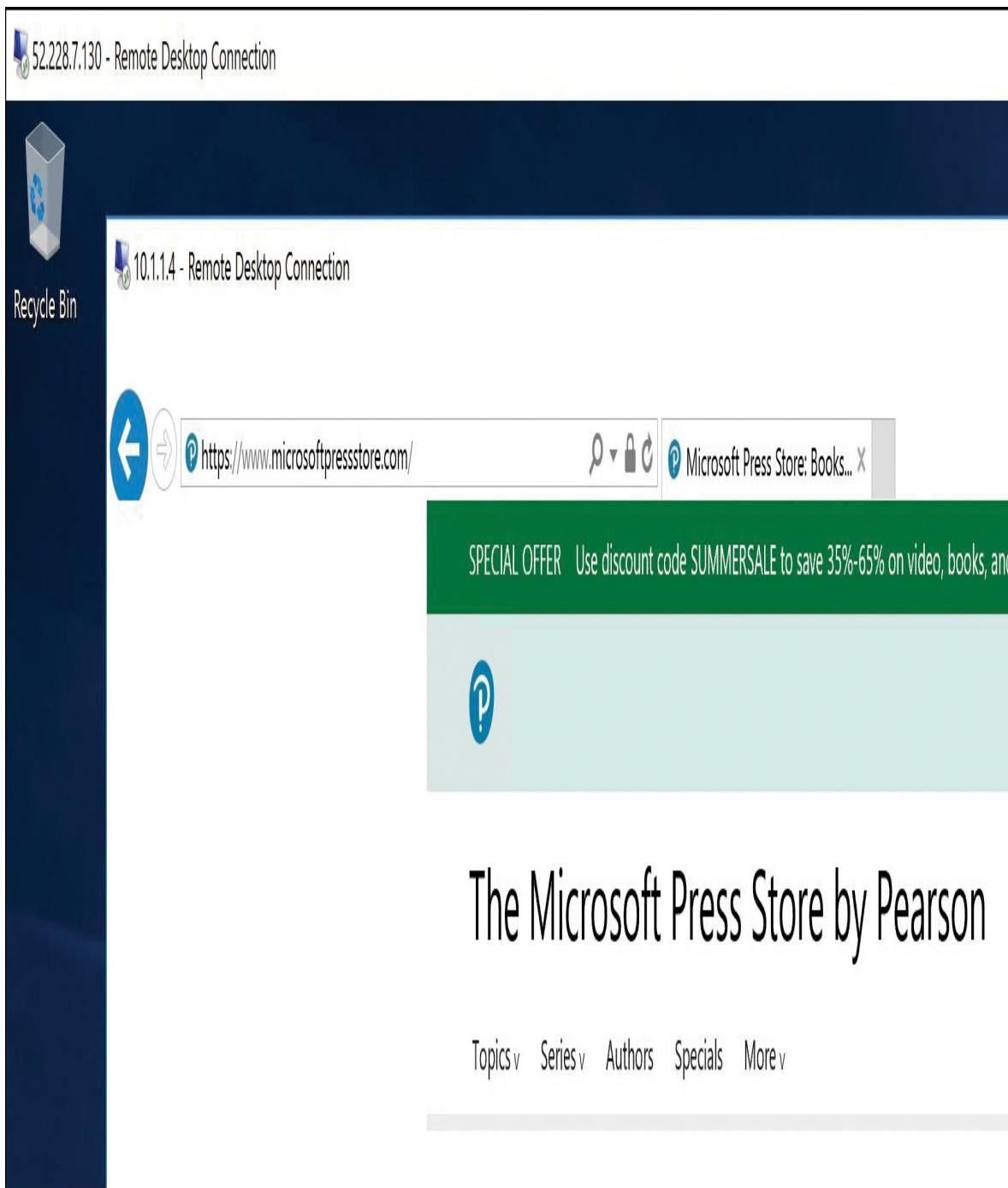


FIGURE 4-58 Test firewall application rules for the Microsoft Press Store website

This should work just fine, but if you try to access any other websites like www.microsoft.com or www.google.com, those will be blocked by the firewall. If you want to access these websites, you should create application rule for each of

them in Azure Firewall.

Create a DNAT rule

If you don't want to use **ExamRef-Jump** VM to connect to the **ExamRef-Work** VM, then you should consider creating a DNAT rule in firewall. This rule will allow you to connect to the **ExamRef-Work** VM through the firewall (using firewall's public IP address). Follow these steps to create a DNAT rule:

1. Go to the **ExamRef-Firewall > Rules** section, choose **NAT Rule Collection**, and click **+ Add NAT Rule Collection**, as shown in [Figure 4-59](#).
2. Specify the NAT rule **Name** and **Priority**, as shown in [Figure 4-60](#).
3. Under **Rules**, add an entry that allows RDP using private IP addresses through Azure Firewall.
4. Specify the following settings:
 - **Protocol.** TCP
 - **Source.** *
 - **Destination IP Address.** 20.39.143.187 (Firewall Public IP Address)
 - **Destination Port.** 3389
 - **Translated Address.** 10.1.1.4 (ExamRef-Work VM Private IP Address)
 - **Translated Port.** 3389

The screenshot shows the 'Rules' blade for the 'ExamRef-Firewall' resource in the Azure portal. The top navigation bar includes 'Home >' and the resource name 'ExamRef-Firewall | Rules'. Below the navigation is a search bar ('Search (Ctrl+ '/')') and a refresh button. On the left, there's a sidebar with icons for Overview, Activity log, Access control (IAM), Tags, Settings, DNS (preview), Rules (selected), Public IP configuration, and Threat intelligence. The main content area is titled 'NAT rule collection' and contains tabs for 'Network rule collection' and 'Application rule collection'. A blue button labeled '+ Add NAT rule collection' is highlighted with a red box. Below this, a table lists results with columns for 'Priority' and 'Name', showing 'No results'. A note on the right states: 'When a DNAT rule is matched, an implicit corresponding network rule to allow the translated traffic is added. [Learn more](#)'.

FIGURE 4-59 Add NAT rule collection under Rules for ExamRef-Firewall

Add NAT rule collection

Name * ✓

Priority * ✓

Action ✓

Rules

name	Protocol	Source type	Source	Destination Addr...	Destination Ports	Translated address	Translated ports
RDP-NAT ✓	TCP ✓	IP address ✓	* ✓	20.39.143.187 ✓	3389 ✓	10.1.1.4 ✓	3389
	0 selected ✓	IP address ✓	* ,192.168.10.1,192...	192.168.10.0	8080	192.168.10.0	8080

FIGURE 4-60 Adding a NAT rule collection

Once created, you should be able to RDP to the **ExamRef-Work** VM using the firewall's public IP address.

More Info Deploy Azure Firewall using PowerShell or CLI

Azure Firewall can be deployed using PowerShell or CLI using the following documentation.

- PowerShell: <https://docs.microsoft.com/en-us/azure/firewall/deploy-ps>
- CLI: <https://docs.microsoft.com/en-us/azure/firewall/deploy-cli>

Implement Azure Bastion Service

You generally connect to remote machines with either RDP or SSH. To do so, you either need to assign a public IP address (with the RDP/SSH port exposed) to the VM to which you are trying to connect, or you need to provision an additional jump server, assign a public IP address to that jump server, and then connect to the other virtual machines using private IP addresses internally.

You can also try implementing Network Security Groups (NSGs) to restrict the source IP addresses and ports allowed for your network traffic. Still, you are exposing RDP/SSH ports to the source servers over the Internet, which could be a potential security threat.

To overcome this issue, Microsoft has created a managed PaaS service called Azure Bastion to provide secure connections to Azure Virtual Machines using the SSL channel through a browser directly without using any external client. This service helps you to limit threats like port scanning and other malware.

Note Azure Bastion Regions

The Azure Bastion service is only available in selected regions across the globe. You can find the supported regions at <https://docs.microsoft.com/en-us/azure/bastion/bastion-overview#faq>.

The Azure Bastion service is provisioned within a VNet within a separate subnet called `AzureBastionSubnet`. If you have multiple VNets in your environment, you will need to deploy Azure Bastion for each VNet separately.

Deploy and configure Azure Bastion Service using the Azure portal

In the following example, it is assumed that you have already created the `Exam-Ref-VNet` with a subnet named `AzureBastionSubnet` and with a prefix of at least /27. Refer to “Create and configure VNet peering”, for detailed instructions on

how to create a virtual network and subnet.

To create a Bastion service using Azure portal, follow these steps:

1. Click **Create A Resource**, and then search for and select **Bastion**.
2. Once the **Create A Bastion** blade loads, you will need to provide a **Name**, the **Subscription** where your resources are located, the **Resource Group** for the Bastion, and the region (select the supported region).
3. You also need to select the **Virtual Network** and **Subnet** and create a **Public IP Address**, as shown in [Figure 4-61](#).

Create a Bastion

Basics Tags Review + create

Bastion allows web based RDP access to your vnet VM. [Learn more.](#)

Project details

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

ExamRef-RG

[Create new](#)

Instance details

Name *

ExamRef-Bastion

Region *

Canada Central

Configure virtual networks

Virtual network * ⓘ

ExamRef-VNet

[Create new](#)

Subnet *

AzureBastionSubnet (10.1.3.0/24)

[Manage subnet configuration](#)

Public IP address

Public IP address * ⓘ

Create new Use existing

Public IP address name *

ExamRef-Bastion-ip

Public IP address SKU

Standard

Assignment

Dynamic Static

FIGURE 4-61 Creating a Bastion

4. Once created, the **ExamRef-Bastion** overview blade will appear, as shown in [Figure 4-62](#).

ExamRef-Bastion

Bastion

Search (Ctrl+ /)

Delete Lock

Overview

Resource group (change) : ExamRef-RG

Virtual network/subnet : ExamRef-VNet/AzureBastionSubnet

Activity log

Location : Canada Central

Public DNS name : bst-36b2107a-febf-4915-adba-726416ea124.bastion.azure.com

Access control (IAM)

Subscription (change) : Visual Studio Ultimate with MSDN

Public IP address : ExamRef-Bastion-ip

Tags

Subscription ID : 00672028-9dce-4729-9b2a-a10e92054447

Provisioning state : Succeeded

Tags (change) : Click here to add tags

Settings



Sessions

Properties

Locks

Export template

Monitoring

Logs

Diagnostics settings

Support + troubleshooting

New support request

FIGURE 4-62 Overview blade of ExamRef-Bastion

5. To test this Bastion, browse to the **Overview** blade of your **ExamRef-Work** VM, click **Connect**, and select the **Bastion** tab, as shown in [Figure 4-63](#).

ExamRef-Work | Connect

Virtual machine

 Search (Ctrl+ /)

«

 To improve security, enable just-in-time access on this VM. →

 Overview

 Activity log

 Access control (IAM)

 Tags

 Diagnose and solve problems

Settings

 Networking

 Connect

 Disks

 Size

 Security

 Advisor recommendations

 Extensions

RDP SSH **BASTION**

Connect with Bastion

To connect to your virtual machine over the web, enter login credentials and click connect (opens a new browser window).

Open in new window

Username * 



Password * 



Connect

FIGURE 4-63 Connecting to the ExamRef-Work VM through Azure Bastion

6. Once you click **Connect**, you will be redirected to the interactive browser session to the **ExamRef-Work** VM through Bastion, as shown in [Figure 4-64](#).

Server Manager

Server Manager • Local Server

Manage Tools View Help

Dashboard Local Server All Servers File and Storage Services >

PROPERTIES

For ExamRef-Work

Computer name: ExamRef-Work Last installed updates: Today at 6:08 AM
Workgroup: WORKGROUP Windows Update: Install updates automatically using Windows Update
Last checked for updates: Today at 2:59 AM

Windows Firewall: Public On Windows Defender: Real-Time Protection: On
Feedback & Diagnostics: Settings
IE Enhanced Security Configuration: On
Time zone: (UTC) Coordinated Universal Time
Product ID: 00376-40000-00000-AA947 (activated)

Clipboard

Text copied/cut within Bastion will appear here. Changes to the text below will affect the remote clipboard.

Windows Server 2016 Datacenter
Processor: Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz
Virtual Machine
Installed memory (RAM): 3.5 GB
Total disk space: 133.51 GB

Log Date and Time

System 8/3/2020 8:21:00 AM
System 8/3/2020 8:20:58 AM
System 8/3/2020 8:20:56 AM
System 8/3/2020 8:20:54 AM
System 8/3/2020 8:20:54 AM
System 8/3/2020 8:20:54 AM
ExamRef-Work 10016 Error Microsoft-Windows-DistributedCOM System 8/3/2020 8:20:49 AM

Fullscreen

SERVICES

All services | 191 total

8:22 AM 8/3/2020

This screenshot shows the Windows Server 2016 Server Manager interface. The left navigation bar includes links for Dashboard, Local Server (which is selected), All Servers, and File and Storage Services. The main content area displays the 'PROPERTIES' for the local server 'ExamRef-Work', showing details like computer name, workgroup, and security settings. A 'Clipboard' section allows copying and pasting text between the host and guest OS. Below that is a 'System Log' table showing recent system events. The bottom section is titled 'SERVICES' and lists all services running on the server. The taskbar at the bottom shows the date and time as 8:22 AM on 8/3/2020.

FIGURE 4-64 Managing ExamRef-Work VM through Bastion

More Info Deploy Azure Bastion using PowerShell or CLI

Azure Bastion can be deployed using PowerShell or CLI using the following documentation.

- PowerShell: <https://docs.microsoft.com/en-us/azure/bastion/bastion-create-host-powershell>
- CLI: <https://docs.microsoft.com/en-us/azure/bastion/create-host-cli>

Skill 4.3: Configure load balancing

Load balancing is one of the crucial requirements of a network design. Azure offers various options to design load balancing solutions. In this section, you will learn how to configure Azure Application Gateway and different Load Balancers in Azure.

Azure Application Gateway is a type of Load Balancer that can manage traffic for web applications. The web traffic routing occurs at application layer (OSI layer 7). The Azure Application Gateway offers additional features, such as SSL/TLS termination, autoscaling, URL-based routing, redirection, and the like.

More Info Azure Application Gateway Documentation

For a complete list of features, see the official documentation:
<https://docs.microsoft.com/en-us/azure/application-gateway/features>.

Azure Load Balancer is a fully managed load-balancing service, which is used to distribute inbound traffic across a pool of back-end servers running in an Azure virtual network. It can receive traffic on either Internet-facing or Intranet-facing endpoints and supports both UDP and TCP traffic.

Azure Load Balancer operates at the transport layer (OSI layer 4) to route inbound and outbound connections at the packet level. It does not terminate TCP connections, and thus, it does not have visibility into application-level

constructs. For example, it cannot support SSL offloading, URL path-based routing, or cookie-based session affinity. (For these, see “Application Gateway” in Skill 3.1.)

Azure Load Balancer provides low latency and high throughput, scaling to millions of network flows. It also supports automatic failover between back-end servers based on health probes and enables high availability applications.

This section covers how to:

- [Configure Application Gateway](#)
- [Azure Load Balancer](#)
- [Configure an internal or public load balancer](#)
- [Troubleshoot load balancing](#)

Configure Azure Application Gateway

An application gateway routes application web traffic to defined resources in a back-end pool. To create and configure an application gateway, follow the steps in the next section. Remember, the load-balancing options in Azure can be configured in multiple ways to address various requirements.

Create an application gateway using the Azure portal

To create an application gateway using Azure portal, follow these steps:

1. Click **Create A Resource**, and then search for and select **Application Gateway**.
2. On the **Create Application Gateway** page, you will have four different blades to input the parameters for the application gateway: **Basics**, **Frontends**, **Backends**, and **Configuration**.
3. On the **Basics** blade, you need to select the **Subscription**, **Resource Group**, **Region**, **Tier**, and **Name** for the application gateway. Leave the default values for autoscaling.
4. You also need to select the virtual network and dedicated subnet for your application gateway (refer [Figure 4-65](#)). Note that you need a separate

subnet for your back-end servers (in our case, `ExamRef-WorkloadSubnet`).

Create application gateway

1 Basics

2 Frontends

3 Backends

4 Configuration

5 Tags

6 Review + create

An application gateway is a web traffic load balancer that enables you to manage traffic to your web application. [Learn more about application gateway](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Resource group * ⓘ

(New) ExamRefAG-RG

[Create new](#)

Instance details

Application gateway name *

ExamRef-AG

Region *

Canada Central

Tier ⓘ

Standard V2

Enable autoscaling

Yes No

Minimum scale units * ⓘ

0

Maximum scale units

10

Availability zone ⓘ

None

HTTP2 ⓘ

Disabled Enabled

Configure virtual network

Virtual network * ⓘ

ExamRef-VNet

[Create new](#)

Subnet * ⓘ

ApplicationGatewaySubnet (10.1.4.0/24)

[Manage subnet configuration](#)

[Previous](#)

[Next : Frontends >](#)

FIGURE 4-65 Creating an application gateway

5. On the **Frontends** blade, you need to configure the **Frontend IP Address Type** for the application gateway. Choose either **Public** or **Private** based on your requirements. In this example, we will create a new Public IP named **ExamRef-AG-ip**, as shown in [Figure 4-66](#).

Create application gateway

✓ Basics

2 Frontends

3 Backends

4 Configuration

5 Tags

6 Review + create

Traffic enters the application gateway via its frontend IP address(es). An application gateway can use a public IP address, private IP address, or one of each type.

Frontend IP address type ⓘ

Public Private Both

Public IP address *

(New) ExamRef-AG-ip



Add new

Add a public IP

Name *

ExamRef-AG-ip

SKU

Basic

Standard

Assignment

Dynamic

Static

OK

Cancel

FIGURE 4-66 Frontends blade while creating an application gateway

6. On the **Backends** blade, you can create a back-end pool with resources to which the application gateway can send traffic. Click **Add A Backend Pool** and add both the VMs created earlier in the Azure Firewall configuration as back-end targets (see [Figure 4-67](#)).
7. Instead of VMs, you can also specify the IP address, VMSS, or app services.

Home > New > Application Gateway >

Create application gateway

✓ Basics ✓ Frontends 3 Backends 4 Configuration 5 Tags 6 Review + create

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, app services, IP addresses, or fully qualified domain names (FQDN).

Add a backend pool

Backend pool	Targets
No results	

Add a backend pool.

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, app services, IP addresses, or fully qualified domain names (FQDN).

Name * ExamRefAG-Backend ✓

Add backend pool without targets Yes No

Backend targets

2 items

Target type	Target
Virtual machine	examref-jump399
Virtual machine	examref-work131

IP address or FQDN

Virtual machine

VMS

App Services

FIGURE 4-67 Backends blade while creating application gateway

8. The **Configuration** blade is shown in [Figure 4-68](#). You can still add or update frontends and backends from here.



FIGURE 4-68 Configuration blade while creating application gateway

9. Click the **+ Add A Routing Rule** button to connect your frontend and backend pool.

10. In addition to choosing a routing **Rule Name**, on the **Listener** tab, specify the **Listener Name** and select the **Frontend IP**, **Protocol**, **Port**, and **Listener Type**, as shown in [Figure 4-69](#).

Add a routing rule

X

Configure a routing rule to send traffic from a given frontend IP address to one or more backend targets. A routing rule must contain a listener and at least one backend target.

Rule name *

ExamRefAG-RoutingRule



*Listener *Backend targets

A listener "listens" on a specified port and IP address for traffic that uses a specified protocol. If the listener criteria are met, the application gateway will apply this routing rule.

Listener name * ⓘ

ExamRefAG-Listener



Frontend IP * ⓘ

Public



Protocol ⓘ

HTTP HTTPS

Port * ⓘ

80



Additional settings

Listener type ⓘ

Basic Multi site

Error page url

Yes No

FIGURE 4-69 Add A Routing Rule—Listener tab

11. On the **Backend Targets** tab, select the **Target Type** and **Backend Target** created in the previous step. Also, choose new **HTTP Settings**, as shown in [Figure 4-70](#).
12. Once a routing rule is added, you can create the application gateway; it takes several minutes for the deployment to complete (see [Figure 4-71](#)).
13. Now, you can use application gateway public IP to serve the web request for backend servers. You should see it as load-balanced between all the backend servers, based on the settings.

Add a routing rule

X

Configure a routing rule to send traffic from a given frontend IP address to one or more backend targets. A routing rule must contain a listener and at least one backend target.

Rule name *

ExamRefAG-RoutingRule



* Listener * Backend targets

Choose a backend pool to which this routing rule will send traffic. You will also need to specify a set of HTTP settings that define the behavior of the routing rule.

Target type

Backend pool Redirection

Backend target * ⓘ

ExamRefAG-Backend



[Add new](#)

HTTP settings * ⓘ

ExamRef-HTTPSettings



[Add new](#)

Path-based routing

You can route traffic from this rule's listener to different backend targets based on the URL path of the request. You can also apply a different set of HTTP settings based on the URL path.

Path based rules

Path	Target name	HTTP setting name	Backend pool
------	-------------	-------------------	--------------

No additional targets to display

[Add multiple targets to create a path-based rule](#)

FIGURE 4-70 Backend Targets tab

ExamRef-AG

Application gateway

Search (Ctrl+ /) Delete Refresh

Overview

Resource group (change) : ExamRef-AG-RG Virtual network/subnet : ExamRef-VNet/ApplicationGatewaySubnet

Location : Canada Central Frontend public IP address : 40.82.169.255 (ExamRef-AG-ip)

Subscription (change) : Visual Studio Ultimate with MSDN Frontend private IP addr... : -

Subscription ID : 00b72028-9dce-4729-9b2a-a10e92054447 Tier : Standard V2

Tags (change) : Click here to add tags

Show data for last 1 hour 6 hours 12 hours 1 day 7 days 30 days

Sum Total Requests

Sum Failed Requests

Sum Response Status by HttpStatus

Sum Throughput

Sum CurrentConnections

Avg Healthy Host Count By BackendPool HttpSettings

Alerts Metrics Diagnostic settings Logs Insights (preview) Backend health Connection troubleshoot

FIGURE 4-71 The application gateway overview page

Azure Load Balancer

The deployment of Azure Load Balancer involves the coordinated configuration of several groups of settings. These settings work together to define the overall Load Balancer behavior.

Basic and Standard Load Balancer tiers

Azure Load Balancer is available in two pricing tiers (SKUs): Basic and Standard. These tiers offer different levels of scale, features, and pricing. [Table 4-9](#) provides a comparison of the main feature differences between the Basic and Standard tiers.

TABLE 4-9 Standard and Basic Load Balancer Tiers

FEATURE	Standard	Basic
Availability Zones	Supports zone-specific or zone-redundant deployments, including cross-zone load-balancing	Not supported
Backend Pools	Up to 1,000 servers, any mix of VMs, availability sets, and VM Scale Sets—in the same VNet	Up to 300 servers Must be VMs in the same availability set or a single VM Scale Set
Health Probes	TCP, HTTP, HTTPS	TCP, HTTP
Diagnostics	Rich metrics via Azure Monitor, including byte statistics and packet counters, health probe status, connection attempts, outbound connection health, and more	Azure Monitor for public Load Balancer only

		Alerts and backend pool health count
Security	Inbound flows closed by default Access less-permitted inbound flows using Network Security Groups	Open by default Can optionally restrict flows using Network Security Groups
Outbound Connectivity	Supports multiple outbound IP addresses that are configurable via outbound rules	Single outbound IP Not configurable
Other Features	Supports HA Ports, TCP Reset on idle timeout, and faster management operations	N/A
Pricing	Based on the number of rules and data processed	Free
SLA	99.99 percent availability for a data path with two healthy VMs	None

Frontend IP configuration

Azure Load Balancer supports two modes: internal Load Balancer or public Load Balancer. In each case, the frontend IP configuration defines the endpoint upon which the Load Balancer receives incoming traffic.

- **Internal Load Balancer.** Used to load-balance traffic for Intranet-facing applications, or between application tiers. The frontend IP configuration references a subnet, and an IP address from that subnet is allocated using either dynamic or static assignment to the Load Balancer.
- **Public Load Balancer.** Used to load-balance traffic for Internet-facing applications. The frontend IP configuration references a separate public IP address resource, which is used to receive inbound traffic.

When used with IaaS VMs, each Load Balancer can support multiple

frontend IP configurations. This allows it to receive traffic on multiple IP addresses, to load-balance traffic for multiple applications. All frontend configurations, however, must be of the same type: internal or public.

A public Load Balancer must be associated with a public IP address resource. If the load-balancer uses the standard pricing tier, then the public IP address must also use the standard pricing tier. Standard tier Load Balancers support both zone specific and zone redundant deployment options. The choice of deployment option is taken from the associated public IP address, rather than being explicitly in the Load Balancer properties.

Backend configuration

The backend pool defines the backend servers over which the Load Balancer will distribute incoming traffic.

When using a basic-tier Load Balancer, this backend pool must comprise either a single virtual machine, virtual machines in the same availability set, or a VM scale set. (Traffic will be distributed to all virtual machines in the VM scale set.) You cannot distribute traffic to multiple virtual machines unless they are members of the same availability set or VM scale set.

With a standard-tier load-balancer, these restrictions are lifted. Backend pools can comprise a combination of virtual machines across availability sets and VM scale sets.

Health Probes

Azure Load Balancer supports continual health probing of backend pool instances to determine which instances are healthy and able to receive traffic. The Load Balancer will stop sending traffic flows to any backend pool instance that is determined to be unhealthy. Unhealthy instances continue to receive health probes, so the Load Balancer can resume sending traffic to that instance once it returns to a healthy state.

Azure Load Balancer supports three types of health probes:

- **TCP.** Probes attempt to initiate a connection by completing a three-way TCP handshake (SYN, SYN-ACK, ACK). If successful, the connection is then closed with a four-way handshake (FIN, ACK, FIN, and ACK).
- **HTTP.** Probes issue an HTTP GET with a specified path.

- **HTTPS.** Probes are similar to HTTP probes, except that a TLS/SSL wrapper is used. HTTPS probes are only supported on the standard-tier Load Balancer.

All three probe types must also specify the probe port or the interval. The minimum probe interval is five seconds in length, and the minimum consecutive probe failure threshold is two seconds. For HTTP and HTTPS probes, the probe path must also be given.

An endpoint is marked as unhealthy in the following settings:

- For HTTP or HTTPS probes only, the endpoint returns an HTTP status code other than `200 OK`.
- The probe endpoint closes the connection using a TCP reset.
- The probe endpoint fails to respond during the timeout period `snf` for a consecutive number of requests. The number of failed requests required to mark the endpoint unhealthy is configurable.

Configuring a dedicated health check page, such as `/healthcheck.php`, enables each backend server to implement custom application logic to decide whether it is healthy. Checking the availability of a backend database is an example of this.

When configuring network security groups (NSGs) for backend servers, it is important to allow both inbound traffic and probe traffic. Azure Load Balancer does not modify the source IP address of inbound traffic, so inbound traffic rules should be configured as if the Load Balancer was not in use. Approved-listing inbound probe traffic is achieved by allowing traffic originating from the `AzureLoadBalancer` service tag.

Note Load balancers and Network Security Groups

Standard-tier Load Balancers use standard-tier public IP addresses, which by default, are closed to inbound traffic. When using a standard-tier Load Balancer, traffic *must* be approved using NSGs. In contrast with basic-tier Load Balancers, traffic *should* be approved using NSGs but will also flow if NSGs are not used.

Configure an internal or public load balancer

As discussed earlier, both internal and public Load Balancers involve the coordinated configuration of several groups of settings. These settings work together to define the overall Load Balancer behavior.

Create an Azure Load Balancer using the Azure portal

To use the Azure Load Balancer, the administrator must first provision the resource, which includes the frontend IP configuration. After this step has been completed, you can create the backend pool, the health probes, and finally the load balancing rule.

To create the Load Balancer in Azure portal, click **+Create A Resource > Networking > Load Balancer**. This will open the **Create Load Balancer** blade, as shown in [Figure 4-72](#). Complete the blade as follows:

- **Name.** Provide a name for the Load Balancer resource.
- **Type.** Choose **Public or Internal**.
- **SKU.** Select the pricing tier: **Basic or Standard**.
- **Public IP Address Load Balancer.** (Public Load Balancers only.) Choose an existing public IP address resource or create a new one. Standard-tier Load Balancers must use standard-tier public IP addresses.

Create load balancer

Basics Tags Review + create

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. [Learn more](#).

Project details

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

ExamRef-RG

[Create new](#)

Instance details

Name *

ExamRef-LB

Region *

(Canada) Canada Central

Type * ⓘ

Internal Public

SKU * ⓘ

Basic Standard

Public IP address

Public IP address * ⓘ

Create new Use existing

Public IP address name *

ExamRef-LB-ip

Public IP address SKU

Basic

Assignment *

Dynamic Static

Add a public IPv6 address ⓘ

No Yes

[Review + create](#)

< Previous

Next : Tags >

[Download a template for automation](#)

FIGURE 4-72 Creating a Public Load Balancer with the Azure portal

- **Virtual Network, Subnet And IP Assignment.** (Internal Load Balancers only.) Choose the virtual network and subnet from which the frontend IP address will be allocated and choose between static and dynamic allocation (see [Figure 4-73](#)).
- **Availability Zone and Load Balancer.** (Standard-tier Load Balancers only.) For public Load Balancers, the availability zone is configured as part of the public IP address configuration. For internal Load Balancers, it is explicitly specified. (See [Figure 4-73](#).)
- **Subscription, Resource Group, And Location.** Specify as required.

Create load balancer

Basics Tags Review + create

Azure load balancer is a layer 4 load balancer that distributes incoming traffic among healthy virtual machine instances. Load balancers uses a hash-based distribution algorithm. By default, it uses a 5-tuple (source IP, source port, destination IP, destination port, protocol type) hash to map traffic to available servers. Load balancers can either be internet-facing where it is accessible via public IP addresses, or internal where it is only accessible from a virtual network. Azure load balancers also support Network Address Translation (NAT) to route traffic between public and private IP addresses. [Learn more](#).

Project details

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

ExamRef-RG

[Create new](#)

Instance details

Name *

ExamRef-LB

Region *

(Canada) Canada Central

Type * ⓘ

Internal Public

SKU * ⓘ

Basic Standard

Standard Load Balancer is secure by default. This means Network Security Groups (NSGs) are used to explicitly permit and whitelist allowed traffic. If you do not have an NSG on a subnet or NIC of your virtual machine resource, traffic is not allowed to reach this resource. Please configure an NSG to ensure communication if needed. For outbound communication, an explicit outbound rule is needed. [Learn more about outbound connectivity](#)

Configure virtual network.

Virtual network * ⓘ

ExamRef-VNet

Subnet *

WorkloadSubnet (10.1.1.0/24)

[Manage subnet configuration](#)

IP address assignment *

Static Dynamic

Availability zone * ⓘ

Zone-redundant

[Review + create](#)

< Previous

Next : Tags >

[Download a template for automation](#)

FIGURE 4-73 Creating an Internal Load Balancer with the Azure portal

After the Load Balancer has been created, the next steps are to create the backend pool, the health probe, and finally the load-balancing rule.

To create a backend pool, open the Load Balancer blade in the Azure portal, and then click **Backend Pools** > **+Add**. This opens the **Add Backend Pool** blade, as shown in [Figure 4-74](#). Specify the backend pool **Name** and, for a standard Load Balancer, select the virtual machines (and their IP addresses) to include in the backend pool. For basic Load Balancers, you will need to choose between adding an individual virtual machine, an availability set, or a VM scale set (refer [Figure 4-74](#)).

Home > ExamRef-LB | Backend pools >

Add backend pool

ExamRef-LB

Name * ExamRef-Backend

Virtual network ExamRef-VNet (ExamRef-RG)

IP version IPv4

Associated to Virtual machines

You can only attach virtual machines that are in the same location and on the same virtual network as the loadbalancer. Virtual machines must have a basic SKU public IP or no public IP. All virtual machines must be in the same availability set.

Filter by name...

Location == canadacentral Virtual network == ExamRef-VNet Resource group == all Availability set == all

	Virtual machine	Resource group	IP Configuration	Availability set	Tags	Notes
<input checked="" type="checkbox"/>	ExamRef-Jump	ExamRef-RG	ipconfig1 (10.1.2.5)			
<input checked="" type="checkbox"/>	ExamRef-Work	ExamRef-RG	ipconfig1 (10.1.1.4)			

+ Add X Remove

Virtual machine ↑ IP Configuration ↑ Availability set ↑

No virtual machines selected

FIGURE 4-74 Creating a backend pool and adding virtual machines, using a Basic Load Balancer

To create a health probe, navigate to the Load Balancer blade and click **Health Probes > +Add**. This opens the **Add Health Probe** blade, as shown in [Figure 4-75](#). Specify the health probe name, together with the protocol, port, probe interval, and consecutive probe failures threshold.

Add health probe

ExamRef-LB

Name *

HealthProbe1



Protocol ⓘ

TCP



Port * ⓘ

80

Interval * ⓘ

5

seconds

Unhealthy threshold * ⓘ

2

consecutive failures

FIGURE 4-75 Creating a health probe in Azure Load Balancer

The final step is to configure a load balancing rule, which links the frontend IP configuration to the backend pool, specifying the health probe and other load balancing settings. From the Load Balancer blade, click **Load Balancing Rules** > **+Add**. This opens the **Add Load Balancing Rule** blade, as shown in Figure

4-76. Choose the frontend IP configuration, backend pool, and health probe selected earlier. For HTTP traffic, select **TCP**, specify port **80** for both the frontend and backend ports. Select **None** for **Session Persistence** and leave the **Idle Timeout** at the default value of 4 minutes.

Note Floating IP

The last setting, Floating IP (direct server return), is only recommended when load-balancing traffic for a SQL Server Always On Availability Group listener. For other scenarios, the Floating IP setting should be left disabled.

Add load balancing rule

ExamRef-LB

i A load balancing rule distributes incoming traffic that is sent to a selected IP address and port combination across a group of backend pool instances. Only backend instances that the health probe considers healthy receive new traffic.

Name *

LBRule1 

IP Version *

IPv4 IPv6

Frontend IP address * ⓘ

null (LoadBalancerFrontEnd) 

Protocol

TCP UDP

Port *

80 

Backend port * ⓘ

80 

Backend pool ⓘ

ExamRef-Backend (1 virtual machine) 

Health probe ⓘ

HealthProbe1 (TCP:80) 

Session persistence ⓘ

None 

Idle timeout (minutes) ⓘ

 4

Floating IP (direct server return) ⓘ

Disabled Enabled

OK

FIGURE 4-76 Creating a load balancing rule in Azure Load Balancer

The final step is to ensure NSGs are configured to allow incoming traffic and health probe traffic. With this in place, if the VMs added to the backend pool are configured with a web server, you should be able to connect to the public IP address of the Load Balancer and see the webpage.

Troubleshoot load balancing

Basic- and standard-tier Load Balancers also support additional diagnostic logs to enable common troubleshooting scenarios. These logs are different between the basic and standard tiers.

Basic-tier Load Balancer metrics and diagnostics

The basic tier Load Balancer provides the following diagnostic logs:

- **Alert event logs.** These logs record Load Balancer alert events. They are written whenever a Load Balancer alert is raised (max every 5 minutes).
- **Health probe logs.** These logs allow you to investigate the status of health probes for backend servers. They are written whenever there is a change in health probe status.
- **Metrics.** Used to track common Load Balancer metrics.

To enable basic-tier load-balancer logs, open the **Load Balancer** blade in the Azure portal, select **Diagnostic Logs**, and click **Turn On Diagnostics** to open the **Diagnostics Configuration** blade, as shown in [Figure 4-77](#).

Diagnostics setting

Save Discard Delete Provide feedback

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)

Diagnostic setting name *

LBDiagSetting1



Category details

Destination details

log

LoadBalancerAlertEvent

Retention (days)

0

LoadBalancerProbeHealthStatus

Retention (days)

10

Send to Log Analytics

Archive to a storage account

Info Retention only applies to storage account. Retention policy ranges from 1 to 365 days. If you do not want to apply any retention policy and retain data forever, set retention (days) to 0.

Info You'll be charged normal data rates for storage and transactions when you send diagnostics to a storage account.

Info Showing all storage accounts including classic storage accounts

Location

Canada Central

Subscription

Visual Studio Ultimate with MSDN



Storage account *

examrefrgdiag



Stream to an event hub

FIGURE 4-77 Configuring diagnostics logs in a basic-tier load balancer

Having configured the diagnostics logs, they can be downloaded for offline analysis or analyzed using Log Analytics.

Standard-tier Load Balancer metrics and diagnostics

The standard Load Balancer also supports diagnostics via metrics routed automatically to Azure Monitor. Available metrics include byte count, packet count, health probe status, SYN count (for new connections), and more. Azure monitor supports charting and alerting based on these metrics. In addition, they are exposed as *multi-dimensional* metrics, meaning that charts and alerts can be built using filtered views.

Skill 4.4: Monitor and troubleshoot virtual networking

Azure offers numerous features and services to enable you to monitor your network and investigate network issues. These features provide a wide range of diagnostic and alerting capabilities. A good understanding of the range of features available will enable you to investigate network issues quickly and effectively.

This section covers how to:

- [Monitor on-premises connectivity](#)
- [Configure and use Network Performance Monitor](#)
- [Use Azure Network Watcher](#)
- [Troubleshoot external networking](#)
- [Troubleshoot virtual network connectivity](#)

Monitor on-premises connectivity

Azure Network Performance Monitor (NPM) is a network monitoring solution for hybrid networks that enables you to monitor network connectivity and performance between various points in your network, both in Azure and on premises. It can provide reports of network performance and raise alerts when network issues are detected.

NPM provides three services:

- **Performance Monitor.** Used to monitor connectivity between various points in your network, both in Azure and on premises. You can monitor nodes at both ends, and you can gather data about connectivity, packet loss, latency, and available network paths.
- **Service Connectivity Monitor.** Used to monitor outbound connectivity from nodes on your network to any external service with an open TCP port, such as web sites, applications, or databases. This measures latency, response time, and packet loss, enabling you to determine whether poor performance is caused by network or application issues.
- **ExpressRoute.** Used to monitor end-to-end connectivity between your on-premises network and Azure over ExpressRoute. This service can auto-discover your ExpressRoute network topology. It can then track your ExpressRoute bandwidth utilization, packet loss, and latency. These are measured at the circuit, peering and Azure virtual network level.

NPM also provides a dashboard giving an overview of the network status, as well as detailed per-service charts and reports.

Configure and use Network Performance Monitor

NPM is a Log Analytics solution. Log Analytics agents are installed on each node used to measure network connectivity and performance. These agents perform synthetic transactions over either TCP or ICMP to measure network performance. Data gathered from these agents is channeled into a Log Analytics workspace. NPM analyzes this data to provide both reporting and alerting.

NPM can be installed from the Azure Marketplace (from the Azure portal, click **+Create A Resource** and search for **Network Performance Monitor**). It is also available from Network Watcher, an Azure service that acts as a hub for a wide range of network monitoring and diagnostic tools. You will be required to create a Log Analytics workspace or select an existing workspace to use as shown in [Figure 4-78](#). Be sure to deploy your Log Analytics workspace to one of the regions supported by Network Performance Monitor, as listed at

<https://docs.microsoft.com/azure/azure-monitor/insights/network-performance-monitor#supported-regions>.

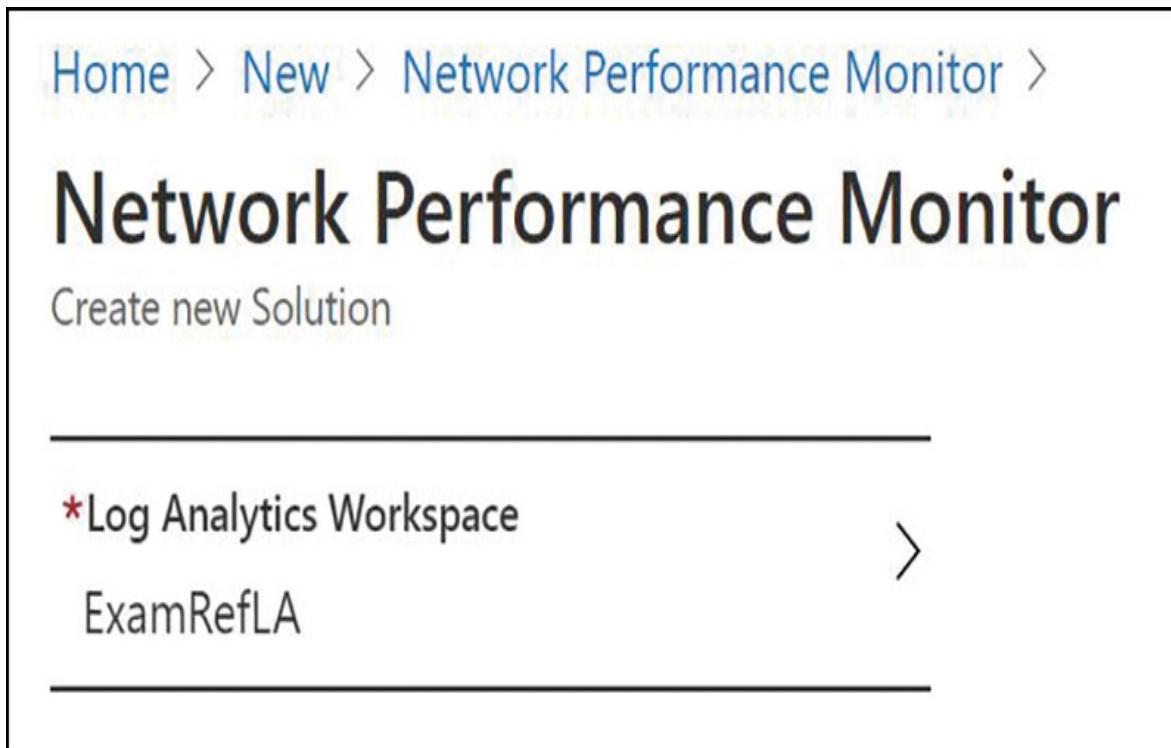


FIGURE 4-78 Create Network Performance Monitor

Having deployed NPM, the monitoring agents must be installed and configured. The choice of where to install the agents depends on your network topology and which parts of your network you plan to measure. To monitor a given network link, agents should be installed on servers at both ends of that link. To monitor connections between subnets, an agent on at least one server in each subnet is required.

To install the NPM monitoring agent on an Azure virtual machine, simply open the Log Analytics workspace, and click **Virtual Machines** (under **Workspace Data Sources**) to see a list of virtual machines and the status of their Log Analytics connection (Figure 4-79). From there, click a VM and click **Connect** to add the VM to Log Analytics. After a few minutes, refresh the list of virtual machines to see the updated list.

ExamRefLA | Virtual machines

Log Analytics workspace

Search (Ctrl+I) Refresh

Advanced settings

Name	OS	Subscription	Resource group	Location	
ExamRef	Not connected	Windows	00b72028-9dce-4729-9b2a-a10e92054447	ExamRef-RG	canadacentral
ExamRef-Jump	Connecting	Windows	00b72028-9dce-4729-9b2a-a10e92054447	ExamRef-RG	canadacentral
ExamRef-Work	This workspace	Windows	00b72028-9dce-4729-9b2a-a10e92054447	ExamRef-RG	canadacentral

General

- Quick Start
- Workspace summary
- View Designer
- Workbooks
- Logs
- Solutions
- Saved searches
- Pricing tier
- Usage and estimated costs
- Properties
- Service Map

Workspace Data Sources

- Virtual machines
- Storage accounts logs
- Azure Activity log
- Scope Configurations (Preview)

FIGURE 4-79 Connecting Azure Virtual Machines to a Log Analytics workspace

To connect on-premises servers with Log Analytics, you need to install the Log Analytics agent. Open the Log Analytics Workspace and click **Solutions**. Select the NPM solution and click the **Solution Requires Additional Configuration** tile, as shown in [Figure 4-80](#).

NetworkMonitoring(ExamRefLA)

Solution

Search (Ctrl+ /) Delete

Overview Essentials ▲

Activity log Resource group
examref-rg

Access control (IAM) Status
Active

Diagnose and solve problems Location
Canada Central

Subscription name (change)
Visual Studio Ultimate with MSDN

Settings Subscription ID
00b72028-9dce-4729-9b2a-a10e92054447

Locks

Export template

General

Workbooks

Log Analytics Workspace

Properties

Saved searches

Workspace Data Sources

Virtual machines

Azure Activity log

Solution Targeting (Preview)

Support + troubleshooting

Resource health

New support request

Summary

Network Performance Monitor Solution Resources

1

NetworkMonitoring(ExamRefLA)

FIGURE 4-80 Solution Requires Additional Configuration tile in Network Performance Monitor

Here, you will find options to download and install the Log Analytics agent, the workspace IDs and keys needed to configure the agent, and a PowerShell script to open the necessary firewall ports, as shown in [Figure 4-81](#).

Network Performance Monitor Configuration

ExamRefLA

X

⟳ Refresh ⚡ Logs

SETUP
NETWORKS (1)
SUBNETWORKS (1)
NODES (1)
PERFORMANCE MONITOR
SERVICE CONNECTIVITY MONITOR
EXPRESSROUTE MONITOR (0)

Common Settings **Performance Monitor** Service Connectivity Monitor ExpressRoute Monitor

Setup OMS Agents

1. Install OMS Agents

OMS agents are required to be installed on your computers in order to work with the following capabilities of Network Performance Monitor

① Service Connectivity Monitor: Install an OMS agent on each node from which you want to monitor the network connectivity to the service/application endpoint. [Learn More](#)

② Performance Monitor: Install an OMS agent on at least one node connected to each subnetwork from which you want to monitor network connectivity to other subnetworks. [Learn More](#)

③ ExpressRoute Monitor: Install at least one OMS agent in your Azure VNET and in your on-premises subnetwork which are connected through the ExpressRoute Private Peering. [Learn More](#)

[Download Windows Agent \(64-bit\)](#) You'll need the Workspace ID and Key to install the agent.

Workspace ID 21246a61-86d3-4065-a8f4-778d37e72e91

[Download Windows Agent \(32-bit\)](#) Primary Key DxnpVS2EVJ7OTBwpfKqPMCN03FVzvRT2

Secondary Key HPr+QIA4nWifFa8X7kqMjyozgfOh4hRW

OMS Gateway

If you have machines with no internet connectivity to OMS, download the OMS Gateway to act as a proxy. [Learn more](#).

[Download OMS Gateway](#)

2. Configure OMS agents for monitoring using TCP protocol

Firewall ports are required to be opened on the servers so that the agents can connect to each other. Run the script without any parameters in a power shell window with administrative privilege. This script creates few registry keys required by NPM and creates windows firewall rules to allow agents to create TCP connections with each other. The port opened by default would be 8084. You have the option of using a custom port by providing the parameter 'portNumber' to the script. However, the same port should be used on all the machines where the script is executed.

Note that the script will configure only windows firewall locally. If you have a network firewall you should make sure that it is allowing traffic destined for the TCP port being used by NPM

Note that for Service Connectivity Monitor, you do not need to run the below powershell script.

[Download Powershell Script](#)

SCOM Setup

① For SCOM setup, once the NPM solution is enabled on your OMS workspace the required management packs for NPM will automatically flow down to the machines that are connected to OMS via SCOM

② In case you want to connect SCOM with OMS but haven't figured out how to do it yet, click on the link below.

<https://technet.microsoft.com/en-us/library/mt484104.aspx>

FIGURE 4-81 Network Performance Monitor Configuration

Having installed and configured the agents, ensure that Network Security Groups and on-premises firewalls are configured to allow the agents to communicate. The default port used is TCP 8084.

Finally, in the left-navigation pane, complete the **Network**, **Subnetworks**, and **Nodes** sections to describe your network topology, as shown in [Figure 4-82](#). This allows you to define the networks and subnets in your network and identify which monitoring nodes sit within each network segment.

Network Performance Monitor Configuration

X

ExamRefLA

[Refresh](#) [Logs](#)

SETUP	Search by Subnetwork Id or Description X Y		
NETWORKS (1)			
SUBNETWORKS (1)	<input checked="" type="checkbox"/> SUBNETWORK ID <input checked="" type="checkbox"/> 10.1.2.0/24	DESCRIPTION	10.1.2.0/24 <input type="checkbox"/> Use for Monitoring
NODES (1)	UNMONITORED NODES (0)		
PERFORMANCE MONITOR	MONITORED NODES (1)		
SERVICE CONNECTIVITY MONITOR	Filter by IP or Node FQDN X Y <input checked="" type="checkbox"/> IP <input checked="" type="checkbox"/> 10.1.2.5 NODE FQDN ExamRefJump		
EXPRESSROUTE MONITOR (0)			
	Add + Remove ←		
	100 items per page 1-1 of 1 items		
	Save		
	100 items per page 1-1 of 1 items		

FIGURE 4-82 Network Performance Monitor Network and Subnetwork Configuration

Performance Monitor

Performance Monitor enables you to monitor packet loss and latency between your endpoints, both in Azure and on-premises. A VM or server running the Log Analytics agent is required at both ends of each monitored connection.

To configure Performance Monitor, first complete the Performance Monitor tab in the Setup section of the Network Performance Monitor Configuration blade. This allows you to specify TCP or ICMP-based monitoring.

Next, use the Performance Monitor section to define your monitoring rules. Each rule requires you to specify the source and destination networks, and the network protocol. You can also choose whether to enable health monitoring events based on defined criteria and whether to raise alerts based on those events. An example Performance Monitor rule is shown in [Figure 4-83](#).

Once configured, Performance Monitor will continually gather data from the Log Analytics agents, enabling both reporting and alerts.

Network Performance Monitor Configuration

ExamRefLA

⟳ Refresh 🗑 Logs

SETUP

Add Rule **Remove Rule**

Search by Rule Name or Description X Y

RULE NAME: Default

DESCRIPTION: Default Rule

Enable Rule

MONITOR

Monitor Connectivity Between

All Networks ▾ All Subnetworks ▾ And All Networks ▾ All Subnetworks ▾

EXCEPTIONS

Links between these entries will not be monitored

PROTOCOL (Help me choose)

ICMP TCP

HEALTH MONITORING

Enable Health Monitoring on the links covered by this rule

A health event will be logged when following conditions are met

CONDITION

Loss greater than equal to % Auto Detect Sudden Changes

Latency greater than equal to ms Auto Detect Sudden Changes

Save

FIGURE 4-83 Example Performance Monitor Rule Configuration

Service Connectivity Monitor

Service Connectivity Monitor is used to test outbound connectivity from your network to open a TCP port, such as a website, application, or database. It supports pre-configured endpoints for Microsoft Office365 and Dynamics. You can also configure custom tests to arbitrary endpoints.

To use the pre-configured endpoints, select the **Service Connectivity Monitor** tab from the **Setup** section of the **Network Performance Monitor Configuration** blade, as shown in [Figure 4-84](#). Select the services to monitor, click **+Add Agents** to choose which of your network nodes should monitor these services, and then click **Save & Continue**.

Network Performance Monitor Configuration

X

ExamRefA

Refresh Logs

SETUP

Common Settings Performance Monitor Service Connectivity Monitor ExpressRoute Monitor

NETWORKS (1)

SUBNETWORKS (1)

NODES (1)

PERFORMANCE MONITOR (1)

SERVICE CONNECTIVITY MONITOR (0)

EXPRESSROUTE MONITOR (0)

Office 365 Monitoring

Select service(s) to monitor

Search by Service name X Y

O365 SERVICE NAME

Office 365 portal and shared

Office 365 authentication and identity

Office Online

Exchange Online

Skype for Business Online

Microsoft Teams

Planner

Sway

Office 365 Video and Microsoft Stream

1 2 3 4 5 6 7 8 9 100 items per page 1 - 9 of 9 items

Dynamics 365 Monitoring

Select organization location(s) to monitor

Search by Organization Location X Y

ORGANIZATION LOCATION

North America-based organizations

South America-based organizations

EMEA based organizations

Asia/Pacific area-based organizations

Japan area-based organizations

India area-based organizations

Canada area-based organizations

Oceania area-based organizations

CRM US Government environments

1 2 3 4 5 6 7 8 9 100 items per page 1 - 9 of 9 items

Select Agents to Monitor

+Add Agents

Select Agents to Monitor

+Add Agents

Save & Continue

Figure 4-84 Configuring Service Connectivity Monitor for Microsoft Services

Now move to the Service Connectivity Monitor section in the left-navigation pane, which shows the existing tests and allows you to configure custom tests. [Figure 4-85](#) shows a custom test to check the availability of the Azure management portal. See <https://portal.azure.com>.

Network Performance Monitor Configuration

ExamRefLA

 [Refresh](#)
 [Logs](#)

SETUP	Add Test	Remove Test	
NETWORKS (1)	<input type="text" value="Search by Test Name or Description"/> X		
SUBNETWORKS (1)			
NODES (1)	TEST NAME	TEST DESCRIPTION	TEST TYPE
PERFORMANCE MONITOR (1)	Azure Portal Test	Test http://portal.azure.com	Custom Test
SERVICE CONNECTIVITY MONITOR (1)			
EXPRESSROUTE MONITOR (0)			

TEST NAME:
AZURE PORTAL TEST

DESCRIPTION
Test http://portal.azure.com

Enable Test

TYPE OF TEST

Web Network

HTTPS HTTP

Perform network measurements

TARGET
portal.azure.com

PORT NUMBER
443 Enter custom port number

TEST FREQUENCY
5 min

1 Agent Selected [+Add Agents](#)

Enable Health Monitoring in the targets covered by this test
A health event will be logged when following conditions are met

- Response Time greater than 250 ms
equal to
- Network Latency greater than 100 ms
equal to
- Network Loss greater than 5 %
equal to

ALERTS
Click on the link below to create alert notifications whenever a health event is generated for links covered by this rule
[Create Alerts](#)

[Save](#)

FIGURE 4-85 Configuring a custom test in Service Connectivity Monitor

Once configured, Service Connectivity Monitor will generate packet loss and network performance charts (showing latency and response times) for each tested endpoint.

ExpressRoute Monitor

ExpressRoute Monitor allows you to monitor end-to-end network connectivity and performance between on-premises and Azure endpoints over ExpressRoute connections. It can auto-detect ExpressRoute circuits and your network topology, and track bandwidth utilization, packet loss and network latency. Reports are available for each ExpressRoute circuit or peering, and also for each Azure virtual network using ExpressRoute.

To configure ExpressRoute Monitor, use the ExpressRoute Monitor section of the Network Performance Monitor Configuration blade (see [Figure 4-86](#)). First, ExpressRoute resources (such as gateways and circuits) are identified in your subscriptions. Next, the monitoring for each peering can be enabled, configuring health events and choosing monitoring agents.

Once configured, it takes 30-60 minutes for the first ExpressRoute reporting data to become available. Several reports and charts are available, including bandwidth utilization, latency, and packet loss for each ExpressRoute circuit and for each peering. A network topology view shows network connections and status. Log Analytics alerts can be configured for a wide range of events, such as high latency, packet drops, high and low utilization, and more.

Network Performance Monitor Configuration

ExamRefA

Refresh Logs

SETUP	Discovery	ExpressRoute Resources	Monitoring				
NETWORKS (1)	<p>To monitor your ExpressRoute (ER) resources:</p> <ol style="list-style-type: none"> ① From among the subscriptions you have access to, choose the ones that contain the ER circuits and associated peering connections you intend to monitor ② Trigger discovery of the ER resources by clicking on Discover ExpressRoute resources button below ③ Once the discovery is complete, move to the ExpressRoute Resources view by clicking on the tab above and choose the ER resources you intend to monitor <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <input type="text" value="Search by Subscription Name or Subscription Id"/> X Y </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">SUBSCRIPTION</th> <th style="width: 85%;">DISCOVERY STATUS</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> Visual Studio Ultimate with MSDN</td> <td>Discovery not started</td> </tr> </tbody> </table>			SUBSCRIPTION	DISCOVERY STATUS	<input checked="" type="checkbox"/> Visual Studio Ultimate with MSDN	Discovery not started
SUBSCRIPTION	DISCOVERY STATUS						
<input checked="" type="checkbox"/> Visual Studio Ultimate with MSDN	Discovery not started						
SUBNETWORKS (1)							
NODES (1)							
PERFORMANCE MONITOR (1)							
SERVICE CONNECTIVITY MONITOR (1)							
EXPRESSROUTE MONITOR (0)							

FIGURE 4-86 Configuring ExpressRoute Monitor

Use network resource monitoring

Earlier in this chapter, you saw how Azure Application Gateways and Azure Load Balancers emit diagnostic logs, which can be used for detailed insight into

the status of each service. These logs can be captured in a storage account, streamed to an EventHub, or integrated with an Azure Log Analytics workspace, which enables customized queries and log-based alerting. In the case of App Gateway, you also saw how the Azure Application Gateway Analytics Log Analytics solution provides a pre-configured dashboard and charts showing App Gateway status.

Diagnostic logs are also available for a number of other networking resources, including Traffic Manager, Azure DNS, and Network Security Groups. In each case, they give deeper insight into the status and operation of each service, as well as supporting log-based alerts through Log Analytics. In the case of NSGs, the Traffic Analytics Log Analytics solution provides detailed reports giving insight into the successful and blocked traffic flows into and out of your Azure services.

Use Azure Network Watcher

Network Watcher provides a central hub for a wide range of network monitoring and diagnostic tools. These tools are valuable across a wide range of network troubleshooting scenarios, and also provide access to other tools listed in this skill section, such as the Network Performance Monitor and Connection Monitor.

Deploying Network Watcher

Network Watcher is enabled as a single instance per Azure region. It is not deployed like a conventional Azure resource, although it does appear as a resource in a resource group.

Any subscription containing a virtual network resource will automatically have Network Watcher enabled. Otherwise, it can be enabled via the Azure portal, under **All Services > Network Watcher**. Also, you can see the Network Watcher status per region. Network Watcher can also be deployed via the command line (using the `New-AzNetworkWatcher` cmdlet or the `az network watcher configure` commands), which unlike the Azure portal, provides control over the resource group used.

Some of the Network Watcher tools require the Network Watcher VM extension to be installed on the VM being monitored. This extension is available for both Windows and Linux VMs. It is installed automatically when using

Network Watcher via the Azure portal.

IP Flow Verify

The IP Flow Verify tool provides a quick and easy way to test whether a given network flow will be allowed into or out of an Azure virtual machine. It will report whether the requested traffic is allowed or blocked, and in the latter case, which NSG rule is blocking the flow. It is a useful tool for verifying that NSGs are correctly configured.

It works by simulating the requested packet flow through the NSGs applied to the VM. For this reason, the VM must be in a running state.

To use IP Flow, verify via the Azure portal, open Network Watcher, and click **IP Flow Verify**. Select the VM and NIC to verify, and specify the protocol, direction, and remote and local IP addresses and ports, as shown in [Figure 4-87](#).

Home >

Network Watcher | IP flow verify

Microsoft

 Search (Ctrl+ /) <<



Overview

Monitoring



Topology



Connection monitor



Connection monitor (Preview)



Network Performance Monitor

Network diagnostic tools



IP flow verify



Next hop



Effective security rules



VPN troubleshoot



Packet capture



Connection troubleshoot



Metrics



Usage + quotas

Logs



NSG flow logs



Diagnostic logs



Traffic Analytics

Network Watcher IP flow verify checks if a packet is allowed or denied to or from a virtual machine based on 5-tuple information. The security group decision and the name of the rule that denied the packet is returned.

[Learn more.](#)

Specify a target virtual machine with associated network security groups, then run an inbound or outbound packet to see if access is allowed or denied.

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Resource group * ⓘ

ExamRef-RG

Virtual machine * ⓘ

ExamRef

Network interface *

examref913

Packet details

Protocol

TCP UDP

Direction

Inbound Outbound

Local IP address * ⓘ

10.0.0.5 ✓

Local port * ⓘ

23001

Remote IP address * ⓘ

10.0.1.4 ✓

Remote port * ⓘ

443

Check

FIGURE 4-87 Using Network Watcher IP Flow Verify

IP Flow verify can also be used from PowerShell, using the `Test-AzNetworkWatcherIPFlow` cmdlet, or the Azure CLI, using the `az network watcher test-ip-flow` command.

Next Hop

The Next Hop tool provides a useful way to understand how a VM's outbound traffic is being directed. For a given outbound flow, it shows the next hop IP address and type and the route table ID of any user-defined route in effect. Possible next hop types are

- Internet
- VirtualAppliance
- VirtualNetworkGateway
- VirtualNetwork
- VirtualNetworkPeering
- VirtualNetworkServiceEndpoint
- None (this is used for user-defined routes)

To use Next Hop via the Azure portal, open Network Watcher and click **Next Hop**. Select the source VM, NIC and IP address, and the destination address, as shown in [Figure 4-88](#). The destination can be any IP address, either on the internal network or the Internet.

Home >

Network Watcher | Next hop

Microsoft

Search (Ctrl+ /)

«

Next Hop provides the next hop from the target virtual machine to the destination IP address.

[Learn more.](#)



Monitoring

Specify a target virtual machine and destination IP address to view the next hop.



Topology

Subscription * ⓘ

Visual Studio Ultimate with MSDN

▼



Connection monitor

Resource group * ⓘ

ExamRef-RG

▼



Connection monitor (Preview)



Network Performance Monitor

Virtual machine * ⓘ

ExamRef

▼

Network diagnostic tools



IP flow verify



Next hop

Network interface *

examref913

▼



Effective security rules

Source IP address * ⓘ

10.0.0.5

✓



VPN troubleshoot

Destination IP address * ⓘ

20.48.147.87

✓



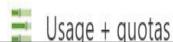
Packet capture



Connection troubleshoot

Metrics

[Next hop](#)



Usage + quotas

FIGURE 4-88 Using Network Watcher Next Hop

Next Hop can also be used from PowerShell using the `Get-AzNetworkWatcherNextHop` cmdlet, or the Azure CLI using the `az network watcher show-next-hop` command.

Packet Captures

The Packet Capture tool allows you to capture network packets entering or leaving your virtual machines. It is a powerful tool for deep network diagnostics.

You can capture all packets, or a filtered subset based on the protocol and local and remote IP addresses and ports. You can also specify the maximum packet and overall capture size, and a time limit (captures start almost immediately once configured).

Packet captures are stored as a file on the VM or in an Azure storage account, in which case NSGs must allow access from the VM to Azure storage. These captures are in a standard format and can be analyzed off-line using common tools such as Wireshark or Microsoft Message Analyzer.

To use the Packet Capture tool, open Network Watcher and click **Packet Capture** > **+Add**. Select the VM, give the capture a name, and specify the destination, packet and total size, time limit, and filters. An example is shown in [Figure 4-89](#).

Add packet capture

X

Subscription *

Visual Studio Ultimate with MSDN



Resource group *

ExamRef-RG



Target virtual machine *

ExamRef



Packet capture name *

PacketCapture1



Capture configuration

The packet capture output file (.cap) can be stored in a storage account and/or on the target VM.

Storage account File Both

Storage accounts *

examrefrgdiag



Maximum bytes per packet ⓘ

1024



Maximum bytes per session ⓘ

5000000



Time limit (seconds) ⓘ

300



Filtering (optional)

Protocol *

Any TCP UDP

Local IP address ⓘ

10.0.0.4



Local port ⓘ

80;443



Remote IP address ⓘ

"127.0.0.1", "127.0.0.1-127.0.0.255" or "127.0.0.1;127.0.0.5"



Save

Cancel

FIGURE 4-89 Using Network Watcher Packet Capture

Network Topology

The Network Topology view in Network Watcher provides a diagrammatic view of the resources in your virtual network. It is not a diagnostic or alerting tool. It is a quick and easy way to review your network resources and manually check for misconfiguration.

A limitation of the tool is that it only shows the topology within a single virtual network. All common network resource types are supported, although for application gateways, only the backend pool connected to the network interface is shown.

To use Network Topology via the Azure portal, open Network Watcher and click **Topology**. Select the resource group and virtual network, and the topology will be shown.

An example topology is given in [Figure 4-90](#).



FIGURE 4-90 Using Network Watcher Network Topology

The underlying topology data can be downloaded in JSON format via Azure PowerShell or the Azure CLI, using the `Get-AzNetworkWatcherTopology` cmdlet or the `az network watcher show-topology` command, respectively.

Troubleshoot external networking

We have already seen how the Network Performance Monitor provides a range of powerful features to monitor and diagnose issues across both Azure and on-premises networks, including detailed analytics for ExpressRoute connections.

Another pair of useful tools to investigate issues with external networks are the Connection Monitor and Connection Troubleshoot tools in Network Watcher. These are discussed in the next section: “[Troubleshoot virtual network connectivity](#).”

In this section, we discuss VPN Troubleshoot, which is designed specifically to diagnose problems with VPN connections.

Remember, for simple validation that a VPN connection is working, it’s also always worthwhile trying to connect between VMs on either end of VPN tunnel using standard tools such as `tcping`.

VPN Troubleshoot

The VPN Troubleshoot feature in Network Watcher provides automated diagnostics of Azure VPN gateways and connections. The results provide a detailed report on gateway health and connection health, providing accurate pointers regarding common issues that might occur when enabling informed remediations.

VPN Troubleshoot only supports route-based VPN gateways (not policy-based gateways or ExpressRoute gateways). It supports both IPsec Site-to-Site VPNs and VNet-to-VNet connections; it does not support ExpressRoute connections or Point-to-Site connections.

During the troubleshooting process, logs are written to a storage account. This account must be created before starting the troubleshooting process.

To use VPN Troubleshoot via the Azure portal, open **Network Watcher** and

click **VPN Troubleshoot**. Select the storage container for the troubleshooting logs and then select which VPN resources to troubleshoot. Finally, click **Start Troubleshooting**.

The troubleshooting process takes a few minutes to run. Once complete, the results will be shown at the bottom of the page.

Troubleshoot virtual network connectivity

A number of the tools we have already seen can be useful for troubleshooting connectivity issues between and within virtual networks. Network Watcher offers two more tools that are particularly useful in this scenario: Connection Troubleshoot and Connection Monitor.

Connection Troubleshoot

Connection Troubleshoot is a Network Watcher feature designed to allow you to test the connectivity between an Azure VM or an App Gateway and another endpoint—either another Azure VM, or an arbitrary Internet or Intranet endpoint. This diagnostic tool can identify a range of problems, including guest VM issues, such as guest firewall configuration, low memory or high CPU, Azure configuration issues such as Network Security Groups blocking traffic, or routing issues diverting traffic. It can also diagnose other network issues, such as DNS failures.

To use Connection Troubleshoot from the Azure portal, open **Network Watcher** and then click **Connection Troubleshoot**. Specify the source VM, then specify the destination, either as another VM or by giving a URI, FQDN, or IPv4 address. Specify the protocol to use (either TCP or ICMP). For TCP, you can specify the destination port, and, under **Advanced Settings**, the source port. An example configuration is shown in [Figure 4-91](#).

Home >

Network Watcher | Connection troubleshoot

Microsoft

«

 Overview

 Monitoring

 Topology

 Connection monitor

 Connection monitor (Preview)

 Network Performance Monitor

 Network diagnostic tools

 IP flow verify

 Next hop

 Effective security rules

 VPN troubleshoot

 Packet capture

 Connection troubleshoot

 Metrics

 Usage + quotas

 Logs

 NSG flow logs

 Diagnostic logs

 Traffic Analytics

Network Watcher Connection Troubleshoot provides the capability to check a direct TCP connection from a virtual machine (VM) to a VM, fully qualified domain name (FQDN), URI, or IPv4 address. To start, choose a source to start the connection from, and the destination you wish to connect to and select "Check".

[Learn more.](#)

Source

Subscription *

Visual Studio Ultimate with MSDN

Resource group *

ExamRef-RG

Source type *

Virtual machine

Virtual machine *

ExamRef-Jump

Destination

Select a virtual machine Specify manually

URI, FQDN or IPv4 *

azure.microsoft.com

Probe Settings

Protocol

TCP ICMP

Destination port *

443

Advanced settings

Source port

Check

Checking connectivity....

FIGURE 4-91 Network Watcher Connection Troubleshoot configuration

The test takes a few minutes to run. Upon completion, the results will be shown at the bottom of the page. An example output is shown in [Figure 4-92](#).

Connection Troubleshoot is also available via PowerShell using the `Test-AzNetwork WatcherConnectivity` cmdlet and via the Azure CLI using the `az network watcher test-connectivity` command.

Status

✓ Reachable

Agent extension version

1.4

Source virtual machine

ExamRef-Jump

[Grid view](#) [Topology view](#)

Hops

Name	IP address	Status	Next hop IP address	RTT f
ExamRef-Jump	10.1.2.5	✓	52.228.83.164	3
Destination (por...	52.228.83.164	✓	-	-

Average Latency in milliseconds

2

Minimum Latency in milliseconds

2

Maximum Latency in milliseconds

3

Probes Sent

66

Probes Failed

0

FIGURE 4-92 Network Watcher Connection Troubleshoot results

Connection Monitor

The Connection Monitor in Network Watcher is similar to Connection Troubleshoot, in that it uses the same mechanism to test the connection between an Azure VM or App Gateway and another endpoint. The difference is that Connection Monitor provides ongoing connection monitoring, whereas Connection Troubleshoot only provides a point-in-time test.

Data from Connection Monitor is surfaced in Azure Monitor. Charts show key metrics such as round-trip time and probe failures. Azure Monitor can also be used to configure alerts, triggered by connection failures or a drop in performance.

To use Connection Monitor via the Azure portal, open **Network Watcher** and click **Connection Monitor**. A list of active monitored connections is shown. Click **+Add** to create a new monitored connection and then fill in the connection settings. The settings are almost the same as for Connection Troubleshoot. Also, you will need to specify the probing interval in seconds. An example is shown in [Figure 4-93](#).

The monitored connection will be listed on the Connection Monitor blade within Network Watcher. Click a monitored connection to open the results panel, as shown in [Figure 4-94](#). The chart shows average round-trip time and percentage of probe failures. Click the chart to view the data in Azure Monitor. From there, alerts can be configured based on these metrics exceeding thresholds you define. The table below the chart shows the current connection status—clicking each line provides further details about the status, which is similar to how the results obtained from Connection Troubleshoot are shown.

Add connection monitor

X

Name *

JumpVM-to-WorkVM



Source

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Virtual machine *

ExamRef-Jump



Destination

Select a virtual machine Specify manually

Virtual machine * ⓘ

ExamRef-Work



Port * ⓘ

443



Advanced settings

Source port ⓘ

Probing interval (seconds) ⓘ

Add

FIGURE 4-93 Network Watcher Connection Monitor configuration

Status

⚠ Unreachable

Agent extension version

1.4

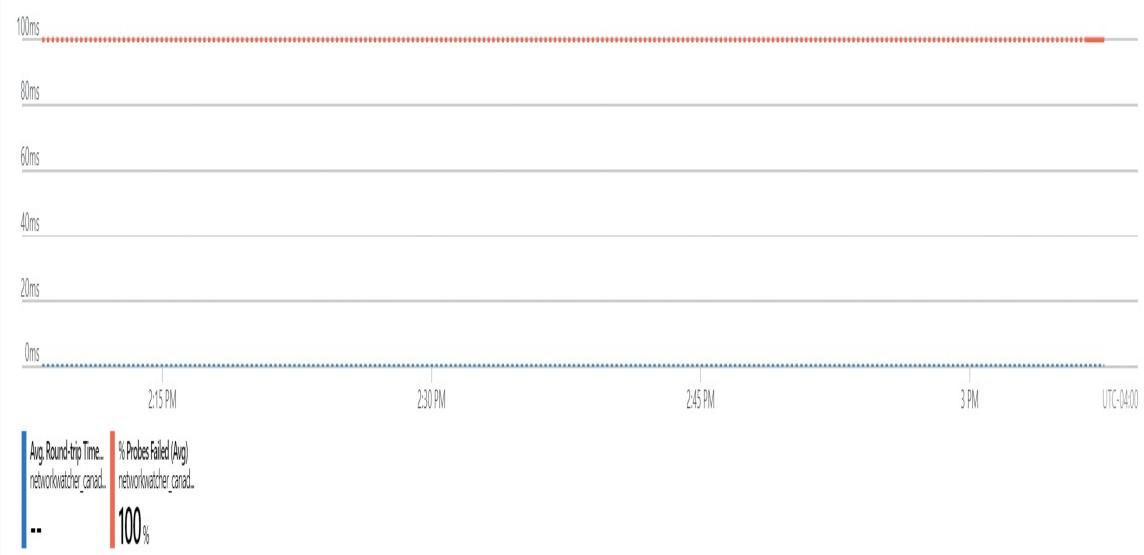
Source virtual machine

ExamRef-Jump

Show data for last:

1 hour 6 hours 12 hours 1 day 7 days 30 days

Avg. Round-Trip Time and % Probes Failed



Grid view Topology view

Hops

Name	IP address	Status	Next hop IP address	RTT from source (ms)
ExamRef-Jump	10.1.2.5	🟢	10.1.1.4	-
ExamRef-Work	10.1.1.4	🔴	-	-

FIGURE 4-94 Network Watcher Connection Monitor status

Skill 4.5: Integrate an on-premises network with an Azure virtual network

Many Azure deployments require connectivity between the on-premises network and the Azure VNet. This integrated network is called a *hybrid network*.

Hybrid networks are commonly used for Intranet applications, which may be hosted in Azure but only accessed from the on-premises network. They are also used by Azure applications that require access to an on-premises resource, such as a database.

Hybrid networks provide connectivity between the private IP space of the on-premises network and the private IP space of the Azure VNet. The VNet can be thought of as an extension of the existing on-premises network. The concept is similar to extending the on-premises network to a new office location.

This section covers how to:

- [Create and configure Azure VPN Gateway](#)
- [Create and configure Azure ExpressRoute](#)
- [Configure Azure Virtual WAN](#)

Create and configure Azure VPN Gateway

A virtual network gateway allows you to create connections from your virtual network to other networks. When creating a gateway, you must specify if it will be used for VPN connections or ExpressRoute connections. Virtual network gateways used for VPN connections are called a VPN gateway, while those used for ExpressRoute connections are called ExpressRoute gateways.

Earlier in this chapter we saw how VPN gateways can be used to connect one Azure VNet to another. They can also be used to create VPN tunnels between Azure VNets and on-premises networks—this is called a site-to-site VPN. They can also be used as a hub for point-to-site networks, where individual machines

connect to an Azure VNet via the VPN client on the machine.

Gateway subnets

VPN gateways can only be deployed to a dedicated gateway subnet within the VNet. A gateway subnet is a special type of subnet that can only be used for virtual network gateways. Under the hood, the VPN gateway is implemented using Azure virtual machines (these are not directly accessible and are managed for you). While the minimum size for the gateway subnet is a CIDR /29, the Microsoft-recommended best practice is to use a CIDR /27 address block to allow for future expansion.

A VPN connection between an on-premises network and an Azure VNet can only be established if the network ranges do not overlap. Network address ranges should be planned carefully to avoid restricting future connectivity options.

Gateway SKUs

VPN Gateways are available in several pricing tiers, or SKUs. The correct tier should be chosen based on the required network capacity, as shown in [Table 4-10](#).

TABLE 4-10 Comparison of VPN Gateway Pricing Tiers

SKU	Max Site-to-Site VPN Connections	Throughput
Basic	10	100 Mbps
VpnGw1 and VpnGw1Az	30	650 Mbps
VpnGw2 and VpnGw2Az	30	1 Gbps
VpnGw3 and VpnGw3Az	30	1.25 Gbps

Note Resizing VPN Gateways

You can resize a gateway between the VpnGw1, VpnGw2, and VpnGw3 tiers. You cannot, however, resize a Basic tier gateway.

BGP

Border Gateway Protocol (BGP) is a standard used in the Internet to exchange routing information between networks. BGP can be optionally enabled on your VPN gateway, if the on-premises gateway also supports it. If used, it enables the VPN gateway and the on-premises gateway to exchange routing information automatically, avoiding the need to configure routes manually.

BGP also enables high availability redundant connections (see next section) advanced features such as transit routing across multiple networks. It is also used where a VPN connection is used as a failover in case the primary connection, using ExpressRoute, were to fail.

High Availability

By default, each VPN gateway is deployed as two VMs in an active-standby configuration. To reduce downtime in the event the active instance fails, an active-active configuration can also be used (not supported for Basic SKU gateways). In this mode, both gateway instances have their own public IP addresses, and two connections are made to the on-premises VPN endpoint.

Dual on-premises VPN endpoints can also be used. This requires BGP to be enabled and works with both active-standby or active-active VPN gateways. Combining dual on-premises endpoints with active-active VPN gateways provides a fully redundant configuration, avoiding single points of failure, as shown in [Figure 4-95](#). In this configuration, traffic will be distributed over all four VPN tunnels.

For increased resilience to data center-level failures, virtual network gateways can be deployed to availability zones. This requires the use of dedicated SKUs, called VpnGw1Az, VpnGw2Az, and VpnGw3Az. Both zone-redundant and zone-specific deployment models are supported, the choice being inferred from the associated public IP address rather than being specified explicitly as a gateway property.

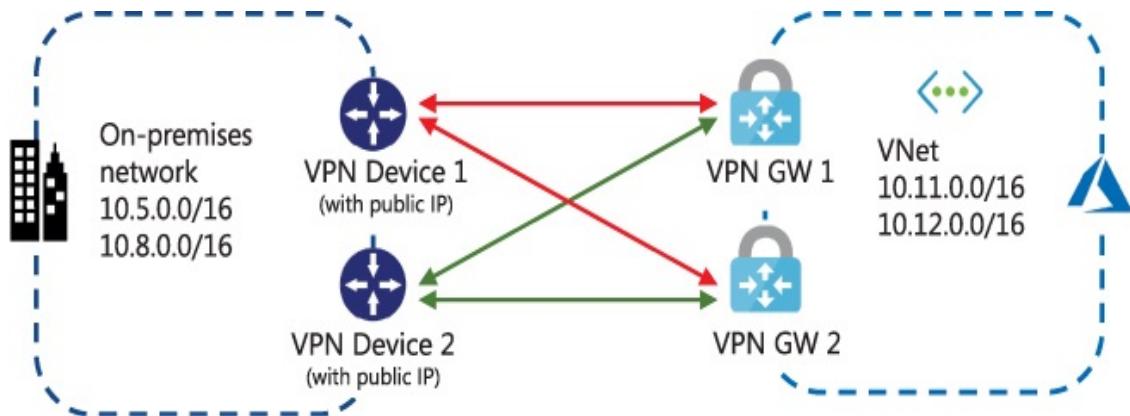


FIGURE 4-95 Dual on-premises VPN endpoints connected to active-active VPN gateways

Create a VPN Gateway using the Azure portal

Before creating the VPN gateway, first create the gateway subnet. Using the Azure portal, navigate to your virtual network and click the **Subnets** link under **Settings** to open the **Subnets** blade. Click the **+Gateway Subnet** button and assign an address space using a /27 CIDR, as seen in [Figure 4-96](#). Do not modify the other subnet settings.

Add subnet

X

ExamRef-VNet

Name

GatewaySubnet

Address range (CIDR block) * ⓘ

10.1.5.0/27



10.1.5.0 - 10.1.5.31 (27 + 5 Azure reserved addresses)

NAT gateway ⓘ

None



Add IPv6 address space

Network security group

None



Route table

None



Service endpoints

Services ⓘ

0 selected



Subnet delegation

Delegate subnet to a service ⓘ

None



FIGURE 4-96 Adding a Gateway Subnet to a virtual network

Next, provision a VPN gateway as follows. From the Azure portal, click **+Create A Resource > Networking > Virtual Network Gateway**. Complete the **Create Virtual Network Gateway**' blade as follows:

- **Name.** Choose **VNet-GW**.
- **Gateway type.** Choose **VPN**.
- **VPN Type.** Choose **Route-Based**.
- **SKU.** Choose **VpnGw1.Generation**. Choose **Generation1**.
- **Virtual Network.** Choose your VNet.
- **Gateway subnet address range.** Choose based on your configuration.
- **Public IP Address.** Choose **Create New** or **Use Existing**.
- **Location.** This is the same as your VNet.

Do not select the checkboxes for **Enable Active-Active Mode** or **Configure BGP ASN**. [Figure 4-97](#) shows the completed gateway settings.

Create virtual network gateway

Basics Tags Review + create

Azure has provided a planning and design guide to help you configure the various VPN gateway options. [Learn more.](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Visual Studio Ultimate with MSDN

Resource group ⓘ

ExamRef-RG (derived from virtual network's resource group)

Instance details

Name *

VNet-GW

Region *

(Canada) Canada Central

Gateway type * ⓘ

VPN ExpressRoute

VPN type * ⓘ

Route-based Policy-based

SKU * ⓘ

VpnGw1

Generation ⓘ

Generation1

Virtual network * ⓘ

ExamRef-VNet

[Create virtual network](#)

i Only virtual networks in the currently selected subscription and region are listed.

Gateway subnet address range * ⓘ

10.1.5.0/27

10.1.5.0 - 10.1.5.31 (32 addresses)

Public IP address

Public IP address * ⓘ

Create new Use existing

Choose public IP address *

ExamRef-ip

Enable active-active mode * ⓘ

Enabled Disabled

Configure BGP ASN * ⓘ

Enabled Disabled

Azure recommends using a validated VPN device with your virtual network gateway. To view a list of validated devices and instructions for configuration, refer to Azure's [documentation](#) regarding validated VPN devices.

[Review + create](#)

< Previous

Next : Tags >

[Download a template for automation](#)

FIGURE 4-97 Creating an Azure VPN Gateway

Note Gateway Subnets

When creating the gateway subnet, there is no special parameter or cmdlet name to denote that this is a gateway subnet rather than a normal subnet. The only distinction that identifies a gateway subnet is the subnet name, `GatewaySubnet`.

Create and configure Azure ExpressRoute

ExpressRoute is a secure and reliable private connection between your on-premises network and the Microsoft cloud. The connection is provided mostly by a third-party network provider who has partnered with Microsoft to offer ExpressRoute services. This third party is known as the *ExpressRoute provider*. Alternatively, with ExpressRoute Direct, you can connect directly to the MSEE.

Unlike a Site-to-Site VPN, network traffic using ExpressRoute uses your provider's network and does not pass over the Internet. Therefore, the latency and bandwidth for an ExpressRoute circuit is more predictable and stable because traffic stays on your provider's network. (An ExpressRoute connection is called a circuit.)

Another key difference between ExpressRoute connections and Site-to-Site VPN connections is that Site-to-Site VPN connections only provide connectivity to your Azure VNet, whereas ExpressRoute provides connectivity to all Microsoft cloud services. This includes Azure VNets, Azure platform services (such as CosmosDB), and Microsoft services outside of Azure such as Office 365 and Dynamics 365.

Connectivity models

ExpressRoute connectivity can be established in one of three ways. The capabilities and features of ExpressRoute are the same in each case.

- If your network already has a presence at a co-location facility with a cloud exchange, your co-location provider can establish a virtual cross-connection with the Microsoft Cloud. This provides either a layer 2 or a

managed layer 3 connection.

- Your connectivity provider may be able to provide a point-to-point ethernet connection from their network to your on-premises network. Again, this approach offers either a layer 2 or managed layer 3 connection.
- Finally, your existing IPVPN WAN provider may be able to integrate ExpressRoute into your WAN, if they are registered as an ExpressRoute provider. In this case, your provider will typically offer managed layer 3 connectivity.

These connectivity options are shown in [Figure 4-98](#).

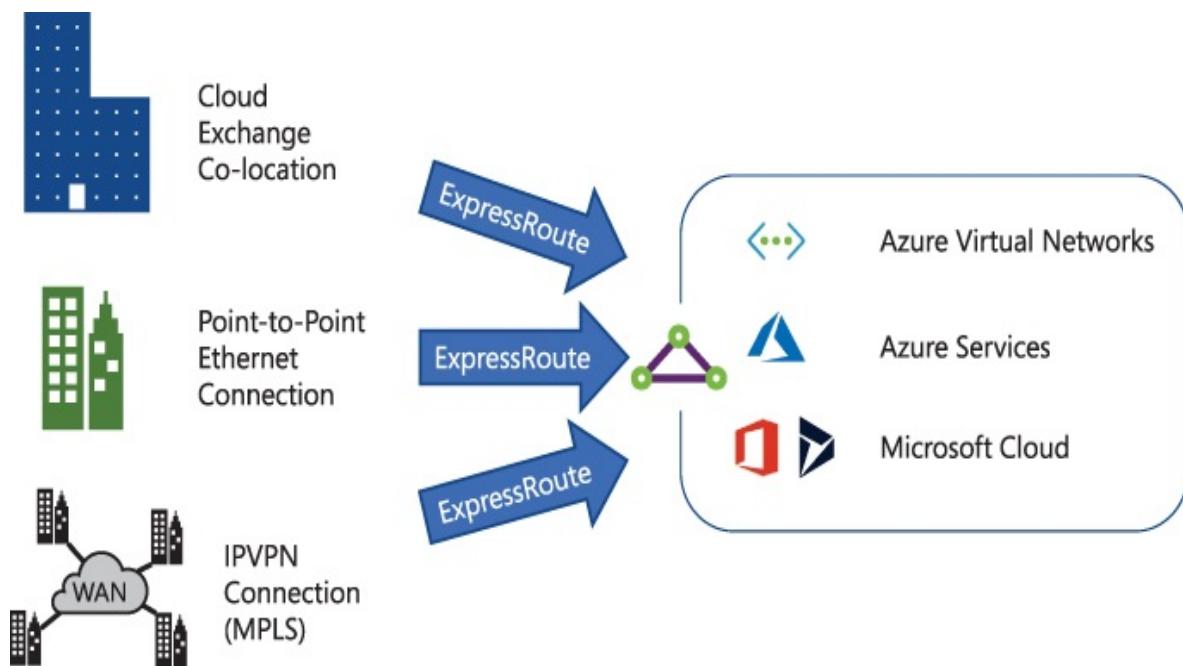


FIGURE 4-98 ExpressRoute connectivity models

Circuits and peering

An ExpressRoute circuit is an Azure resource used to represent the logical connection between your on-premises network and Microsoft. Each circuit is identified by a GUID called a *service key (s-key)*, which is shared with your connectivity provider.

Each circuit has a fixed bandwidth, and a specific peering location. The available bandwidth options are 50 Mbps, 100 Mbps, 200 Mbps, 500 Mbps, 1

Gbps, 2 Gbps, 5 Gbps, and 10 Gbps. This bandwidth can be either metered or unlimited:

- **Metered.** All inbound data transfer is free of charge, and all outbound data transfer is charged based on a predetermined rate. Users are also charged a fixed monthly port fee (based on high-availability dual ports).
- **Unlimited.** All inbound and outbound data transfer is free of charge. Users are charged a single fixed monthly port fee (based on high-availability dual ports).

New ExpressRoute circuits offer two peering options, also known as routing domains: Private or Microsoft Peering. Each circuit can use either one or both peerings. These peerings are shown in [Figure 4-99](#).

- **Azure Private Peering.** Provides connectivity over the Intranet address space into your Azure virtual network. This peering is considered a trusted extension of your core network into Azure.
- **Microsoft Peering.** Provides connectivity over the Internet address space into Microsoft services such as Office 365, Dynamics 365, and Internet-facing endpoints of Azure platform (PaaS) services.

Older circuits may use a third peering model, Azure Public Peering, which provides connectivity to Azure PaaS services only. This is deprecated for new circuits.

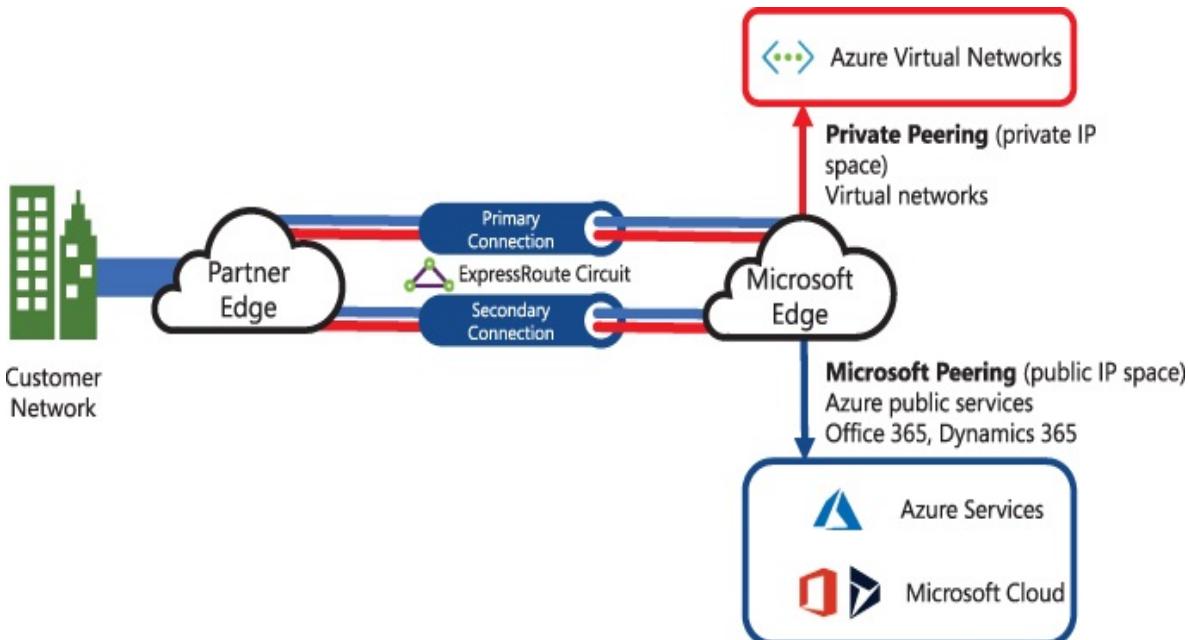


FIGURE 4-99 ExpressRoute peering options

Each ExpressRoute circuit has two connections from your network edge to two Microsoft edge routers, configured using BGP. Microsoft requires dual BGP connections from your edge to each Microsoft edge router. You can choose not to deploy redundant devices or ethernet circuits at your end; however, connectivity providers use redundant devices to ensure that your connections are handed off for high availability to Microsoft in a redundant manner. [Figure 4-100](#) shows a redundant connectivity configuration.

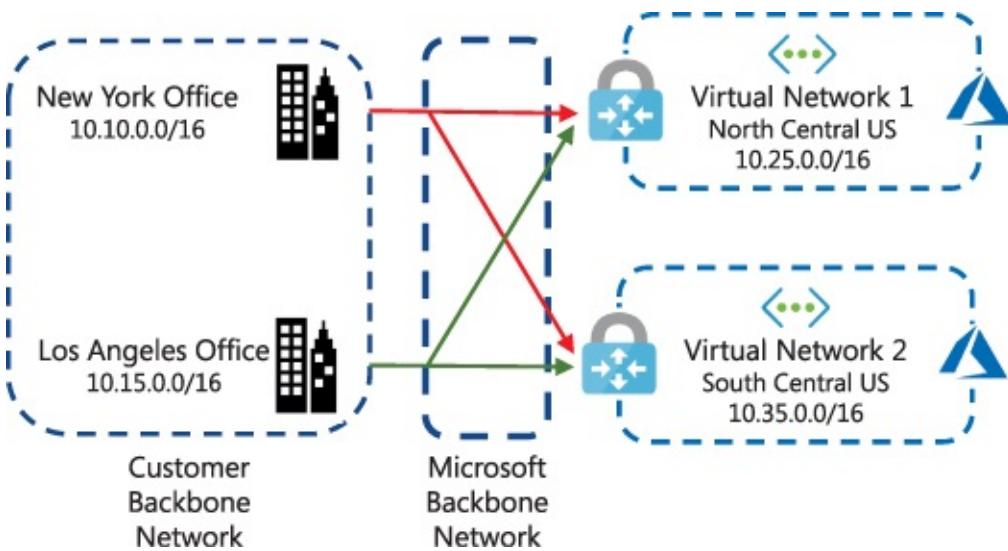


FIGURE 4-100 Multiple cities connected to ExpressRoute in two Azure regions

Global availability and ExpressRoute Premium

ExpressRoute is only available in certain cities throughout the world, so it is important to check with your local providers to determine availability. For a list of ExpressRoute providers and their supported locations, see: <https://docs.microsoft.com/azure/expressroute/expressroute-locations>.

By default, each ExpressRoute circuit enables connectivity to Microsoft data centers within a geopolitical region. For example, a connection in Amsterdam gives you access to all Microsoft data centers in Europe.

With the ExpressRoute Premium add-on, connectivity is extended to all

Microsoft data centers worldwide. This add-on also raises the number of routes permitted for the Azure Private Peering from 4,000 to 10,000. It also increases the number of virtual networks that can be connected to each ExpressRoute circuit, from 10 to between 20 and 100 (depending on the bandwidth of the circuit).

Creating an ExpressRoute circuit

To create an ExpressRoute circuit using the Azure portal, click **+Create A Resource > Networking**, and then choose **ExpressRoute** to open the **Create ExpressRoute** blade (see [Figure 4-101](#)). Select the **Subscription, Resource Group**, and **Resource Location** and specify the **Name** of the ExpressRoute circuit.

Note ExpressRoute locations

When creating an ExpressRoute circuit, you must specify both the peering location and the location of the ExpressRoute circuit resource. There are independent settings, although Microsoft suggests the best practice is for them to be nearby.

On the next screen, on the **Configuration** blade, select the **Port Type, Provider, and Peering Location**, and then specify the **Bandwidth, Billing Model**, and whether the classic operations are required (see [Figure 4-102](#)).

Create ExpressRoute

Basics Configuration Tags Review + create

Use Azure ExpressRoute to create private connections between Azure datacenters and infrastructure on your premises or in a colocation environment. Establish connections to Azure at an ExpressRoute location, such as an Exchange provider facility, or directly connect to Azure from your existing WAN network, such as a multiprotocol label switching (MPLS) VPN, provided by a network service provider.

[Learn more about Express Route circuits](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Resource group * ⓘ

ExamRef-RG

[Create new](#)

Instance details

Region * ⓘ

Canada Central

Name * ⓘ

ExamRef-ER



FIGURE 4-101 Creating an ExpressRoute circuit

Create ExpressRoute

Basics Configuration Tags Review + create

ExpressRoute circuits can connect to Azure through a service provider or directly to Azure at a global peering location.

[Learn more about circuit types](#)

Port type *

Provider

Direct

Create new or import from classic * ⓘ

Create new

Import

Provider * ⓘ

AT&T



Peering location * ⓘ

Washington DC



Bandwidth * ⓘ

200Mbps



SKU * ⓘ

Standard

Premium

Billing model * ⓘ

Metered

Unlimited

Allow classic operations ⓘ

Yes

No

FIGURE 4-102 Creating an ExpressRoute circuit

Note ExpressRoute Billing

Billing for the circuit begins immediately upon resource creation and does not depend upon completing the configuration with the ExpressRoute provider. ExpressRoute circuits can be expensive, so care is advised. It is a good practice to restrict the ability to create ExpressRoute circuits using Azure Policy.

The ExpressRoute circuit will be created. The resource overview blade will show the provider status as Not Provisioned, and also shows the service key. Copy the service key and share it with your ExpressRoute provider. The provider status will change to Provisioning and finally to Provisioned once the provider setup is complete.

Next, you need to provision either Azure Private Peering or Microsoft Peering for your circuit. From the ExpressRoute circuit blade, click **Peerings**, and select the type of peering to configure. Fill in the BGP ASN, and subnets as promoted, and then save the configuration.

For Microsoft Peering, you may see the status Validation Needed for the advertised public IP prefixes. This is because Microsoft needs to validate that you own these IP prefixes before updating their routing to use the ExpressRoute connection. In this case, use the Azure portal to raise a support ticket to perform the validation.

Connecting virtual networks to ExpressRoute

Virtual networks are connected to ExpressRoute circuits using an ExpressRoute gateway. An ExpressRoute gateway is a virtual network gateway, created with the ExpressRoute option (rather than the VPN option, used to create VPN gateways). Just as with VPN gateways, the ExpressRoute gateway must be created in the gateway subnet of the virtual network.

Once the ExpressRoute gateway is created, it can be connected to the ExpressRoute circuit. The process is the same as adding a VPN connection to a VPN gateway, except that the ExpressRoute connection type is selected, and the ExpressRoute circuit specified. The circuit must be enabled by your connectivity

provider and have Azure Private Peering enabled beforehand.

Verify and troubleshoot on-premises connectivity

To verify connectivity or troubleshoot connectivity between on-premises networks and Azure:

- Verify the status and configuration of all VPN connections, virtual network gateways, ExpressRoute connections, or ExpressRoute circuits involved.
- For ExpressRoute, try to reset a failed circuit using the `Get-AzExpressRouteCircuit` and `Set-AzExpressRouteCircuit` PowerShell cmdlets, as described at:
<https://docs.microsoft.com/azure/expressroute/reset-circuit>.
- Try to connect between an on-premises server and an Azure VM, and vice-versa, such as using SSH or TCP.
- Use standard network tools such as `tcping` or `tracert` to confirm connectivity between networks.
- Use the Azure network diagnostics tools described in [Skill 4.5](#).

Configure Azure Virtual WAN

Azure Virtual WAN is a combination of many networking, security, and routing functionalities together to provide a single operational interface for various networking solutions. Azure Virtual WAN facilitates Point-to-site, Site-to-site, ExpressRoute connectivity and Azure Firewall configuration all at one place. The Azure Virtual WAN can be leveraged to either use one of the scenarios or use multiple of them. The Virtual WAN leverages hub-and-spoke topology. The hubs are nothing, but an Azure regions and spokes are considered as individual endpoints. The hubs relate to each other using Microsoft's backbone network wherein for spoke connectivity various VPN devices are used. Microsoft has partnered with many VPN solution providers with ability to export the device info, configure and establish connectivity with Azure Virtual WAN.

Create Azure Virtual WAN using Azure portal

To create Azure Virtual WAN in Azure portal, click **+Create A Resource**, search

for **Virtual WAN**, then click **Virtual WAN**. This will open the **Create WAN** blade, as shown in [Figure 4-103](#). Select the **Subscription**, **Resource Group**, **Location**, and **Type**. Lastly, specify the **WAN Name**.

The screenshot shows the 'Create WAN' blade in the Azure portal. At the top left, there's a breadcrumb navigation: Home > New > Virtual WAN >. The main title is 'Create WAN'. Below it, there are two tabs: 'Basics' (which is selected) and 'Review + create'. A descriptive text states: 'The virtual WAN resource represents a virtual overlay of your Azure network and is a collection of multiple resources.' It includes a 'Learn more' link.

Project details

Subscription *: Visual Studio Ultimate with MSDN

Resource group *: ExamRef-RG

[Create new](#)

Virtual WAN details

Resource group location *: Canada Central

Name *: ExamRef-WAN

Type ⓘ: Standard

FIGURE 4-103 Creating an Azure Virtual WAN

Note Basic vs Standard WAN

With Basic WAN, you can only create Basic Hubs. Basic Hubs are only capable of creating site-to-site connections. For any other connectivity, it is recommended to use Standard WAN.

Once created, Virtual WAN will look like [Figure 4-104](#). Each hub will be represented on the graph once created. You can manage the configuration and connectivity using various options.

Home > VirtualWANDeployment | Overview >

ExamRef-WAN

Virtual WAN

Search (Ctrl+F) Delete Refresh

Overview Resource group : ExamRef-RG Status : Succeeded

Activity log Location : Canada Central Branch-to-branch : Enabled

Access control (IAM) Subscription : Visual Studio Ultimate with MSDN

Tags Subscription ID : 00b72028-9dce-4729-9b2a-a10e92054447

Tags (change) : Click here to add tags

Settings

Configuration Properties Locks Export template

Connectivity

Hubs VPN sites User VPN configurations ExpressRoute circuits Virtual network connections

Monitor

Connection monitor

Support + troubleshooting

Getting started New support request

Each point represents a hub.

There are no hubs. Go to the Getting started page for instructions to connected hubs to VPN sites.

Hub	Hub status	Address Space	Region	VPN sites	Azure Firewall	Point-to-site	ExpressRoute circuits	Virtual network connections
No results								

FIGURE 4-104 Overview page of Azure Virtual WAN

Create a Site-to-Site connection

To establish site-to-site connectivity, follow these steps:

1. Go to **Virtual WAN** created earlier, and under the **Connectivity** options, select **Hubs > + New Hub**.
2. You will need to specify the **Location**, **Name**, and **Private Address Space** for the hub on the **Basics** tab, as shown in [Figure 4-105](#). Note that creating a hub with a gateway will take about 30 minutes.

Create virtual hub

Basics Site to site Point to site ExpressRoute Tags Review + create

A virtual hub is a Microsoft-managed virtual network. The hub contains various service endpoints to enable connectivity from your on-premises network (vpnsite). The hub is the core of your network in a region. There can only be one hub per Azure region. When you create a hub using Azure portal, it creates a virtual hub VNet and a virtual hub vpngateway. [Learn more](#)

Project details

The hub will be created under the same subscription and resource group as the vWAN.

Subscription

Visual Studio Ultimate with MSDN



Resource group

ExamRef-RG



Virtual Hub Details

Region *

Canada Central



Name *

SiteToSite-Hub



Hub private address space * ⓘ

10.1.0.0/16



FIGURE 4-105 Basics blade while creating Virtual Hub for Azure

Virtual WAN

3. On the next screen, you can provide consent to create a site-to-site VPN gateway. Azure will auto populate the autonomous system number (known as the AS Number). You also need to select the Gateway scale units. The gateway scale unit is aggregated throughput of the gateway. If you select **1 Scale Unit—500 Mbps x 2**, that means two instances will be deployed to provide high availability, each with maximum throughput of 500 Mbps. The proper sizing should be done to establish the right scale unit value for your sites. See [Figure 4-106](#).

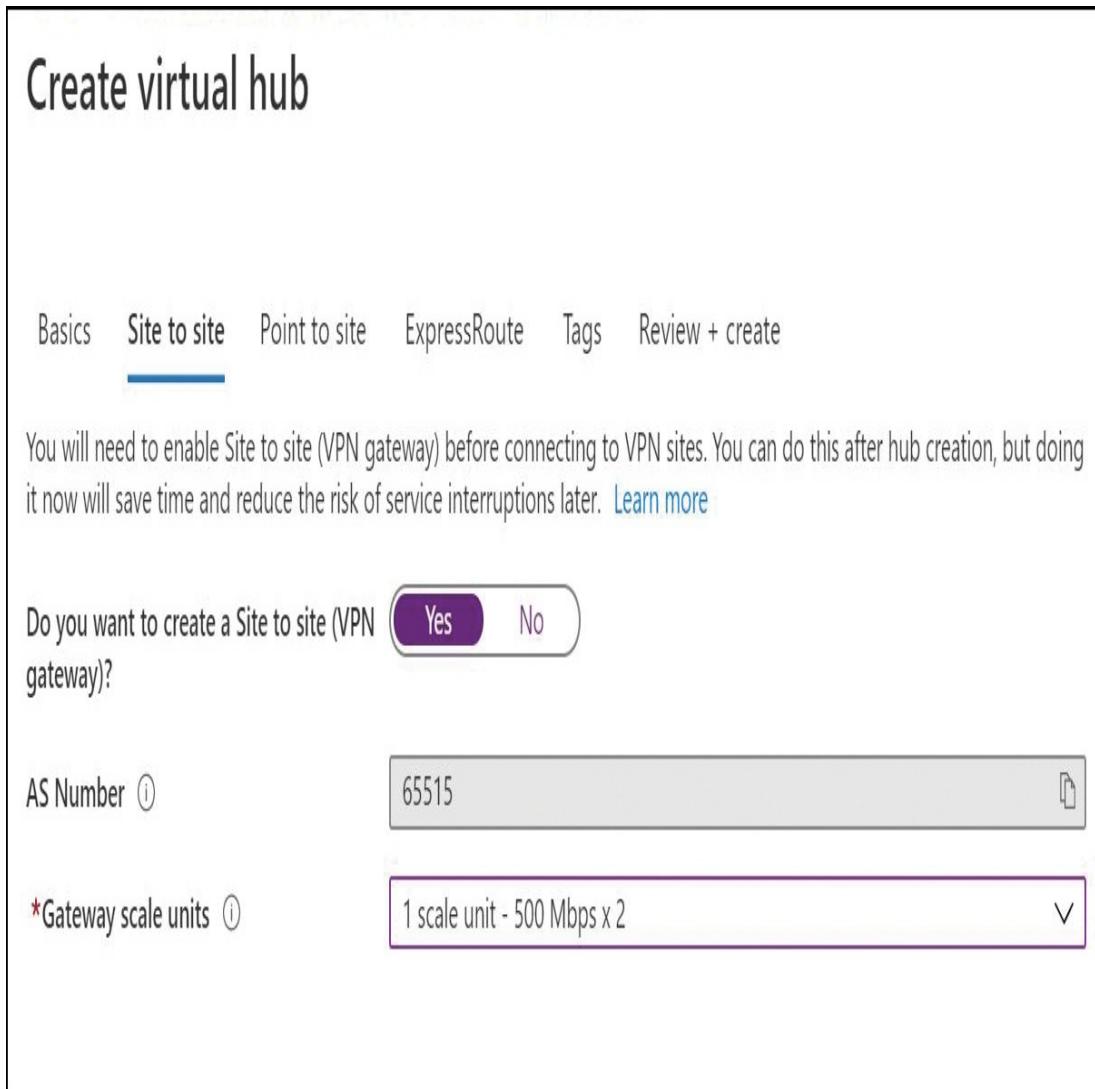


FIGURE 4-106 Site to site blade while creating Virtual Hub for Azure

Virtual WAN

4. Once the hub is created, you can create sites based on your physical locations. These sites will have their own endpoint to communicate to the Hub. The sites can be created from Virtual WAN, under **Connectivity**, **VPN Sites** -> + **Create site**. There are few additional steps listed below which needs to be followed to establish site-to-site connectivity.
5. Connect a VPN site to the hub you created.
6. Connect all your VNets to the hub you created.
7. Download the VPN configuration file and configure your on-premises VPN device.
8. Configure your VPN gateway (optional only for custom BGP IP Address).

Further details can be found at <https://docs.microsoft.com/en-us/azure/virtual-wan/virtual-wan-site-to-site-portal>.

Create a User VPN connection

To establish point-to-site connectivity, follow these steps:

1. Go to the Virtual WAN we created earlier, and under **Connectivity Options**, select **User VPN Configurations** > + **Create User VPN Config**.
2. You will be asked to specify configuration name; select the tunnel type authentication method and supply the certificate name and Base-64-encoded X.509 certificate data.
3. Now, go back to the Virtual WAN you created earlier, and under **Connectivity Options**, select **Hubs** > + **New Hub**.
4. You will be asked to specify the **Location**, **Name**, and **Private Address Space For The Hub** on the **Basics** tab, as previously shown in [Figure 4-105](#). Please note that creating a hub with a gateway will take about 30 minutes.
5. On the **Point To Site** tab, you can provide consent to create a point-to-site user VPN gateway. You need to select the point-to-site configuration that you created earlier. You also need to select the **Gateway Scale Units** and then supply the client address pool CIDR range for the remote users and custom DNS server, as shown in [Figure 4-107](#).

Create virtual hub

Basics Site to site Point to site ExpressRoute Tags Review + create

If you plan to use this hub with Point-to-site connections, you will need to enable Point-to-site gateway before connecting end-user devices. You can do this after hub creation, but doing now will save time and reduce the risk of service interruptions later. [Learn more](#)

Do you want to create a Point to site (User VPN gateway)? Yes No

*Gateway scale units ⓘ

1 scale unit - 500 Mbps x 2, supports 500 clients



Point to site configuration * ⓘ

ExamRefUserConfig



[Create new](#)

or import from file

Client address pool

10.1.0.0/24



i.e. 10.0.0.0/24

Custom DNS Servers

208.67.222.222



At the most 5 custom DNS servers can be provided

FIGURE 4-107 Site to site blade while creating Virtual Hub for Azure Virtual WAN

6. Once the hub is created, you can download the virtual WAN user VPN profile by accessing Virtual WAN, choosing **Connectivity > User VPN Configurations**, selecting the **User VPN config** created earlier, and clicking **Download Virtual WAN User VPN Profile**. This profile can be used to configure individual clients.

Further details can be found at <https://docs.microsoft.com/en-us/azure/virtual-wan/virtual-wan-point-to-site-portal>.

Create an ExpressRoute association

To establish connectivity using ExpressRoute circuit, go to the **Virtual WAN** created in the “Create Azure Virtual WAN using Azure portal” section; in the **Connectivity** options, select **Hubs > + New Hub**. You will be asked to specify a Region, Name, and Hub Private Address Space for the hub on the **Basics** tab, as shown earlier in [Figure 4-105](#). Note that creating a hub with a gateway will take about 30 minutes.

On the **ExpressRoute** tab, you can provide consent to create an ExpressRoute gateway. You also need to select the **Gateway Scale Units**, as shown in [Figure 4-108](#).

Create virtual hub

Basics Site to site Point to site ExpressRoute Tags Review + create

If you plan to use this hub with ExpressRoutes, you will need to enable an ExpressRoute gateway before connecting to ExpressRoute circuits. You can do this after hub creation, but doing it now will save time and reduce the risk of service interruptions later. [Learn more](#)

Do you want to create an ExpressRoute gateway? ⓘ

Yes

No

*Gateway scale units

1 scale unit - 2 Gbps



FIGURE 4-108 ExpressRoute blade while creating Virtual Hub for Azure Virtual WAN

1. Once a hub is created, follow these steps to establish site-to-site connectivity.
2. Create a gateway in the hub you created.
3. Connect all your VNets to the hub you created.
4. Connect your ExpressRoute circuit to the gateway.

Further details can be found at <https://docs.microsoft.com/en-us/azure/virtual-wan/virtual-wan-expressroute-portal>.

Thought experiment

In this thought experiment, apply what you have learned about this objective. You can find answers to these questions in the next section.

Your company, Contoso, wants to lift and shift an existing HR application to Azure. The application architecture comprises two web servers, and a database tier implemented using three servers in a SQL Server Always-On Availability Group. The web application uses an in-memory session state that requires each user to be consistently routed to the same web server instance. The application should be accessible only to the company Intranet, and not exposed to the Internet.

In addition, Contoso has already migrated several other applications to Azure. A recent finance review, however, has highlighted the increasing of Azure spend, and your manager has identified the duplication of infrastructure components (such as domain controller virtual machines) across each migrated application as a potential area where savings can be made. Each of these applications is managed by a separate team, and the team should have administrative access only to their application.

1. How should the web tier be load-balanced?
2. How should the database tier be load-balanced?
3. How can you restrict network traffic between application tiers, and prevent on-premises users from having direct access to the database tier?
4. How should the application be integrated into the company Intranet, avoiding exposing an Internet endpoint?
5. How can you reduce costs by consolidating duplicated components?
6. How does your design maintain administrative separation between applications?

Thought experiment answers

This section contains the solution to the thought experiment for the chapter.

1. The web tier should be load-balanced using Azure Application Gateway. This option is chosen in preference to Azure Load Balancer because it supports cookie-based session affinity, which enables each user to be consistently routed to the same backend server. The App Gateway will be deployed in a separate subnet of the same virtual network that is used to host the web tiers and database tiers of the HR application (each tier uses a

separate subnet). NSGs associated with each subnet are used to restrict network flows, such as to ensure that only the web tier has access to the database.

2. The database tier should be load balanced using Azure Load Balancer. The Load Balancer will be configured with an internal (Intranet) IP address only. Because the Load Balancer is being used as a SQL Server Always-On Availability Group Listener, the **Floating IP (Direct Server Return)** option should be enabled.
3. Network security groups should be used to restrict inbound and outbound traffic for the subnets used by each application tier. Optionally, application security groups can be used to simplify the IP address management and reduce the number of subnets and NSGs required.
4. Connectivity between the application and the on-premises network can be achieved in two ways. The simplest option is to establish a Site-to-Site VPN between the on-premises network and the Azure virtual network. This creates an encrypted tunnel (over the Internet) linking the two networks together. A compatible on-premises VPN device with a static Internet-facing IPv4 address is required, together with a VPN gateway in Azure (hosted in a dedicated gateway subnet). Alternatively, an ExpressRoute connection can be used. This provides a more reliable and consistent connection over a dedicated connection from a connectivity provider. In this case, an ExpressRoute gateway is used to connect the ExpressRoute circuit to the Azure virtual network.
5. A dedicated VNet should be created to contain common services (such as Active Directory servers), which are consumed by multiple applications. Each application should remain in its own VNet, which should only contain application-specific components. The application VNets should be peered with the shared services VNet, in a hub-and-spoke configuration (with the shared services VNet as the hub). This peering will give the applications network access to the shared components.
6. Because each application retains its own VNet containing all application-specific components, there is no loss of isolation or control for the application owners. These application components can even be deployed in separate subscriptions, making separate role-based access and billing straightforward. Peering of Resource Manager VNets is supported across subscription boundaries.

Chapter summary

This chapter covered many of the advanced networking features available in Azure. Below are some of the key takeaways from this chapter.

- Azure virtual networks (VNets) are isolated networks using a private IP address space.
- Virtual networks are divided into subnets, which allow you to isolate workloads.
- Azure reserves the first 4 and last IP address in each subnet. The first IP address allocated to VMs is therefore typically the .4 IP address.
- Private IP addresses for a VM are assigned from a subnet and configured as settings on the IP configuration of a network interface resource.
- A VM can be associated with one or more network interfaces, and each network interface can contain multiple IP configurations.
- Private IP addresses support two allocation methods: dynamic or static. Dynamic IP addresses are released when the VM is stopped (deallocated).
- Public IP addresses are managed as a standalone resource, which can be associated with a network interface IP configuration.
- Public IP addresses support two pricing tiers (SKUs). The Basic tier supports dynamic and static assignment and provides open connectivity (which can be restricted using NSGs). The Standard tier supports zone-redundant deployments, use static allocation only, and is closed by default (access is enabled using NSGs).
- User-defined Routes (UDRs) change the default behavior of subnets allowing you to direct outbound traffic to other locations. Typically, traffic is sent through a virtual appliance such as a firewall.
- If a UDR is used to send traffic to a virtual appliance, IP forwarding must be enabled on the NIC of the virtual appliance VM.
- Routing outbound Internet traffic via a VPN connection to a network security device is known as forced tunneling.
- The effective routes for each network interface can be reviewed to help diagnose routing issues.
- VNets can be connected using either VNet peering or VNet-to-VNet VPN

connections.

- To connect two VNet, they must have non-overlapping IP address spaces.
- Virtual networks can be connected using VNet peering. This is supported both within a region or across regions.
- By default, peered VNets appear and perform as a single network. There is an option to limit connectivity, in which case NSG rules must be used to define the permitted connections.
- VNet peering allows VMs to see each other as one network, but their relationships are non-transitive. If VNetA and VNetB are peered and VNetB and VNetC are peered VNetA and VNetC are not peered.
- A common approach is to use a hub-and-spoke network architecture, in which separate spoke VNets are used by each application, peered to a hub VNet containing a network virtual appliance (NVA). The peering connections must enable Allow Forwarded Traffic.
- Using VNet peering to provide access to a central VNet containing shared services, such as Active Directory domain controllers, is known as service chaining.
- Alternatively, virtual networks can be connected using a VNet-to-VNet VPN connection.
- A virtual network gateway can be used to create VPN connections between virtual networks (and is then called a VPN gateway).
- The size of the VPN gateway should be chosen based on the throughput required.
- The GatewaySubnet is a special subnet that is only used for virtual network gateways.
- A VPN gateway can be shared by peered VNets. The peering connections must enable the settings to Use Remote Gateway (on the peering toward the gateway) and Allow Gateway Transit (on the peering from the gateway).
- Both global VNet peering and VNet-to-VNet VPN connections route traffic between Azure regions over the Microsoft backbone network, not the public Internet.
- Azure DNS provides an authoritative DNS service for hosting Internet-

facing domains.

- DNS zones in Azure DNS must be delegated from the parent domain. This is achieved by setting up appropriate NS records in the parent domain, pointing to the name servers assigned by Azure DNS.
- DNS records in Azure DNS are managed using record sets, which are the collection of records with the same name and the same type.
- DNS records at the zone apex use the record name `@`. You cannot create records with the CNAME record type at the zone apex.
- Azure DNS Alias records allow DNS records to reference other Azure resources, such as a public IP address.
- DNS zone files are a standard format used to transfer DNS records between DNS systems. DNS zone files can only be imported into or exported from Azure DNS by using the Azure CLI.
- Azure-provided DNS, also known as Internal DNS, provides VM-to-VM DNS lookups within a virtual network.
- Alternatively, a customer can implement their own DNS servers, which can be configured either at the VNet or the network interface level.
- Azure DNS also supports private DNS zones, which can also be used to enable VM-to-VM DNS lookups.
- Network security groups are used to create firewall rules to control network flows.
- NSGs can be applied at the subnet level, or on individual VM network interfaces.
- Each NSG includes a list of default rules, which can be overridden using user-defined rules. Rules are applied in priority order (processing stops at the first rule matching the traffic in question).
- Source and destination IP address ranges in NSG rules can be specified explicitly using CIDR ranges.
- IP address ranges can also be specified using service tags which are platform shortcuts for the IP ranges for key Azure services. Commonly used service tags include VirtualNetwork, Internet, Azure Cloud, Storage, and SQL.
- IP address ranges can also be specified using application security groups

(ASGs). ASGs allow NSG rules to be defined for groups of VMs without needing to allocate the VMs into separate subnets.

- Tools to help identifying the required NSG rules include service map and NSG flow logs.
- Effective security rules can be reviewed for each network interface. This allows you to see the exact IP ranges used by each service tag and ASG.
- Azure Firewall is a managed service which provides out of box network security to secure Azure resources.
- Azure Firewall allows us to create and configure application and network rules. Application rules are created with the list of fully qualified names which allowed to be accessed from a subnet. While network rules are combination of source and destination IP addresses along with their ports and protocols.
- The Azure Bastion service is provisioned within a Virtual Network within a separate subnet called `AzureBastionSubnet`. If you have multiple VNets in your environment, you will need to deploy Azure Bastion for each VNet separately.
- Azure Application Gateway is a type of Load Balancer which can manage traffic for web applications. The web traffic routing occurs at application layer (OSI layer 7).
- The Azure Application Gateway routes application web traffic to defined resources in a backend pool.
- Azure Load Balancer (ALB) is a fully managed, high performance load-balancing service for TCP and UDP traffic. It operates at the transport layer (OSI Layer 4). Unlike App Gateway, it does not have visibility into application-level traffic.
- ALB can be deployed with either a public (Internet) or private (Intranet) frontend IP address.
- ALB comes in two pricing tiers (SKUs): Basic or Standard. The Standard tier supports availability zones, larger and more flexible backend pools, and a number of other features. The Basic tier is free of charge.
- An ALB load-balancing configuration comprises frontend IP configuration, backend pool, health probes, and load-balancing rule.

- ALB also supports port forwarding, using inbound NAT rules. This maps a specific frontend port to a specific backend port on a specific backend server.
- Network Performance Monitor provides monitoring for hybrid networks. It supports performance monitor (for monitoring connections between two endpoints), connectivity monitor (to monitor outbound connections to a given IP or FQDN), and ExpressRoute monitor to monitor ExpressRoute connections.
- Network Watcher is a central hub providing access to a wide range of networking tools in Azure.
- IP Flow Verify is a Network Watcher feature used to test if a given network flow is allowed in or out of an Azure VM.
- Next Hop is used to determine the next hop address and routing rule for a given network flow.
- Packet Captures enables network traffic on a given VM to be captured, either locally or to an Azure storage account.
- Network Topology creates a diagrammatic representation of the resources in your virtual network.
- VPN Troubleshoot provides automated, in-depth troubleshooting of VPN connections.
- Connection Troubleshoot allows you to test the connectivity between two Azure VMs, or between a VM and an arbitrary external endpoint.
- Connection Monitor enables long-term connection monitoring, using similar diagnostics as used by Connection Troubleshoot.
- Site-to-Site VPN connections provide connectivity between an on-premises network and an Azure virtual network, using an encrypted tunnel over the public Internet.
- VPN gateways are virtual network gateways deployed with gateway type VPN. They are used to terminate site-to-site VPN connections.
- Site-to-Site VPNs support BGP routing and active-active gateways and connections to enable high availability.
- A wide variety of physical (and software) devices are supported as the on-premises Site-to-Site VPN endpoint. The device must have an Internet-

facing static IPv4 address.

- A local network connection is an Azure resource used to represent the on-premises VPN device and network in Azure.
- An ExpressRoute connection provides connectivity between an on-premises network and an Azure virtual network, using a dedicated connection from a connectivity provider.
- You can connect to ExpressRoute either via your co-location facility provider, via a point-to-point ethernet connection, or by extending your IPVPN WAN.
- ExpressRoute provides Microsoft Peering (connectivity to Azure PaaS endpoints, and other Microsoft services) or Private Peering (connectivity to Azure virtual networks). The former uses Internet address, and the latter uses Intranet addresses. Azure Public Peering, for Azure PaaS services only, is deprecated for new ExpressRoute circuits.
- ExpressRoute circuits provide different levels of bandwidth, from 50Mbps to 10Gbps. They also provide redundant connections.
- ExpressRoute circuits are connected to an Azure virtual network using an ExpressRoute gateway (a virtual network gateway of type ExpressRoute).
- By default, ExpressRoute provides connectivity to all Microsoft data centers in a given geopolitical region. The ExpressRoute Premium Add-On extends coverage to all data centers, globally. It also increases the number of private peering routes and the number of virtual networks, which can be connected to a circuit.

Chapter 5

Monitor and back up Azure resources

As you begin to deploy services into your Azure subscriptions, how the environment will be monitored is one of the first questions you will need to answer and to answer it you must think about all the services in your deployment. You will most likely have several services deployed, including Infrastructure-as-a-Service services (for example, virtual machines), which include compute, storage, and networking. And even without services deployed today, over time, you might have Platform-as-a-Service services for hosting applications. You will also be using the services that drive your virtual machines in more meaningful ways, such as implementing advanced configurations in Azure Storage and Azure Identity.

You will need to account for all these services—along with the Azure platform itself—in your monitoring strategy. This includes all your infrastructure, applications, and networking.

By developing a proactive monitoring strategy, you will be able to understand the operation of your environment at a component level, including resource health and resource spend. Implementing a robust strategy will help you increase your uptime through proactive notifications, so you can resolve issues before they become problems and optimize your resources for optimal performance, which allows you to increase your ROI with the services you deploy.

As you develop your strategy, there are three areas you should consider:

- **Visibility into services and the Azure Platform.** This is all about understanding how an application or set of services is performing across the board. You will need to understand what metrics you need to monitor and how those can be acted on in Azure through both alerts and visualizations in dashboards.
- **Deeper insights into applications.** This is particularly true with service or dependency maps and advanced tracing. You may even use these insights

to drive automation and remediations within your environments.

- **Resource optimization.** You need to understand which metrics are important to not just the health of your application, but also the effects on users or systems that consume those application. By using the visibility and insights you extract from the Azure platform, you can directly correlate the effects of remediations in your environment.

Azure includes multiple services that perform specific roles for monitoring and optimization. It is critical that you understand both the out-of-the-box monitoring capabilities of Azure and the scenario-specific monitoring capabilities within the platform. This section will focus on out-of-the-box monitoring and optimization through both Azure Monitor, as well as scenario-specific monitoring with Azure Monitor logs and log data that is stored in Log Analytics).

Azure Backup is another critical service that enables simplified disaster recovery for virtual machines by ensuring that data is securely backed up and easily restorable. In this chapter, we'll also review how to implement and manage Azure backup and recovery solutions with an emphasis on the Azure Backup Service and Azure Site Recovery.

Skills covered in this chapter:

- [Skill 5.1: Monitor resources by using Azure Monitor](#)
- [Skill 5.2: Implement backup and recovery](#)

Skill 5.1: Monitor resources by using Azure Monitor

Azure Monitor maximizes the availability and performance of your applications by delivering a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications are performing and proactively identifies issues affecting them and the resources on which they depend. The Azure Monitor landing page provides a jumping off point to configure other more specific monitoring services, such as Application Insights, Network Watcher, Log Analytics, Management Solutions, and so on. [Figure 5-1](#) shows some of the various data sources and how they are collected, either as metric or log data. The data is consumed, visualized, or acted on by various services in Azure.

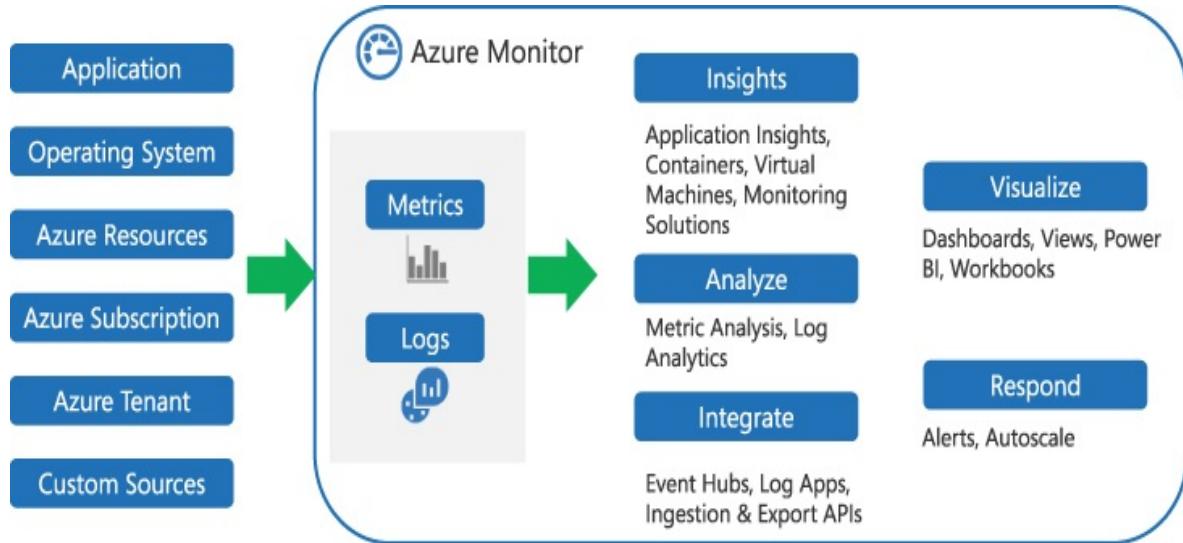


FIGURE 5-1 Azure Monitor data sources for metric and log data and the ways you can act on the data

More Info Azure Monitor

To learn more about the capabilities of Azure Monitor see <https://docs.microsoft.com/azure/monitoring-and-diagnostics/monitoring-overview-azure-monitor>.

More Info Azure Monitor for Containers

Azure Monitor for Containers is an offering that provides new capabilities for monitoring your Managed Kubernetes clusters (AKS) and Azure Container Instances (ACI). You can learn more about the new capabilities at <https://docs.microsoft.com/en-us/azure/azure-monitor/insights/container-insights-overview>.

More Info Azure Monitor for VMs

Azure Monitor for VMs is an offering that provides new capabilities for monitoring your virtual machines and virtual machine scale sets. You can learn more about the new capabilities at <https://docs.microsoft.com/azure/azure-monitor/insights/vminsights>

overview.

Azure Monitor helps you track performance, maintain security, and identify trends by ingesting metrics and telemetry from multiple areas, including applications and the operating systems of virtual machines. It also allows you to query your Azure resources (which emit performance counters), your Azure subscriptions, Azure AD tenant, and event custom sources.

The data from your Azure resources is ingested into either metrics stored within the Azure platform and accessible by the monitor service or as logs into Log Analytics.

Important Log Analytics

Log Analytics must be enabled and configured before insights can be extracted or visualizations can be created that are dependent on that data.

Comparing metrics and logs surfaces some key differentiators:

- **Retention.** Most of the metrics are retained for 93 days within the Azure service, while logs stored in Log Analytics can be retained for up to 2 years. There are opportunities to do long term retention of metrics by storing metrics in Log Analytics as well.
- **Properties.** Metrics have a fixed set of properties (or attributes). These are time, type, resource, value, and dimensions (optional). Logs have different properties for each log type and even support rich data types, such as date and time.
- **Data availability.** Metrics are gathered over time (like once a minute) and available for immediate query. Logs are often gathered after being triggered by an event (such as an event is written to an application log) and can take time to process before they are available for query. While both offer near real-time query capabilities, metrics will typically be used for fast alerts, and logs are used for more complex analysis.

Once the data is collected, Azure Monitor provides a single pane of glass, or entry point, to interacting with your metrics and logs. Interactions can include querying and alerting, building visualizations and dashboards, or even automated responses based on telemetry for functionality, such as autoscaling in

virtual machines.

Data stored in Log Analytics can also be queried directly through a Log Analytics Workspace, where you will have access to the same query interfaces as you have through Azure Monitor, but you also can make customizations to the configuration of the workspace and access workspace-specific solutions, including visualizations and queries.

All the data that you can access through Azure Monitor can be used to create alerts within Azure Monitor with alert rules. Alert Rules are built based on target resources or resource types, such as virtual machines, storage account, and even PaaS services and your custom conditions. Alerts allow you to be proactively notified of the health of the resources you deploy in Azure, and you are not limited to notifications because alert rules leverage actions groups that allow you to even implement automation based on an alert condition.

This section covers how to:

- [Configure and interpret metrics](#)
- [Configure Log Analytics](#)
- [Query and analyze logs](#)
- [Set up alerts and actions](#)
- [Configure Application Insights](#)

Configure and interpret metrics

Recall that metrics are the numerical values that are output by resources and services within Azure. Metrics are available for a number of Azure resources, but not all resources support metrics at this time.

Metrics includes platform metrics, which are created by Azure resources and made available in Azure Monitor for querying and alerting. You can also query application metrics from Application Insights if the service is enabled and you have instrumented your applications—regardless of whether that application is hosted on a virtual machine or even in a PaaS service, such as Azure App Service. Virtual machines in Azure can also push custom metrics to the monitor service using the Windows Diagnostic extension on Windows servers and with

the InfluxData Telegraf Agent on Linux VMs. There is also an opportunity to push custom metrics from other sources through a REST API.

[Figure 5-2](#) shows an example of a metrics chart displaying the percentage of CPU usage for a virtual machine.

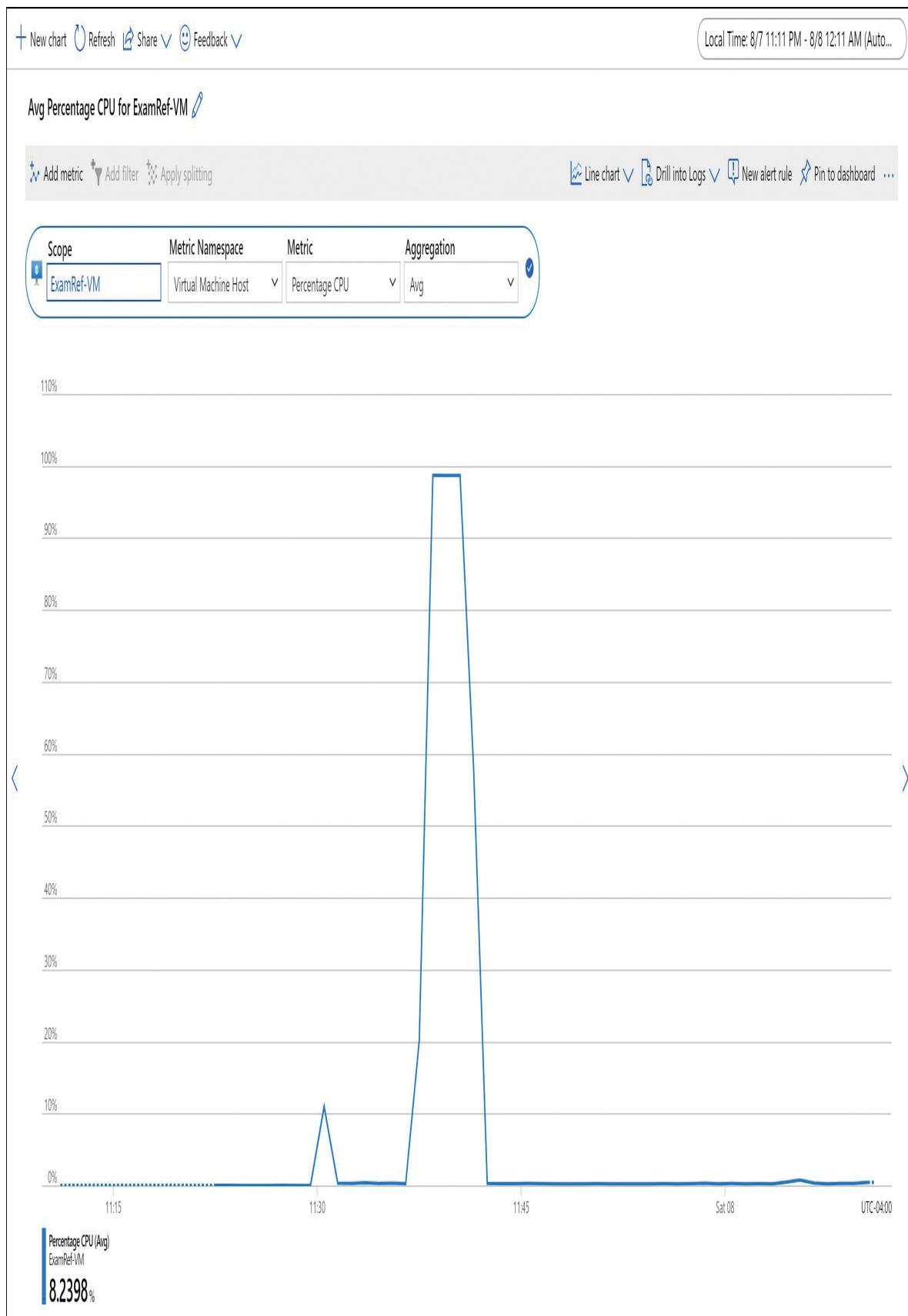


FIGURE 5-2 Azure Metrics

Important Numerical values in Azure

In this case, we are only referring to the numerical values that the resources in Azure generates, not the logs or text-based values, such as the value of an event log that can be stored in a storage account or in a Log Analytics Workspace.

Azure metrics are collected at one-minute intervals (unless otherwise specified) and are identified by a metric name and a namespace (or category). Most of the Azure metrics are retained for 93 days within Azure Monitor, but there are notable exceptions listed below:

Guest OS metrics

- Classic guest OS metrics
 - Collected through diagnostic extensions and sent to an Azure storage account.
 - Retention period of 14 days.
- Guest OS metrics sent to Azure Monitor metrics
 - Monitored by Windows diagnostic extensions or the InfluxData Telegraf agent and are routed to an Azure Monitor data sink.
 - Retention period of 93 days.
- Guest OS metrics collected by Log Analytics agent
 - Collected by the Log Analytics agent and are sent to a Log Analytics workspace.
 - Retention period of 31 days. This retention period can be extended for up to two years.
- Application insights log-based metrics
 - Log-based metrics those are translated into log queries.
 - Retention period of 90 days.



Exam Tip

For longer-term retention, metrics can optionally be sent to Azure Storage for select resources and retained up to the configured retention policy or the storage limits of the account. They can also be sent to Log Analytics with a default retention period of 31 days.

As metrics are collected, each metric has the following properties:

- The time the value was collected
- The type of measurement the value represents
- The resource with which the value is associated
- The value itself

Metrics can be one dimensional or multidimensional with up to 10 dimensions. A nondimensional metric can be thought of as the metric name, and the value of the metric output is collected by the Monitor service over time. A multidimensional metric (both from an Azure resource or a custom metric) is the metric name and an additional name-value pair with additional data. For example, imagine a storage account with multiple Blob containers where you need to track the consumption of storage by container. A nondimensional metric would provide only the total consumed storage for the Blob service in the storage account where a multidimensional metric would provide the consumption by container as it has the additional data stored in the metric record.

To interact with metrics, open the **Metrics** blade in Azure portal and open the Azure Monitor. You will be presented with a blank chart (see [Figure 5-3](#)). You can select the scope and required metrics to customize the metrics chart as needed.

Home >

Monitor | Metrics

Microsoft

Search (Ctrl+ /)

- [Overview](#)
- [Activity log](#)
- [Alerts](#)
- [Metrics](#)
- [Logs](#)
- [Service Health](#)
- [Workbooks](#)

Insights

- [Applications](#)
- [Virtual Machines](#)
- [Storage accounts](#)
- [Containers](#)
- [Networks \(preview\)](#)
- [Azure Cosmos DB](#)
- [Azure Cache for Redis \(preview\)](#)
- [Key Vaults \(preview\)](#)
- [More](#)

Settings

- [Diagnostics settings](#)
- [Data Collection Rules](#)
- [Autoscale](#)
- [Private Link Scopes](#)

[Support + Troubleshooting](#)

Select a scope

Browse Recent

Resource types	Locations
All resource types	All locations

Search to filter items...

Scope	Resource type	Location
<input type="checkbox"/> Visual Studio Ultimate with MSDN	Subscription	-
<input type="checkbox"/> ExamRef-RG	Resource group	-
<input type="checkbox"/> NetworkWatcherRG	Resource group	-

Selected scopes

No scopes selected

No results

Apply Cancel Clear all selections

FIGURE 5-3 Azure Metrics blank chart

Analyze metrics across subscriptions

To begin populating the chart, you need to select a metric. To select a metric, you must select a **Subscription** and a **Resource Group**. Optionally, you can filter by **Resource Type** as well. Selecting a resource will then allow you to select a metric namespace (or category), a metric, and an aggregation if applicable. For example, to view the Ingress metric for a storage account, select the storage account from the **Scope** drop-down menu, choose **Account** from the **Metric Namespace** drop-down menu, choose **Ingress** from the **Metric** drop-down menu, and choose **Sum** from the **Aggregation** drop-down menu, as shown in [Figure 5-4](#).

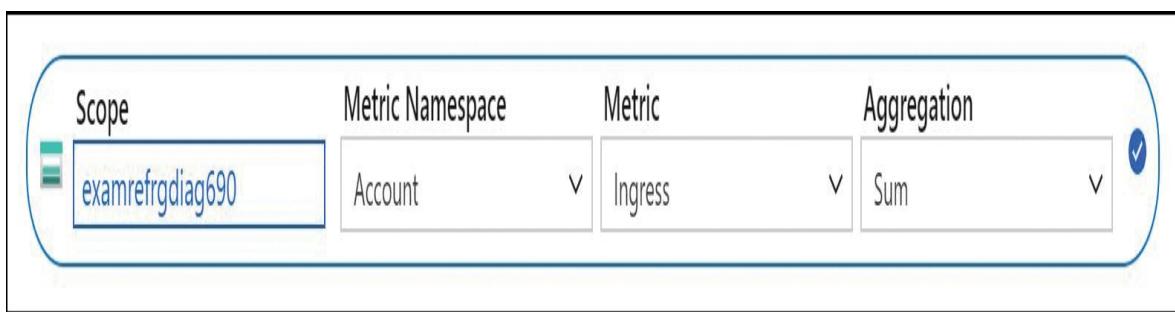


FIGURE 5-4 Azure Metrics selection

You can add multiple metrics to the chart, and you can even mix **Resources**, **Namespaces**, **Metrics**, and **Aggregations** as required (see [Figure 5-5](#)).



FIGURE 5-5 Azure Metrics selection for multiple resources

The chart will be rendered as you complete each resource selection. The period for the query can be changed up to the retention limits of the metrics

service, and the chart can be rendered as a **Line Chart** (default), **Area Chart**, **Bar Chart**, **Scatter Chart**, or **Grid**. An example of a **Line Chart** is shown in Figure 5-6.

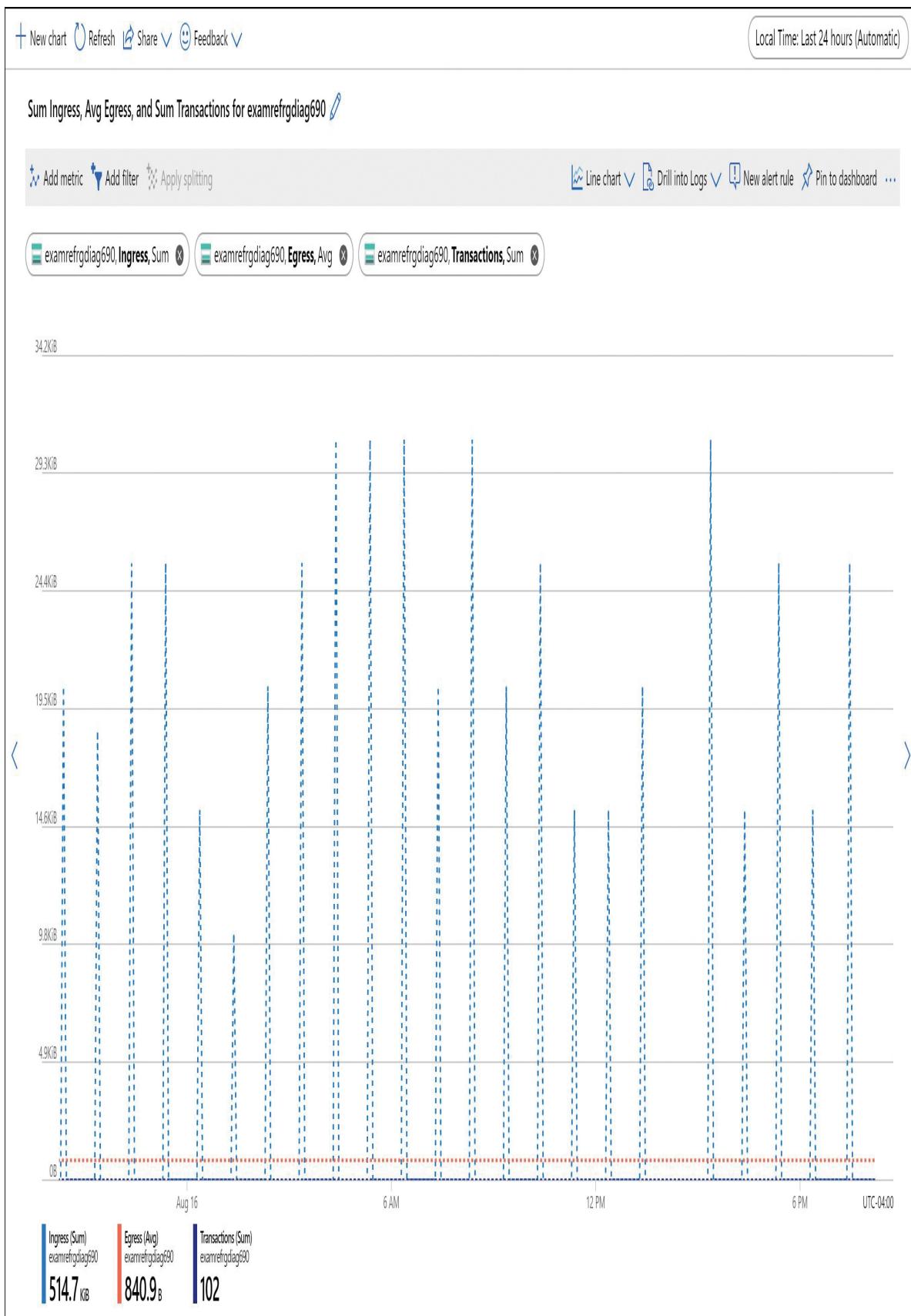


FIGURE 5-6 Azure Metrics line chart

Note that you are not limited to charting resources from the same subscription. You can select metrics for resources of any available type across all the subscriptions to which you have access.

From the **Metrics** blade, you can also create a new alert rule based on the metric query that is visualized. If you need to perform a deeper analysis, the raw metric data can also be exported to Excel.

Note Azure dashboards

Each chart or visualization that you create in Azure Monitor can also be pinned to an Azure dashboard. You can have multiple dashboards in Azure, and you can even share a dashboard with others in your organization.

You also are not limited to creating a single chart. Selecting the **Add Chart** button in the Metrics Explorer will allow you to stack multiple charts, so existing charts can be cloned and then customized.

Note Metrics and visual response times

If you are evaluating a web application, you might want to use multiple charts for visualization response times (in milliseconds) and response size (in kilobytes). This is especially useful when you are working with metrics that have different units of measure or when the scale of the metrics you are evaluating varies widely.

Configure Azure Monitor logs

Log Analytics helps you collect, correlate, search, and act on log and performance data generated by operating systems, applications, and Azure services. It gives you operational insights using rich search and visualizations. Log Analytics provides a single pane of glass for interacting with the data from the entire platform and the workloads you host on it including both Linux and Windows servers. Also, Log Analytics can be used with other cloud providers.

A Log Analytics Workspace is where logs are collected and aggregated. The logs can also be queried and visualized through Log Analytics or through Azure Monitor. A workspace is an Azure resource, meaning that RBAC can be applied for granular access to the service and the data stored within it. This also means that workspaces can be in regions that meet your organization's regulatory requirements, data isolation, and scope. You can create multiple workspaces in a single subscription.

Implement Log Analytics Workspace

A workspace can be created through the Azure portal, Azure PowerShell, the Azure CLI, and Resource Manager templates. To create a workspace through the Azure portal, browse to the Azure Marketplace and search for **Log Analytics Workspace**. Select **Create** to open the **Log Analytics Workspace** configuration blade.

To configure a workspace, you will need to provide (see [Figure 5-7](#)):

- A name for the workspace
- The subscription the workspace will be associated with
- A resource group
- A location
- A selection for pricing tier (see [Figure 5-8](#))

Create Log Analytics workspace

Basics Pricing tier Tags Review + Create

i A Log Analytics workspace is the basic management unit of Azure Monitor Logs. There are specific considerations you should take when creating a new Log Analytics workspace. [Learn more](#) X

With Azure Monitor Logs you can easily store, retain, and query data collected from your monitored resources in Azure and other environments for valuable insights. A Log Analytics workspace is the logical storage unit where your log data is collected and stored.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * i

Visual Studio Ultimate with MSDN

Resource group * i

ExamRef-RG

[Create new](#)

Instance details

Name * i

ExamRef-LAWorkspace

Region * i

Canada Central

[Review + Create](#)

[« Previous](#)

[Next : Pricing tier >](#)

FIGURE 5-7 Log Analytics Workspace configuration

The screenshot shows the 'Create Log Analytics workspace' page. At the top, there is a breadcrumb navigation: 'Home > New > Log Analytics Workspace >'. Below it is the main title 'Create Log Analytics workspace'. A horizontal navigation bar at the top of the form includes tabs: 'Basics' (selected), 'Pricing tier' (underlined), 'Tags', and 'Review + Create'. The 'Pricing tier' section contains the following text: 'The cost of your workspace depends on the pricing tier and what solutions you use.' followed by 'To learn more about Log Analytics pricing [click here](#)'. Below this, under the heading 'Pricing tier', it says 'You can change to a Capacity Reservation tier after your workspace is created. [Learn more](#)' and 'To learn more about access to legacy pricing tiers [click here](#)'. A dropdown menu for 'Pricing tier*' is open, showing 'Pay-as-you-go (Per GB 2018)' as the selected option.

FIGURE 5-8 Pricing tier for Log Analytics Workspace

Note Log Analytics pricing

Details on pricing for Log Analytics can be found at

<https://azure.microsoft.com/pricing/details/monitor/>. This page also includes the pricing details for other services related to Azure Monitor, such as Application Insights and Alert Rules.

Note that Log Analytics is not available in all regions. To select an appropriate region, you can use the Azure Products by Region documentation at <https://azure.microsoft.com/global-infrastructure/services/>.

To select the appropriate pricing tier, review the pricing documentation at <https://azure.microsoft.com/pricing/details/monitor/>. A new workspace will default to the Free tier, which includes 5 GB of log storage per month (31 days) with per-GB pricing and per-GB charges for additional storage and retention.

You can create a Log Analytics Workspace with Azure PowerShell or the Azure CLI, you can also use a Resource Manager template.

After a workspace has been provisioned, you must enable data collection and configure both resource and tenant logs to store their logs within the service.

To collect event and performance data from Windows and Linux machines, open the workspace and configure the Agent Management (see [Figure 5-9](#)). From this blade, you can obtain the Workspace ID, Primary Key, and Secondary Key for associating machines with the service through the monitoring agent. You can use this information when manually onboarding clients to the workspace.

Home >

ExamRef-LAWorkspace | Agents management



Log Analytics workspace

Search (Ctrl+ /)

« Windows servers

Linux servers

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Locks

Export template

Agents management

Advanced settings

General

Quick Start

Workspace summary

View Designer

0 Windows computers connected

[Go to logs](#)

Download agent

Download an agent for your operating system, then install and configure it using the keys for your workspace ID.

You'll need the Workspace ID and Key to install the agent.

[Download Windows Agent \(64 bit\)](#)

[Download Windows Agent \(32 bit\)](#)

Workspace ID

1de7a7fe-5453-43e0-a230-247231b45599



Primary key

1l3nMr1jxncMSe+A6S9ovjsFNkdesSemfuX+l4A09puLXMaNhjFDM...



[Regenerate](#)

Secondary key

8Yzg9NhPpiAP7WBZGLyVYr6fddCzyjN0ecyjvBdCkGRdcudDhzpWs...



[Regenerate](#)

Log Analytics Gateway

If you have machines with no internet connectivity to Log Analytics workspace, download the Log Analytics Gateway to act as a proxy.

[Learn more about Log Analytics Gateway](#)

[Download Log Analytics Gateway](#)

FIGURE 5-9 Log Analytics Workspace Agent Management

By choosing **Advanced Settings -> Data Settings**, you can configure the Windows Event Logs, Windows Performance Counters, Linux Performance Counters, Syslog, IIS Logs, and Custom Fields and Custom Logs (see [Figure 5-10](#)).

Advanced settings

examref-laworkspace

⟳ Refresh 📈 Logs 🖥 Save ✎ Discard

Connected Sources	Windows Event Logs	Collect the following performance counters <small>?</small>
∅ Data	Windows Performance Counters	<input type="text" value="Enter the name of a performance counter to monitor"/> +
Computer Groups	Linux Performance Counters	
IIS Logs		
Custom Fields		
Custom Logs		
Syslog		

COUNTER NAME **SAMPLE INTERVAL**

LogicalDisk(*)% Free Space	10 seconds	Remove
LogicalDisk(*)Avg. Disk sec/Read	10 seconds	Remove
LogicalDisk(*)Avg. Disk sec/Write	10 seconds	Remove
LogicalDisk(*)Current Disk Queue Length	10 seconds	Remove
LogicalDisk(*)Disk Reads/sec	10 seconds	Remove
LogicalDisk(*)Disk Transfers/sec	10 seconds	Remove
LogicalDisk(*)Disk Writes/sec	10 seconds	Remove
LogicalDisk(*)Free Megabytes	10 seconds	Remove
Memory(*)% Committed Bytes In Use	10 seconds	Remove
Memory(*)Available MBytes	10 seconds	Remove
Network Adapter(*)Bytes Received/sec	10 seconds	Remove
Network Adapter(*)Bytes Sent/sec	10 seconds	Remove
Network Interface(*)Bytes Total/sec	10 seconds	Remove
Processor(_Total)% Processor Time	10 seconds	Remove
System(*)Processor Queue Length	10 seconds	Remove

FIGURE 5-10 Log Analytics Workspace advanced settings for data

After the workspace has been configured, you can begin to onboard machines. For machines to report telemetry to Log Analytics, they must be running the Azure Log Analytics (OMS) agent. This agent was previously referred to as the Microsoft Monitoring Agent (MMA) or the OMS Linux agent. The agent binds to a workspace to collect the data defined in the workspace settings or in installed solutions.

The method for installing the agent varies based on the machine operating system, where it is hosted, and how it is managed.

- Azure Virtual Machines can be onboarded manually through the Azure portal; automatically through a Log Analytics Workspace associated with Azure Security Center; programmatically through the Log Analytics VM extension for Windows or using Azure PowerShell; the Azure CLI; or a Resource Manager template for Linux.
- Hybrid Windows computers (server or client) can be onboarded manually by downloading the agent and installing it locally or through Azure Automation DSC when using hybrid workers.
- Hybrid Linux computers (server only) can be onboarded manually by downloading the agent and installing it locally.
- Machines managed with System Center Operations Manager (SCOM) can be integrated directly with Log Analytics by configuring SCOM to forward logs to the service.

Note Agent deployment and installation

For specific guidance on how you should deploy and install the agent based on your scenario, refer to the documentation at

<https://docs.microsoft.com/azure/azure-monitor/platform/log-analytics-agent#install-and-configure-agent>.

For the agent to send telemetry, you must also ensure that the required ports are available, and the required URIs are added to the approved-list. The agent utilizes port 443 for all outbound communication. The required URIs are shown in [Table 5-1](#).

TABLE 5-1 Log Analytics Agent ports and protocols

Agent Resource	Ports	Direction	Bypass HTTPS inspection
*.ods.opinsights.azure.com	Port 443	Outbound	Yes
*.oms.opinsights.azure.com	Port 443	Outbound	Yes
*.Blob.core.windows.net	Port 443	Outbound	Yes
*.azure-automation.net	Port 443	Outbound	Yes

Configure diagnostic settings

While the resources you deploy in Azure create metrics automatically, many of them also offer richer diagnostics logs, which can be configured to send their log data to another location, such as a storage account or a Log Analytics Workspace. In addition to resource logs, there are also tenant-level services, such as Azure Active Directory, which exist outside a subscription from which you might need to collect log data.

Diagnostics logs are one type of log data. There is also log data within the Azure Activity Log, and there is log data that can be obtained from virtual machines with the use of diagnostics agents that is separate from diagnostic logs associated with a tenant-level service or an Azure resource. It is important to understand the differences between the types of log data that are available and where that log data can be stored.

Important Resource and tenant logs are diagnostic logs

Both resource logs and tenant logs are considered diagnostics logs. Diagnostics logs that you configure for a tenant service or a resource are separate from the Azure Activity Log and guest telemetry obtained with diagnostics agents.

The Azure Activity Log surfaces data at the subscription level and can be useful for understanding actions that occur within your environment against the Resource Manager APIs. For example, when a new deployment is submitted, the events associated with that deployment such as the time it was submitted, the

resources that were created, and the user that submitted the request are all tracked within the Activity Log. However, at the subscription level, you are missing any resource-level logs. For example, the Activity Log can show when a Network Security Group (or NSG) was created, but it cannot show when an NSG rule was applied to traffic that was subject to the NSG, such as when a port or protocol is blocked. Diagnostic logs provide this functionality.

Note Retained for 90 days

Events in the Activity Log are retained for 90 days. You can retain the data for a longer period by sending the logs to Azure Storage and/or a Log Analytics Workspace.

Diagnostic logs will need to be enabled for each resource from which you want to collect additional telemetry. Note that metrics are resource-specific and captured automatically, so you only need to enable diagnostic logs to capture log data or to send metrics to another service.

Important Support for Diagnostic Logs

Not all Azure resource types support diagnostic logs. A full list of services that support logs and their service-specific log schemas can be found at <https://docs.microsoft.com/azure/monitoring-and-diagnostics/monitoring-diagnostic-logs-schema>.

To enable diagnostic logs through the Azure portal, you can browse to the resource itself to create the settings. The alternative and recommended method is to browse to the **Azure Monitor** and **Diagnostic Settings** blade. From this blade, you can view all the resource types eligible for diagnostic logs and view the status (**Enabled** or **Disabled**) for log collection on each resource. Also, you can filter by **Subscription**, **Resource Group**, **Resource Type**, and **Resource**. An example is shown in [Figure 5-11](#).

Home >

Monitor | Diagnostics settings

Microsoft

Search (Ctrl+ /) Refresh Provide feedback

Alerts Metrics Logs Service Health Workbooks Insights Applications Virtual Machines Storage accounts Containers Networks (preview) Azure Cosmos DB Azure Cache for Redis (preview) Key Vaults (preview) ... More

Subscription * Resource group Resource type Resource

Visual Studio Ultimate with MSDN 0 selected

Select any of the resources to view diagnostic settings.

Name	Resource type	Resource group	Diagnostics status
examref-vm805	Network interface	ExamRef-RG	Enabled
ExamRef-VM-nsg	Network security group	ExamRef-RG	Disabled
ExamRef-VM-ip	Public IP address	ExamRef-RG	Disabled

Diagnostics settings Data Collection Rules Autoscale Private Link Scopes

Support + Troubleshooting

Usage and estimated costs Advisor recommendations

Timeline.xlsx - Excel

FIGURE 5-11 Azure Monitor Diagnostic settings

To enable diagnostic settings, click a resource with a status of **Disabled**. In the **Diagnostic Settings** blade, you will see a **+ Add Diagnostic Setting** link. Specify the **Diagnostic Setting Name** and select the required logs, as shown in Figure 5-12.

Diagnostics setting

 Save  Discard  Delete  Provide feedback

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)

Diagnostic setting name *

NSG-DiagSetting



Category details

Destination details

log

Send to Log Analytics

NetworkSecurityGroupEvent

Subscription

Visual Studio Ultimate with MSDN



NetworkSecurityGroupRuleCounter

Log Analytics workspace

ExamRef-LAWorkspace (canadacentral)



Archive to a storage account

Stream to an event hub

FIGURE 5-12 Azure Monitor Diagnostic settings for a resource

Note Diagnostic logs

Each resource or tenant service on which you enable diagnostic logs will have varying controls (or settings). For example, not all resources support a retention policy in the diagnostic settings, and not all resources support sending metric data to another location.

When configuring diagnostics settings, you will select where the logs (and optionally metrics) are sent. Valid locations to send data will be to **Archive To A Storage Account**, **Stream To An Event Hub**, or **Send To Log Analytics** (see [Figure 5-14](#)). As you select each location, additional configuration will be required. For example, to archive to a storage account, you will need to select an existing storage account or create a new storage account.

For diagnostics logs that support retention with storage, you can select a retention period in days. A retention period of zero days means the logs will be retained forever. Valid numeric values for the number of days is any number between 1 and 365. If you set the retention period and have only selected an Event Hub or a Log Analytics Workspace(but have not selected a storage account), the retention settings will be ignored.

As you configure each resource or service, you can send the data from multiple log sources to the same destination. For example, you can send the diagnostic logs from a tenant service like Azure Active Directory to a Log Analytics Workspace, and you can send the diagnostics logs from a resource like a Network Security Group to the same Log Analytics Workspace.

It can take several moments for the setting to appear in the list of settings for the resource. Note that even though the setting has been configured, diagnostic data will not be collected until a new event is generated.

All these settings can be configured through the Azure portal, Azure PowerShell, the Azure CLI, or through the Azure Monitor REST API.



Exam Tip

The Azure Diagnostics agent can also be configured through resource manager templates and the command line tools by specifying a configuration file. For the exam you should be aware of the schema of this configuration and how to apply it using automated tools. You can learn more about the Azure Diagnostics schema at:
<https://docs.microsoft.com/azure/monitoring-and-diagnostics/azure-diagnostics-schema>.

Query and analyze logs

As mentioned earlier, Azure Monitor stores and surfaces two types of data: metrics and logs. Metrics are numerical values such as performance counters, while logs can be either numerical data or text. For instance, the full text of an exception that is raised in an application or even the text of an application log from a Windows or Linux server is one example.

Create a query

After the workspace has been configured, tenant logs, resource logs, and machines have been onboarded, you can begin to analyze and visualize data. To interact with the data in Log Analytics, you use log queries, which are used to:

- Perform interactive analysis of log data through the Azure portal in Azure Monitor and a Log Analytics Workspace.
- Build custom alert rules based on the logs in a workspace.
- Generate visualizations to can be shared through Azure Dashboards.
- Export custom data sets to Excel or Power BI.
- Perform automation based on log data with PowerShell or the Azure CLI.

Note Log query usage

To learn more about all the ways that log queries can be used, refer to the documentation at: <https://docs.microsoft.com/azure/azure-monitor/log-query/log-query-overview#where-log-queries-are-used>.

The query language used by Log Analytics is called Kusto. Kusto queries are used to generate read-only requests to process data and return results. This means that the logs stored in Log Analytics are immutable and are only removed from a workspace based on the retention configuration. Queries are authored in plain-text, and the schema used by Log Analytics is like SQL's with databases and tables composed of columns and rows. In each table, data is organized in columns with different data types as indicated by icons next to the column name. Column data types include text, numbers, and datetime.

Authored queries in Log Analytics can take many forms, from basic queries to very advanced queries with multiple aggregates and summarizations. Queries can be used to search terms, identify trends, analyze patterns, and provide many other insights. Queries search tables and can start with either a table name or a search command that defines scope. The pipe (|) character separates commands, and you can add as many commands as required.

In the following example, the Heartbeat table is queried to summarize the count of computers (by IP) and by a time value (`TimeGenerated`) to render a chart that tracks the number of computers reporting a workspace each hour.

[Click here to view code image](#)

```
// Chart the number of reporting computers each hour
Heartbeat
| summarize dcoun(ComputerIP) by bin(TimeGenerated, 1h)
| render timechart
```

To run this query, browse to Azure Monitor and select Logs to open the query interface. This query will not return data if you do not have any virtual machines deployed and running. Those machines must also be associated with the Log Analytics Workspace you are querying.

The query shown above is a table-based query. Queries always begin with a scope—either a table or search-based query. Kusto queries are case-sensitive. Typically, language keywords are written in lowercase. When using the names of tables and columns in queries, you must ensure you are using the correct case. Table-based queries target a single table in a Log Analytics Workspace (or

database), while search-based queries target all tables by default.

Table-based queries start by scoping the query, and therefore tend to be very efficient and generally faster than search queries. Search queries are less structured by nature, which makes them the better choice when searching for a specific value across columns or tables. In other words, a search can scan all columns in a given table or in all tables across an entire workspace for the defined value.

The amount of data being processed by a query could be enormous, which is why these queries can take longer to complete and might return large result sets which are limited by the Log Analytics service to 10,000 results.

To author queries in the Azure portal, browse to Azure Monitor and select the **Logs** blade. From this blade, you can access all the subscriptions and workspaces you have rights to read from. Azure Monitor offers many sample queries for heartbeats, performance, and usage across your machines and services tracked in Log Analytics (see [Figure 5-13](#)).

Home >

Monitor | Logs

Microsoft

Search (Ctrl+ /)

New Query 1 + Example queries Query explorer Documentation X

Overview Activity log Alerts Metrics Logs Service Health Workbooks

Virtual Machines

All Queries

Firewalls Front Doors IoT Hub Key Vaults Kubernetes Services Logic Apps Recovery Services ... Service Bus SQL Databases

SQL Managed Inst... Storage Accounts Stream Analytics jo... Traffic Manager Pr... Virtual Machine Sc... Virtual machines Other

Example queries

Resource Type Add filter

Favorites

Virtual Machines

Track VM availability
Display the VM's reported availability during the last day.

Run Example query

Virtual Machine available memory
Chart the VM's available memory over the last hour.

Run Example query

Chart CPU usage trends ① ★
Calculate CPU usage patterns over the last day, chart by percentiles.

Run Load to editor Example query

Virtual Machine free disk space
Show the latest report of free disk space, per instance.

Run Example query

What data is being collected?
List the collected performance counters and object types (Process, Memory, Processor...)

Run Example query

Reported errors
Show error events from the last hour.

Run Example query

Find Linux kernel events
Show the trend of a selected event

FIGURE 5-13 Azure Monitor logs

Important Azure Monitor Logs

The settings shown in Figure 5-28 apply to the entire workspace and cannot be configured per computer group.

Selecting a query and clicking **Load To Editor** will open an editor with query preview, as shown in Figure 5-14.

The screenshot shows the Azure Monitor Query editor interface. At the top, there's a header bar with a 'New Query 1*' button, a '+' button, and several icons for 'Example queries', 'Query explorer', and others. Below the header are buttons for 'ExamRef-LAWorkspace', 'Select scope', 'Run' (which is highlighted in blue), 'Time range: Last 24 hours', 'Save', 'Copy link', 'New alert rule', 'Export', 'Pin to dashboard', and three dots. On the left, there's a sidebar with tabs for 'Tables', 'Queries' (which is selected and highlighted in blue), and 'Filter'. A search bar is also present. The main area displays a numbered sample query:

```
1 // Chart CPU usage trends
2 // Calculate CPU usage patterns over the last day, chart by percentiles.
3 Perf
4 | where CounterName == "% Processor Time"
5 | where ObjectName == "Processor"
6 | summarize avg(CounterValue) by bin(TimeGenerated, 15min) // bin is used to set the time grain to 15 minutes
7 | render timechart
8 // Perf table stores performance counters for Windows and Linux computers
9 // Counters are specified using ObjectName (performance object), InstanceName and CounterName
10 // % Processor Time captures CPU activity, ObjectNames can be Processor, Process and Process Information
```

Below the main area, there's a 'Favorites' section with the text: 'You can add favorites by clicking on the star icon'.

FIGURE 5-14 Query editor with sample query

Save a query to the dashboard

In addition to sample queries, you can browse the schema for the currently selected workspace. This is useful for determining the proper case for table and column names as Kusto is a case-sensitive query language. Authored queries can be saved for later and/or marked as favorites so they can be retrieved later using the Query explorer (see [Figure 5-15](#)).

New Query 1*

ExamRef-LAWorkspace Select scope Run Time range: Set in query Save Copy link New alert rule Export

Tables Queries Filter

Search Group by: Resource Type Filters: not selected

- Kubernetes Services
- Logic Apps
- Recovery Services Vaults
- Service Bus
- SQL Databases
- SQL Managed Instances
- Storage Accounts
- Stream Analytics jobs
- Traffic Manager Profiles
- Virtual Machine Scale Sets
- Virtual machines

Track VM availability ★

Virtual Machine available memory ★

Chart CPU usage trends

Virtual Machine free disk space

What data is being collected?

Reported errors

Find Linux kernel events

```

1 // Chart CPU usage trends
2 // Calculate CPU usage patterns over the last day, chart by percentiles.
3 Perf
4 | where CounterName == "% Processor Time"
5 | where ObjectName == "Processor"
6 | summarize avg(CounterValue) by bin(TimeGenerated, 15min) // bin is used to set the time
7 | render timechart
8 // Perf table stores performance counters for Windows and Linux computers
9 // Counters are specified using ObjectName (performance object), InstanceName and CounterName
10 // % Processor Time captures CPU activity, ObjectNames can be Processor, Process and Procedure
11 ...

```

Results Chart Display time (UTC+00:00)

Completed

No results found from the last 24 hours
Try [selecting another time range](#) or [adding a custom time filter](#) to your query.

Name* CustomCPUUsageChart

Save as Query

Category* Virtual machines Custom

Save Cancel

FIGURE 5-15 Save or mark as favorite query

Interpret graphs

In Query Explorer, you also get an option to generate charts/graphs based on the log queries. In the output pane, you can select **Chart** to see the graphical representation of query results. You can choose the display option from various categories (from column, bar chart, line, pie, or area). For one of the sample queries, the stacked bar chart is shown in [Figure 5-16](#).

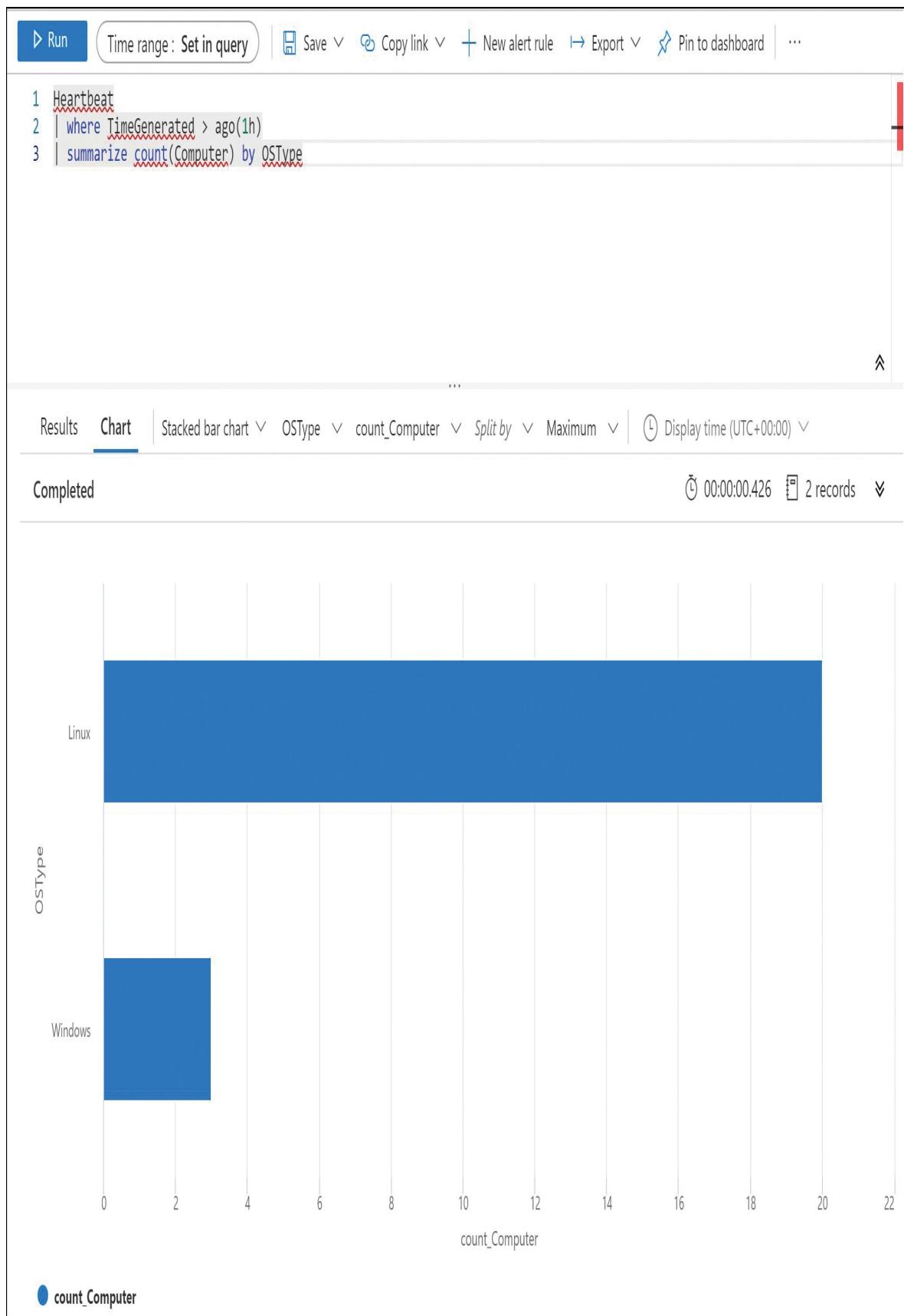


FIGURE 5-16 Stacked bar chart

Similarly, for the same query, the doughnut pie chart is shown in [Figure 5-17](#).

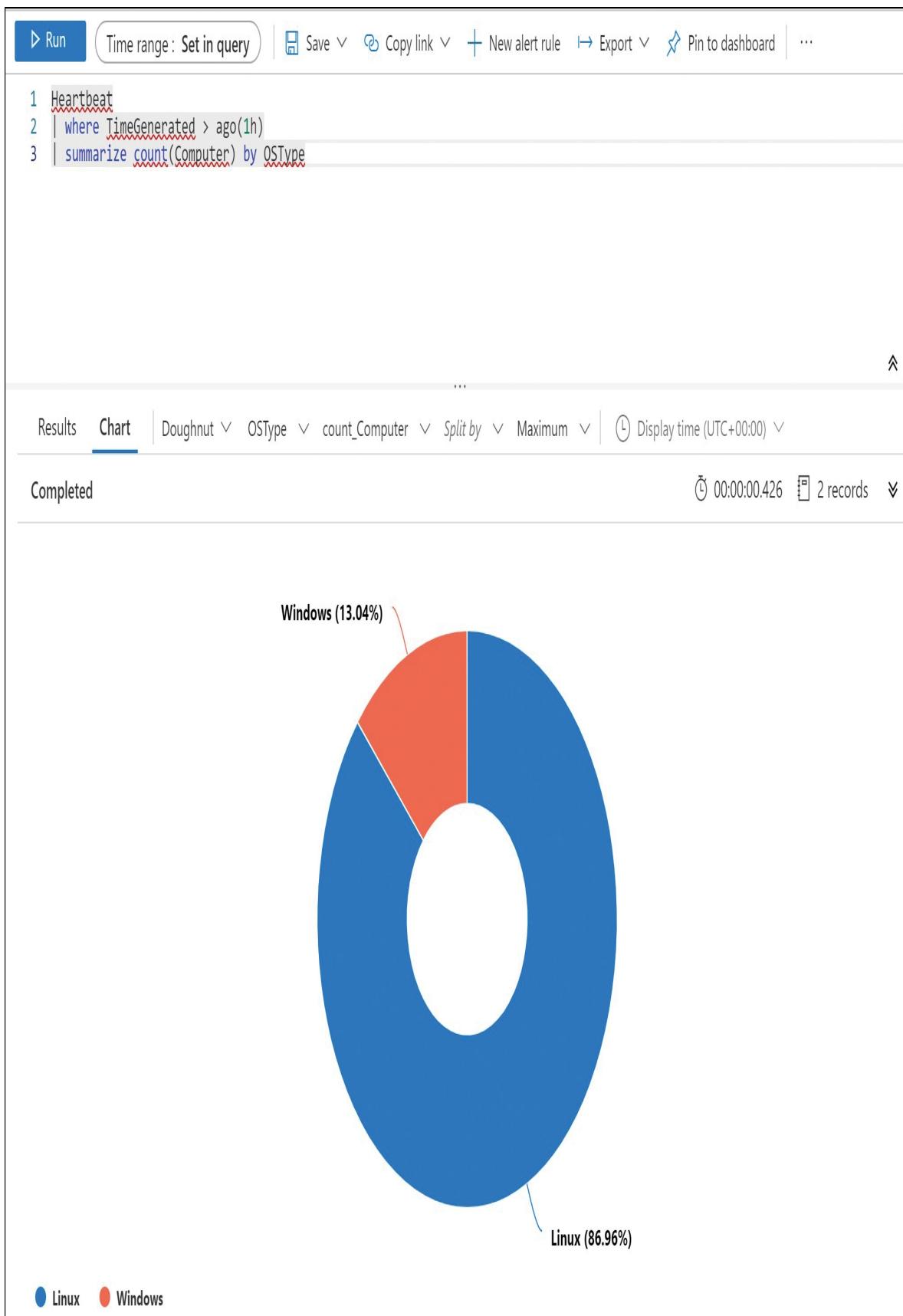


FIGURE 5-17 Doughnut pie chart

You can hover on graph to interpret the results directly from the graph. The highlighted sector will be displayed with its query result, as shown in [Figure 5-18](#).

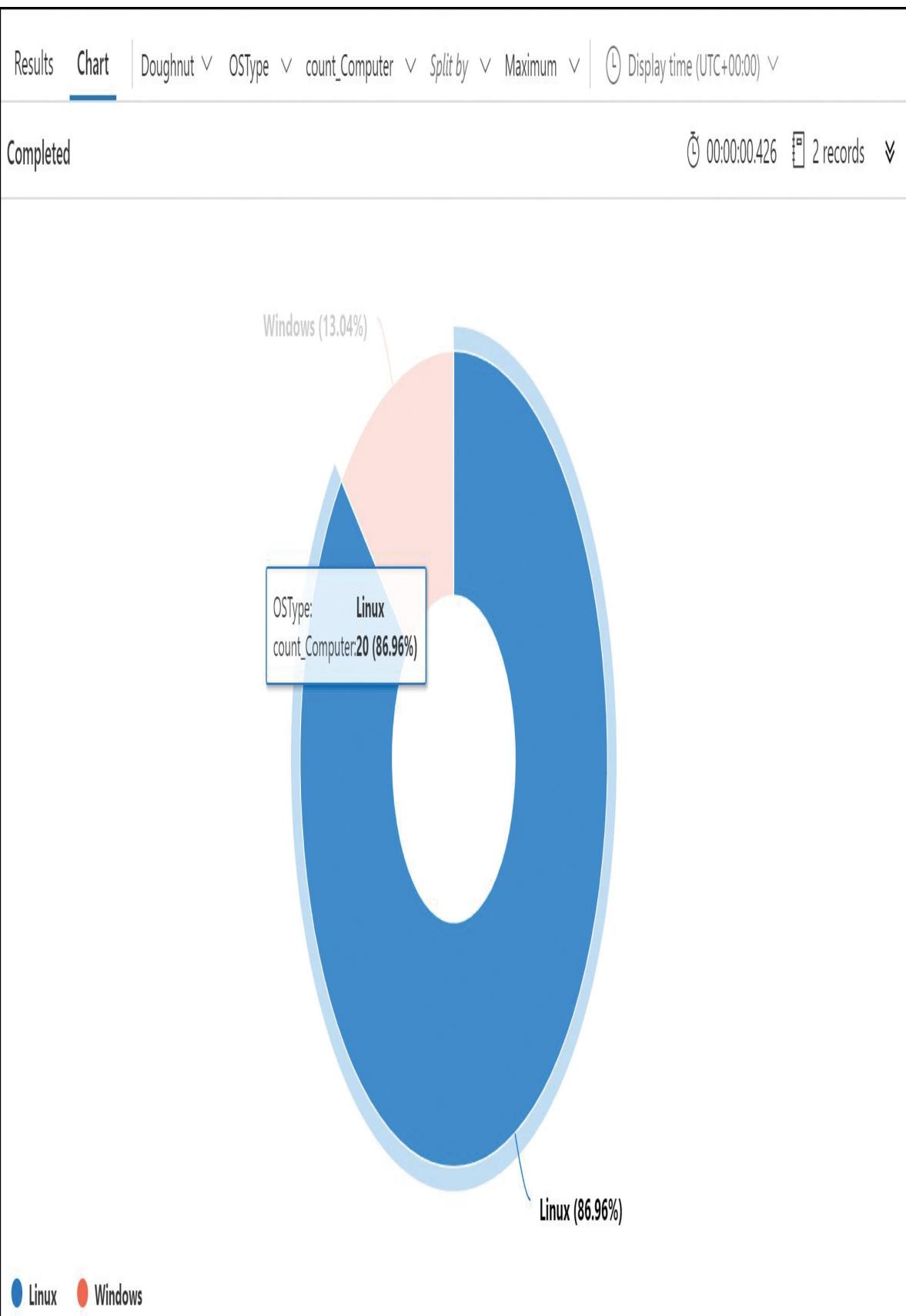


FIGURE 5-18 Save or mark as favorite query

Set up alerts and actions

Alerts proactively notify you when important conditions are found in your monitoring data. They allow you to identify and address issues before the users of your system notice them.

Azure Monitor brings a unified alerting experience to Azure, with a single pane of glass for interacting with metrics, the Activity Log, Log Analytics, service and resource health, and service-specific insights that provide out-of-the-box dashboards with visualizations and queries for:

- Custom applications with Application Insights
- Virtual Machines
- Storage accounts
- Containers
- Networks
- Key vaults (preview)

Alerts have multiple notification options, including:

- Email
- SMS
- Push notifications to the Azure mobile app
- Voice
- Integration with automation services.

Alerts that are generated within Azure Monitor can invoke Azure Automation runbooks, Logic Apps, Azure Functions, and even generate incidents in third-party IT Service Management tools, such as ServiceNow.

Create and test alerts

To create an alert rule, click **Alerts** from within the Azure resource configuration blade or browse to Azure Monitor in the Azure portal, select the **Alerts** blade, and select **+New Alert Rule**, as shown in [Figure 5-19](#).

Home >

Monitor | Alerts

Microsoft

Search (Ctrl+ /) + New alert rule Manage alert rules Manage actions View classic alerts Refresh Provide feedback

Overview Activity log Alerts Metrics Logs Service Health Workbooks

Don't see a subscription? Open Directory + Subscription settings

Subscription * ① Resource group ① Resource Time range

Visual Studio Ultimate with MSDN 2 selected Past 24 hours

Selected subscriptions Selected resource groups

Pay attention to what matters.

You have not configured any alert rules.

Configure alert rules and attend to fired alerts to efficiently monitor your Azure resources. [Learn more](#)



+ New Alert Rule

The classic alerts can be accessed from [here](#).

FIGURE 5-19 The alerts blade from within Azure Monitor

Alerts in Azure Monitor are centered on alert rules. Alert Rules contain the following components:

- A target resource (or resource type)
- Conditional logic for the alert with criteria based on the available signals for the target resource
- An Action Group, or what should happen when the alert rule condition is met
- A name and description for the alert rule

Note Azure Monitor alert rules

Alert Rules in Azure Monitor are not the same as alerts. They are the criteria used to evaluate when an alert should be generated. An alert is generated based on the rule, and then the alerts themselves are acted upon separately, even maintaining their own state (such as **New** or **Closed**).

Pick the target for the alert, which determines the available signals by clicking the **Select Resource** button, as shown in Figure 5-20.

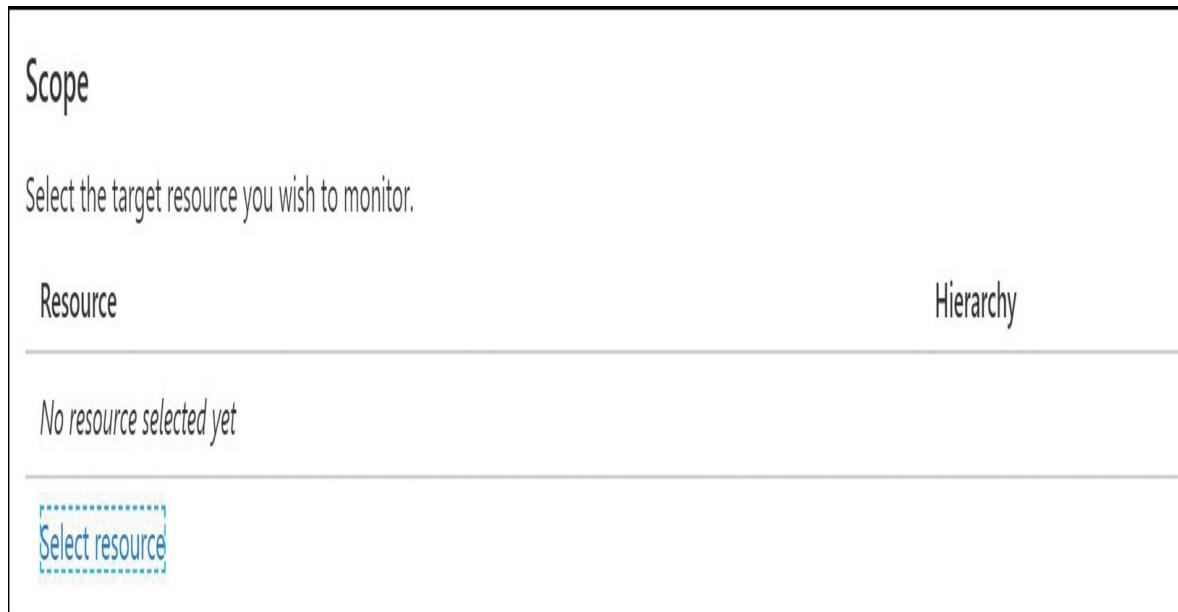


FIGURE 5-20 Azure Monitor Create alert rule

The target resource defines the scope and signals available for the alert. A target resource is an Azure resource that generates signals (such as metrics or the Activity Log) such as a virtual machine or storage account. The signal types that are available for monitoring vary based on the selected target (or targets as you can select more than one target) and the available signal types are as follows:

- Metrics
- Log search queries
- Activity Logs

For instance, selecting Subscription will allow you to select Activity Log signals. Selecting a single resource like a virtual machine will allow you to select signals that include both the Activity Log and metrics, as shown in [Figure 5-21](#).

Select a resource

X

Select the resource(s) you want to monitor. Available signal types for your selection will show up on the bottom right.

Filter by subscription * ⓘ

Visual Studio Ultimate with MSDN ✓

Filter by resource type ⓘ

Virtual machines ✓

Filter by location ⓘ

All ✓

Search to filter items...

Resource	Resource type	Location
<input checked="" type="checkbox"/>  Visual Studio Ultimate with MSDN	Subscription	Canada Central
<input checked="" type="checkbox"/>  examref-rg	Resource group	Canada Central
<input checked="" type="checkbox"/>  ExamRef-VM	Virtual machine	Canada Central
<input checked="" type="checkbox"/>  ExamRef-VMs	Virtual machine	Canada Central

FIGURE 5-21 Azure Monitor alert target

The next step is to configure the alert criteria by clicking the **Select Condition** button, as shown in Figure 5-22.

Condition

Configure when the alert rule should trigger by selecting a signal and defining its logic.

Condition name

No condition selected yet

Select condition

FIGURE 5-22 Azure Monitor Add condition

The condition will allow you to select the signal from the available signals for the target and define the logic test that will be applied to the data from the signal. For example, for a virtual machine you can use the Percentage CPU metric to generate an alert based on a custom threshold for CPU usage, as shown in [Figure 5-23](#). The alert logic conditions are different for Activity Log signals or metric signals.

Configure signal logic

X

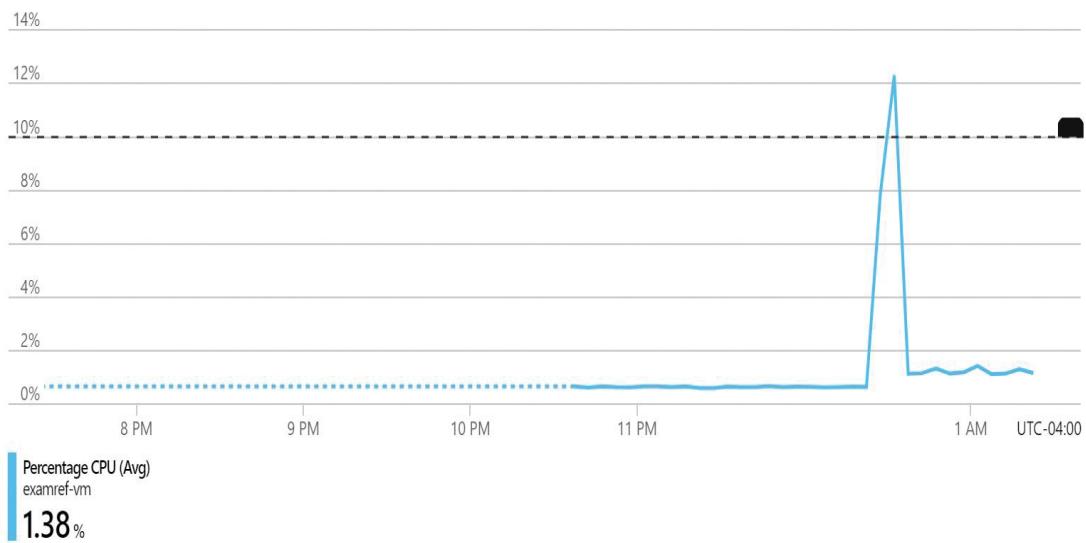
Select time series ⓘ

ExamRef-VM; Aggregate

< Prev Next >

Chart period ⓘ

Over the last 6 hours



Alert logic

Threshold ⓘ

Static

Dynamic

Operator ⓘ

Less than or equal to

Aggregation type * ⓘ

Average

Threshold value * ⓘ

10

✓

%

Condition preview

Whenever the average percentage cpu is less than or equal to 10 %

Evaluated based on

Aggregation granularity (Period) * ⓘ

5 minutes

Frequency of evaluation ⓘ

Every 5 Minutes

Done

FIGURE 5-23 Azure Monitor alert condition

Configure one or more conditions for the alert rule. After the conditions are defined, proceed to Action Groups by clicking the **Select Action Group** button, as shown in [Figure 5-24](#). An Action Group is a collection of actions that should occur in response to an alert being triggered.

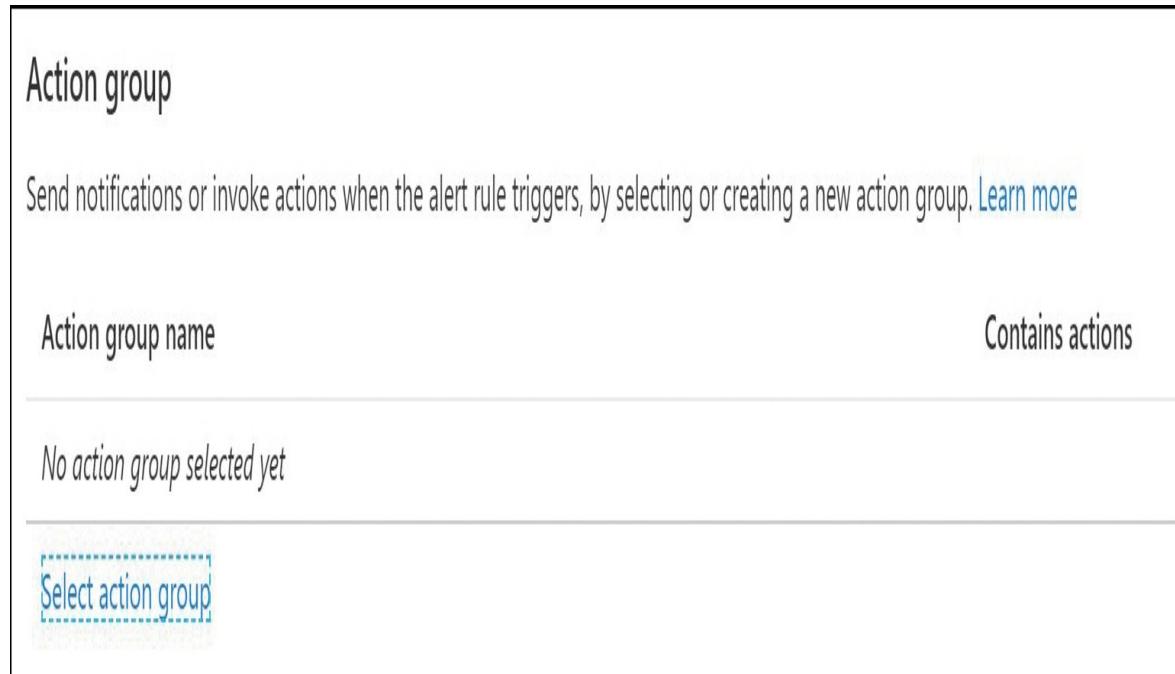


FIGURE 5-24 Azure Monitor Action Groups

Select the existing Action Group if you already have otherwise click **Create Action Group** button to create a new Action Group, as shown in [Figure 5-25](#).

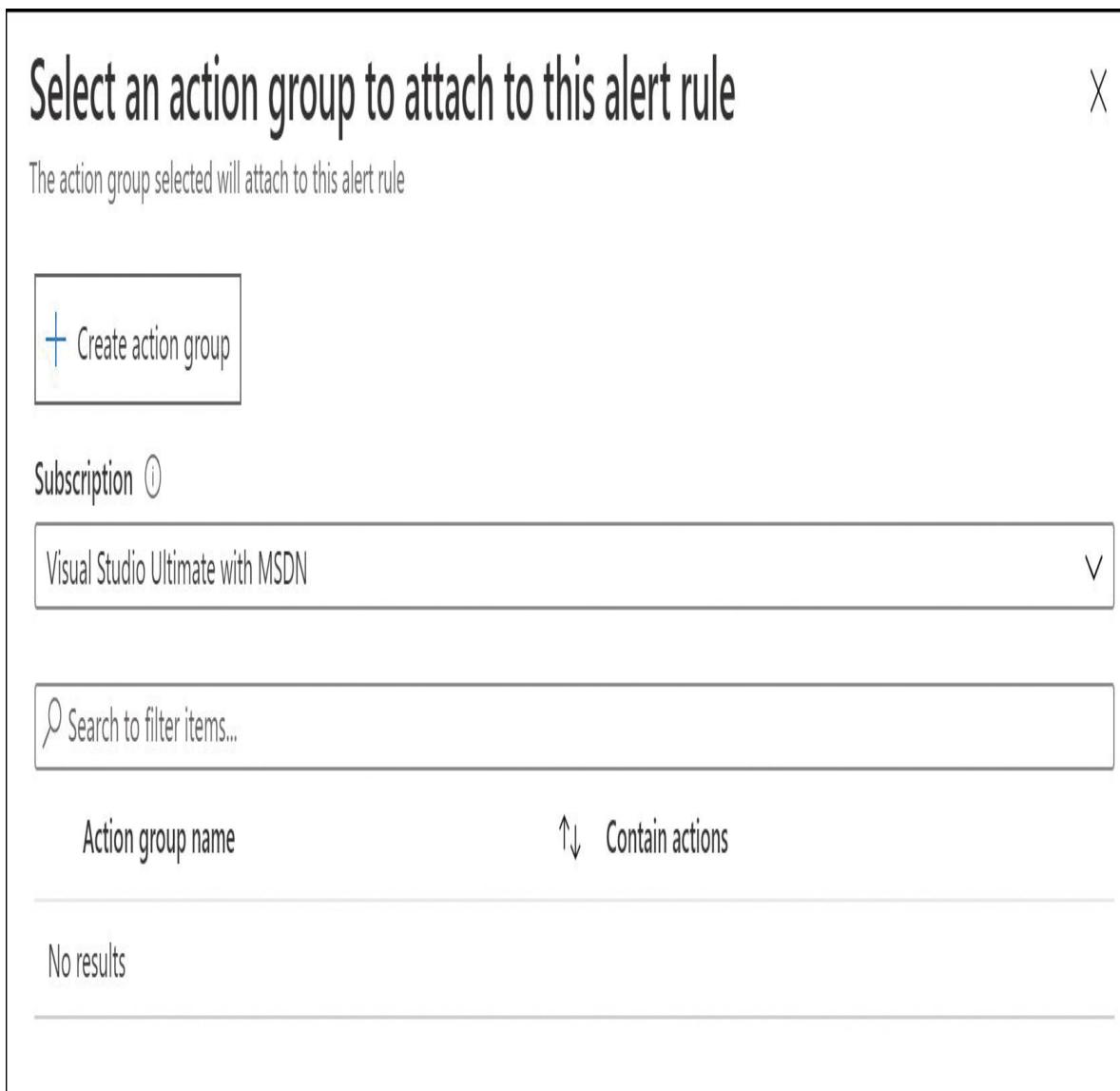


FIGURE 5-25 Create Action Group

Note Action groups

Action groups are separate resources and are independent of the alert rule. This means that the same Action Group can be used across multiple alert rules.

When creating a new Action Group, define the **Action Group Name**, **Display Name**, **Subscription**, and **Resource Group** in which the Action Group

will be created (see [Figure 5-26](#)).

On the next screen, you can configure notifications. Select **Email Azure Resource Manager Role** from the **Notification Type** drop-down menu and select the respective roles from the drop-down menu that appears below the **Notification Type**, as shown in [Figure 5-27](#).

Create action group

Basics

Notifications Actions Tags Review + create

An action group invokes a defined set of notifications and actions when an alert is triggered. [Learn more](#)

Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Resource group * ⓘ

ExamRef-RG



[Create new](#)

Instance details

Action group name * ⓘ

ExamRef-ActionGroup



Display name * ⓘ

ExamRef-AG



This display name is limited to 12 characters

FIGURE 5-26 Basics blade - Create Action Group

Create action group

Basics Notifications Actions Tags Review + create

Notifications

Configure the method in which users will be notified when the action group triggers. Select notification types, provide receiver details and add a unique description. This step is optional.

Notification type 

Name 

Selected 

Email Azure Resource Manager Role  

Email Azure Resource Manager Role

Add or edit an Email Azure Resource Manager Role action

Azure Resource Manager Role  Learn more *

Owner 

Owner

Contributor

Reader

Monitoring Contributor

Monitoring Reader

Review + create

Previous

Next: Actions >

FIGURE 5-27 Notifications blade - Create Action Group

Alternatively, you can configure other notifications by selecting **Email/SMS Message/Push/Voice** as Notification type and select respective options from the pop-up screen, as shown in [Figure 5-28](#). You may have up to 1,000 email actions and 10 SMS/Voice actions in an Action Group.

Create action group

Basics Notifications Actions Tags Review + create

Notifications

Configure the method in which users will be notified when the action group triggers. Select notification types, provide receiver details and add a unique description. This step is optional.

Notification type	Name	Selected
Email Azure Resource Manager Role	NotifyOwner	Owner 
Email/SMS message/Push/Voice	<input type="text"/>	
	<input type="text"/>	

Email/SMS message/Push/Voice

Add or edit an Email/SMS/Push/Voice action

Email

Email *

SMS (Carrier charges may apply)

Country code

1

Phone number

Azure app Push Notifications

Azure account email

Voice

Country code *

1

Phone number *

Enable the common alert schema. [Learn more](#)

OK

[Review + create](#)

[Previous](#)

[Next: Actions >](#)

FIGURE 5-28 Notifications blade - Create Action Group

In addition to sending email notifications, you can execute the following actions:

- **Runbook.** A set of PowerShell code that runs in the Azure Automation Service. See the following to learn more about using Runbooks to remediate alerts at: <https://azure.microsoft.com/blog/automatically-remediate-azure-vm-alerts-with-automation-runbooks/>.
- **Function Apps.** A Function App is a set of code that runs “serverless” that can respond to alerts. This functionality requires Version 2 of Function Apps, and the value of the `AzureWebJobsSecretStorageType` app setting must be set to `files`.
- **ITSM.** You may have up to 10 IT Service Manager (ITSM) actions with an ITSM connection. The following ITSM providers are currently supported: ServiceNow, System Center Service Manager, Provance, and Cherwell. You can learn more about ITSM connections here: <https://docs.microsoft.com/azure/azure-monitor/platform/itsmc-overview>.
- **Logic Apps.** A Logic App provides a visual designer to model and automate your process as a series of steps known as a workflow. There are many connectors across the cloud and on-premises to quickly integrate across services and protocols. When an alert is triggered the Logic App can take the notification data and use it with any of the connectors to remediate the alert or start other services. To learn more about Azure Logic Apps visit: <https://docs.microsoft.com/azure/logic-apps/logic-apps-what-are-logic-apps>.
- **Webhook.** A webhook allows you to route an Azure alert notification to other systems for post-processing or custom actions. For example, you can use a webhook on an alert to route it to services that send text messages, log bugs, notify a team via chat/messaging services, or do any number of other actions. You can learn more about sending alert information to webhooks at: <https://docs.microsoft.com/azure/monitoring-and-diagnostics/insights-webhooks-alerts>.

You can configure above actions for the Action Group on the next blade. Select from the options available in the **Action Type** drop-down menu, as shown in Figure 5-29.

Create action group

Basics Notifications Actions Tags Review + create

Actions

Configure the method in which actions are performed when the action group triggers. Select action types, fill out associated details, and add a unique description. This step is optional.

Action type ⓘ	Name ⓘ	Selected ⓘ
Automation Runbook		
Azure Function		
ITSM		
Logic App		
Secure Webhook		
Webhook		

FIGURE 5-29 Actions blade - Create Action Group

Once Action group is created, specify remaining alert rule details such as Alert rule name, description, resource group to save the alert, severity and enable alert upon creation option (see [Figure 5-30](#)).

Alert rule details

Provide details on your alert rule so that you can identify and manage it later.

Alert rule name * ⓘ

ExamRefAlertRule



Description

Whenever the average percentage cpu is less than or equal to 10 % then
Shutdown the VM.



Save alert rule to resource group * ⓘ

ExamRef-RG



Severity * ⓘ

Sev 3



Enable alert rule upon creation



Create alert rule

FIGURE 5-30 Action GroupAlert rule details

View alerts in Azure Monitor

After an alert rule has been created, the alert rule and Action Group can be managed through Azure Monitor from the **Alerts** blade by selecting **Manage Alert Rules**. Alerts can be managed across multiple subscriptions and can be filtered by **Resource Group**, **Resource Type**, **Signal Type**, and **Status** (see [Figure 5-31](#)).

Rules

Rules management

+ New alert rule Edit columns Manage actions View classic alerts Refresh | Enable Disable Delete

Don't see a subscription? [Open Directory + Subscription settings](#)

Subscription*

Visual Studio Ultimate with MSDN

Resource group

2 selected

Resource type

Virtual machines

Resource

V

Signal type

All signal types

Status

Enabled

[Selected subscriptions](#) > [Selected resource groups](#)

Displaying 1 - 1 rules out of total 1 rules

Search alert rules based on rule name and condition...

Name

Condition

Status

Target resource

Target resource type

Signal type

ExamRefAlertRule

Whenever the average percentage cpu... Enabled

ExamRef-VMs, ExamRef-VM

Virtual machines

Metrics

FIGURE 5-31 Azure Monitor new action alert rule details

Alert Rules do not generate alerts immediately, and metric alerts can take up to 10 minutes. When alerts are generated, they will be distributed based on the actions defined in the Action Group. For example, when an email is sent, the defined users will receive a message with the alert details and a link to view the alert in the Azure portal, as shown in [Figure 5-32](#).



⚠ Your Azure Monitor alert was triggered

Azure monitor alert rule ExamRefAlertRule was triggered for ExamRef-VM at August 17, 2020 6:53 UTC.

Alert rule description Whenever the average percentage cpu is less than or equal to 10 % then Shutdown the VM.

Rule ID [REDACTED]

[View Rule >](#)

Resource ID [REDACTED]

[View Resource >](#)

Alert Activated Because:

Metric name Percentage CPU

Metric namespace virtualMachines/ExamRef-VM

Dimensions [REDACTED]

microsoft.resourceType = Microsoft.Compute/virtualMachines

Time Aggregation Average

FIGURE 5-32 Azure Monitor alert notification email

Note Alert Rules

You can enable and disable alert rules as needed to meet your requirements.

When an alert is resolved by the state of the monitor condition and changed to Resolved, notifications are sent as well.

Analyze alerts across subscriptions

When an alert rule is created, the alert rule targets resources in a single subscription, and the alerts that are generated based on the alert rules are associated with the subscription from which they are generated. Azure operators are not limited to viewing alerts from only a single subscription through Azure Monitor, which again, provides a single pane of glass for not only managing alert rules across multiple subscriptions, but also for managing the generated alerts.

Recall that alert rules and Action Groups are separate entities. The alerts that are generated based on the conditional logic of an alert rule are separate entities as well. This means that they are managed independently of alert rules and maintain their own state.

Alerts can have one of three states:

- **New.** The alert is new and has not been reviewed
- **Acknowledged.** The issue that generated the alert is being actioned by an administrator
- **Closed.** The issue that generated the alert has been resolved, and the alert has been marked as closed

The state of an alert is updated by the user who is interacting with the alert and is not updated automatically by the Azure platform.

Note Alert state

Alert state is not the same as the monitor condition of an alert. When the Azure platform generates an alert based on an alert rule, the alert's monitor

condition is set to *fired* and when the underlying condition clears, the monitor condition is set to *resolved*.

As alerts are generated, they will appear on the **Alerts** blade in Azure Monitor. From the **Alerts** blade, can you view alerts for all subscriptions, and drill into one or more specific **Subscriptions**, **Resource Groups**, and **Resources**. Also, you can filter by Time Range by choosing **Past Hour**, **Past 24 Hours**, **Past 7 Days**, or **Past 30 Days** from the drop-down menu (see [Figure 5-33](#)).

Home >

Monitor | Alerts

Microsoft

Search (Ctrl+ /) New alert rule Manage alert rules Manage actions View classic alerts Refresh Provide feedback

Overview Activity log Alerts Metrics Logs Service Health Workbooks

Don't see a subscription? Open Directory + Subscription settings

Subscription * Resource group Resource Time range

Visual Studio Ultimate with MSDN 2 selected ExamRef-VM Past 24 hours

Selected subscriptions > Selected resource groups > ExamRef-VM

Total alerts Smart groups (preview) Total alert rules Action rules (preview) Learn more About alerts

1 0 1 0 Since 8/16/2020, 3:01:00 AM 0% Reduction Enabled 1 Enabled 0

Severity	Total alerts	New	Acknowledged	Closed
Sev 0	0	0	0	0
Sev 1	0	0	0	0
Sev 2	0	0	0	0
Sev 3	1	1	0	0
Sev 4	0	0	0	0

Applications Virtual Machines Storage accounts Containers Networks (preview) Azure Cosmos DB Azure Cache for Redis (preview) Key Vaults (preview)

The screenshot shows the Azure Monitor Alerts page. On the left, there's a sidebar with links for Overview, Activity log, Alerts (which is selected), Metrics, Logs, Service Health, and Workbooks. The main area has a search bar and navigation buttons for creating new alerts, managing alert rules, and so on. It displays a message about not seeing a subscription and links to open directory and subscription settings. Below that, it shows subscription details: Visual Studio Ultimate with MSDN, 2 selected resources (ExamRef-VM), and a time range of the past 24 hours. A breadcrumb trail indicates the path: Selected subscriptions > Selected resource groups > ExamRef-VM. The main summary section shows 1 total alert, 0 smart groups, 1 total alert rule, and 0 action rules. It also shows the alert was created since 8/16/2020 at 3:01:00 AM with 0% reduction and 1 enabled rule. Below this is a table of alerts by severity: Sev 0 (0 alerts), Sev 1 (0 alerts), Sev 2 (0 alerts), Sev 3 (1 alert, highlighted in blue), and Sev 4 (0 alerts). At the bottom, there are links for Applications, Virtual Machines, Storage accounts, Containers, Networks (preview), Azure Cosmos DB, Azure Cache for Redis (preview), and Key Vaults (preview).

FIGURE 5-33 Azure Monitor Alerts dashboard

Selecting one of the links on the dashboard (such as **Total Alerts**) will open the **All Alerts** blade, as shown in [Figure 5-34](#).

All Alerts

X

[+ New alert rule](#) [Edit columns](#) [Manage alert rules](#) [View classic alerts](#) [Refresh](#) | [✓ Change state](#) [Provide feedback](#)Don't see a subscription? [Open Directory + Subscription settings](#)Subscription ①

Visual Studio Ultimate with MSDN

Resource group ①

ExamRef-RG

Resource type ①

7 selected

Resource ①

ExamRef-VM

Time range ①

Past 24 hours

Monitor service ①

21 selected

Monitor condition ①

2 selected

Severity ①

5 selected

Alert state ①

3 selected

Smart group id ①

Smart group id

[Selected subscriptions](#) > [ExamRef-RG](#) > [ExamRef-VM](#)[All alerts](#) [Alerts by smart group \(preview\)](#)Action rules (preview) allows you to define actions at scale as well as suppress actions. Learn more about this functionality [here](#) Search by name (case-insensitive)

Name

Severity

Monitor condition

Alert state

Affected resource

Monitor service

Signal type

Fired time

Subscription

 ExamRefAlertRule

Sev3

Fired

New

Platform

Metric

8/17/2020, 2:53 AM

Visual Studio Ultimate wit...

FIGURE 5-34 Azure Monitor All alerts blade

The view on this page can be filtered through the drop-down menus on the page and you can also filter, sort, and edit the columns that are displayed with the following limitations:

- When you filter by subscription, you are limited to selecting a maximum of five subscriptions.
- When filtering by resource group, you can only select one resource group at a time.
- The **Resource Type** filter is dynamic and is based on the selection of the resource group. You will not be able to select resource types that are not deployed to the selected resource group you are filtering with.
- The **Time Range** filter shows only alerts fired within the selected time window and supported values are the past hour, the past 24 hours, the past 7 days, and the past 30 days.

Selecting an alert will open the alert details (see [Figure 5-35](#)). From this blade, you can view Alert History, including any changes to monitor condition state. This is also where you can alter the alert state to New, Acknowledged, or Closed. If the state of an alert is changed, that change is included in the alert history for audit purposes.

ExamRefAlertRule

X

Alert Name

Summary History Diagnostics

Severity
Sev3

Fired time
8/17/2020, 2:53:46 AM

Affected resource
 examref-vm

Hierarchy
 Visual Studio U... >  examref-rg

State
New

Monitor condition
 Fired

[Change alert state](#)



Criterion

Metric name
Percentage CPU

Metric namespace
Microsoft.Compute/virtualMachines

Time aggregation
Average

Operator
LessThanOrEqual

Threshold
10 %

Metric value (when alert fired)
1.3 %

Dimension Name

Dimension Value

microsoft.resourceId

microsoft.resourceType

Microsoft.Compute/virtualMachines

Description
Whenever the average percentage cpu is less than or equal to 10 %
then Shutdown the VM.

Target resource type
microsoft.compute/virtualmachines

Monitor service
Platform

Signal type
Metric

Alert id
b09010a4-0292-4017-86e0-c4a9af61acec

Alert rule
[ExamRefAlertRule](#)

Suppression Status
None

FIGURE 5-35 Azure Monitor alert details

Configure Application Insights

Application Insights is used for development and as a production monitoring solution. It works by installing a package into your app, which can provide a more internal view of what's going on with your code. Its data includes response times of dependencies, exception traces, debugging snapshots, and execution profiles. It provides powerful smart tools for analyzing all this telemetry both to help you debug an app, and to help you understand what users are doing with it. You can tell whether a spike in response times is caused by something in an app or an external resourcing issue. Application Insights provides significantly more value when your application is instrumented to emit custom events and exception information.

To create application insight resource, open **Azure Monitor** and under **Insights**, select **Applications** from the left-side navigation pane and choose the **Create Application Insight Apps** button, as shown in [Figure 5-36](#).

Home >

Monitor | Applications

X

Microsoft

Search (Ctrl+ /) Add Edit columns Refresh Assign tags Delete

Subscriptions: Visual Studio Ultimate with MSDN

Filter by name... All resource groups All locations All tags No grouping

0 items

Name ↑ Type ↑ Resource group ↑ Location ↑ Subscription ↑

No Application Insights app to display

Try changing your filters if you don't see what you're looking for.

Create Application Insights apps

- Overview
- Activity log
- Alerts
- Metrics
- Logs
- Service Health
- Workbooks
- Insights
- Applications
- Virtual Machines
- Storage accounts
- Containers
- Networks (preview)
- Azure Cosmos DB
- Azure Cache for Redis (preview)
- Key Vaults (preview)

FIGURE 5-36 Create Application Insights apps

On the **Basics** blade, select the **Subscription**, **Resource Group**, **Region**, **Resource Mode**, and **Log Analytics Workspace** and specify the Name (see [Figure 5-37](#)).

Application Insights

Monitor web app performance and usage

Basics Tags Review + create

Create an Application Insights resource to monitor your live web application. With Application Insights, you have full observability into your application across all components and dependencies of your complex distributed architecture. It includes powerful analytics tools to help you diagnose issues and to understand what users actually do with your app. It's designed to help you continuously improve performance and usability. It works for apps on a wide variety of platforms including .NET, Node.js and Java EE, hosted on-premises, hybrid, or any public cloud. [Learn More](#)

PROJECT DETAILS

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Resource Group * ⓘ

ExamRef-RG

[Create new](#)

INSTANCE DETAILS

Name * ⓘ

ExamREF-Applnights

Region * ⓘ

(US) East US

Resource Mode * ⓘ

Classic Workspace-based (preview)

WORKSPACE DETAILS

Subscription * ⓘ

Visual Studio Ultimate with MSDN

Log Analytics Workspace * ⓘ

ExamRef-LAWorkspace [canadacentral]

[Review + create](#)

[« Previous](#)

[Next : Tags >](#)

FIGURE 5-37 Basics blade – Application Insights

Application Insights has an extensive dashboard depicting all the aspects of your application workload, as shown in [Figure 5-38](#). The dashboard displays application performance, usage, diagnostic, and other app data. The dashboard can be customized based on your preferences.

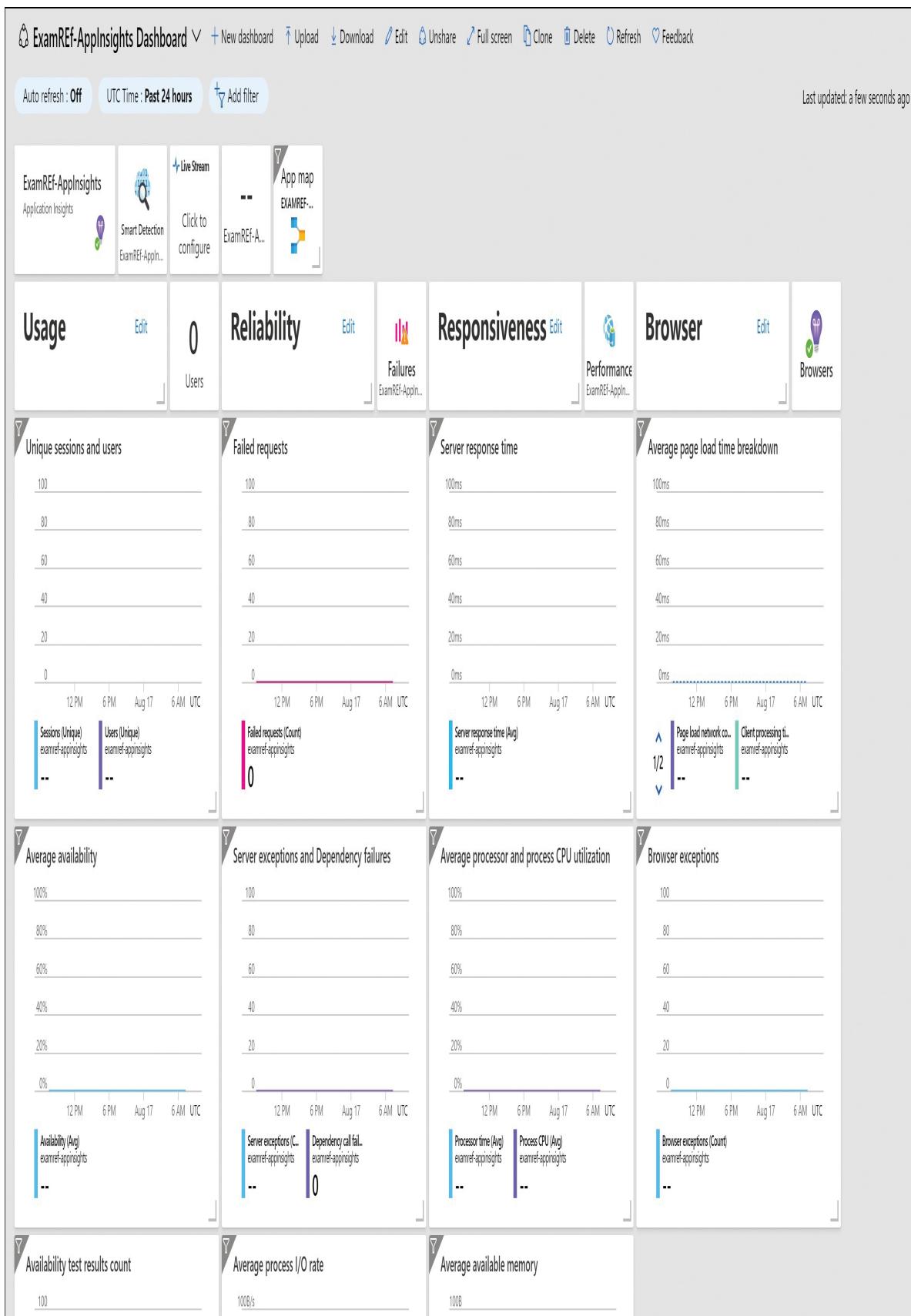


FIGURE 5-38 Application Insights Dashboard

You can learn more about Application Insights including samples for emitting custom telemetry at: <https://docs.microsoft.com/azure/azure-monitor/app/app-insights-overview/>.

Skill 5.2: Implement backup and recovery

Azure Backup is a service that allows you to backup on-premises servers, cloud-based virtual machines, and virtualized workloads such as SQL Server and SharePoint to Microsoft Azure. It also supports backup of Azure Storage file shares.

This section covers how to:

- [Create a Recovery Services vault](#)
- [Create and configure backup policy](#)
- [Perform backup and restore operations by using Azure Backup](#)
- [Perform site-to-site recovery by using Azure Site Recovery](#)
- [Configure and review backup reports](#)

Create a Recovery Services Vault

Within Azure, a single resource is provisioned for either Azure Backup or Azure Site Recovery. This resource is called a Recovery Services Vault. It is also the resource that is used for configuration and management of both Backup and Site Recovery.

Create a Recovery Services Vault (Azure portal)

To create a Recovery Services Vault from the Azure portal, follow these steps:

1. Click **Create A Resource**, and in the marketplace search dialog box, type **Backup and Site Recovery** and click the **Backup And Site Recovery** option.

2. On the marketplace page for **Backup And Site Recovery**, click **Create**.
3. Enter the name of the vault and choose the resource group where it resides or create a new resource group.
4. Then choose the region where you want to create the resource, as shown in [Figure 5-39](#).

Home > New > Backup and Site Recovery >

Create Recovery Services vault

Preview

* Basics Tags Review + create

Project Details

Select the subscription and the resource group in which you want to create the vault.

Subscription * ⓘ

Visual Studio Ultimate with MSDN



Resource group * ⓘ

ExamRef-RG



[Create new](#)

Instance Details

Vault name * ⓘ

ExamRef-RSVault



Region * ⓘ

Canada Central



FIGURE 5-39 Completing the creation of the vault

Use Soft Delete to recover Azure VMs

The default behavior of deleting a backup is that the backup is deleted and lost forever. Soft Delete is a feature that allows you to save and recover your data when backup data are deleted even in the event of an overwrite. This feature must be enabled in Recovery Services Vault by choosing **Properties -> Security Settings** (see [Figure 5-40](#)). When you use Soft Delete, backup data is retained for 14 days after deletion.

More Info Soft delete for Azure VM Backup

You can learn more about using Soft Delete with Azure VM Backup at <https://docs.microsoft.com/azure/backup/backup-azure-security-feature-cloud#soft-delete>.

ExamRef-RSV | Properties

Recovery Services vault

Search (Ctrl+ /) Visual Studio Ultimate with MSDN

- Overview**
- Activity log**
- Access control (IAM)**
- Tags**
- Diagnose and solve problems**
- Settings**
 - Identity**
 - Private endpoint connections**
 - Properties** (selected)
 - Locks**
 - Export template**
- Getting started**
 - Backup**
 - Site Recovery**
- Protected items**
 - Backup items**
 - Replicated items**
- Manage**
 - Backup policies**
 - Backup Infrastructure**
 - Site Recovery infrastructure**
 - Recovery Plans (Site Recovery)**

Security Settings

ExamRef-RSV

Configuration

Have you configured Azure Multi-Factor Authentication?

Soft Delete (For Azure Virtual Machines)

Soft delete protects backup data from accidental deletes. [Learn More](#)

Security Features

Use the following versions to enable Security features:

- MAB agent – use minimum agent version 2.0.9052
- Azure backup server – use minimum MAB agent version 2.0.9052 with Azure Backup Server upgrade 1
- DPM – use minimum MAB agent version 2.0.9052 with DPM 2012 R2 UR12 or DPM 2016 UR2

Security features cannot be disabled, once enabled.

Feature	Enabled	Disabled
Retention of deleted backup data	<input checked="" type="checkbox"/> Backup data retained for 14 days after delete operation	⚠ Instant deletion prevents recoverability from attacks
Minimum retention range checks	<input checked="" type="checkbox"/> Ensures more than one recovery point in case of attacks	⚠ Only one recovery point available for recovery
Alerts and notifications	<input checked="" type="checkbox"/> For critical operations like Stop backup with delete data	⚠ No security alerts or notifications for critical operations
Multiple layers of security	<input checked="" type="checkbox"/> Security PIN required for critical operations	⚠ Single layer of protection

FIGURE 5-40 Enabling Soft Delete on Recovery Services Vault

If the **Soft Delete** option is enabled, you can delete the backup data by clicking the **Stop Backup** button and then selecting **Delete Backup Data** with appropriate reason. Once deleted, your soft-deleted backup item will appear, as shown in Figure 5-41.

ExamRef-VMs

Backup Item

Backup now ⏪ Restore VM ⏪ File Recovery ⏪ Stop backup ⏪ Resume backup ⏪ Delete backup data ⏪ Restore to Secondary Region ⏪ Undelete

⚠ The restore points for this backup item have been deleted and retained in the soft delete state. They were deleted 21 second(s) ago and will be available for 13 more day(s) to recover after which they will be permanently deleted. For more information, [Click here.](#) →

Alerts and Jobs

Backup status

Summary

[View all Alerts](#) (last 24 hours)

Backup Pre-Check Passed

Recovery services vault [ExamRef-RSV](#)

[View all Jobs](#) (last 24 hours)

Last backup status Warning(Backup disabled)

Backup policy

Oldest restore point 8/17/2020, 6:09:14 AM (4 hour(s) ago)

Restore points (1)

This list is filtered for last 30 days of restore points. To recover from restore point older than 30 days, [click here.](#)

CRASH CONSISTENT | APPLICATION CONSISTENT | FILE-SYSTEM CONSISTENT
1 | 0 | 0

Time

↓ Consistency

↑ Recovery Type

8/17/2020, 6:09:14 AM

Crash Consistent

Snapshot and Vault

FIGURE 5-41 Soft Delete–enabled backup item after deletion

You can select **Undelete** anytime within 14 days of the retention period (see [Figure 5-42](#)). Once the data is restored, you can **Resume backup** again.

Undelete ExamRef-VMs

X

All restore points for this backup item will be undeleted and the item will come to 'Stop protection with retain data' state. You can 'Resume backup' to continue the scheduled backup operations as per the selected policy.

Note: Garbage Collection will start with resume backup operation and all the expired restore points will be cleaned.

Backup item

ExamRef-VMs

Deletion time

8/17/2020, 10:59:39 AM (21 second(s) ago)

Day(s) left until permanent deletion

13

Undelete

Cancel

FIGURE 5-42 Soft Delete Undelete option for ExamRef-VMs

Create and configure Backup Policy

You can edit a policy, associate more VMs to a policy, and delete unnecessary policies to meet compliance requirements.

To view your current backup policies in the Azure portal, open the **Recovery Services Vault** blade, and then click **Backup Policies** (Figure 5-43). Click an existing policy to view the policy details, or click **Add** to create a new policy.

 ExamRef-RSV | Backup policies

Recovery Services vault

Search (Ctrl+ /) « + Add

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Filter items ... All policies

Name	Policy Type	...
HourlyLogBackup	SQL Server in Azure VM	...
DefaultPolicy	Azure Virtual Machine	...

Identity Private endpoint connections Properties Locks Export template

Getting started

Backup Site Recovery

Protected items

Backup items Replicated items

Manage

Backup policies Backup Infrastructure

FIGURE 5-43 Backup policies in the Recovery Services Vault

You can create four different types of policies from this view, as shown in Figure 5-44.

- **Azure Virtual Machine.** Allows you to specify the backup frequency, retention period, and the backup point on a weekly, monthly, and yearly schedule.
- **SAP HANA In Azure VM.** Allows you to use SAP HANA specific backup technology such as full, differential, and log backup with an associated schedule for each option.
- **Azure File Share.** Allows you to schedule a daily backup for an Azure fileshare.
- **SQL Server In Azure VM.** Allows you to use SQL Server-specific backup technology, such as full, differential, and log backups, with an associated schedule for each option. Also, you can enable SQL backup compression.

Add

Policy Type

Azure Virtual Machine

SQL Server in Azure VM

SAP HANA in Azure VM

Azure File Share

FIGURE 5-44 Available backup policy options in the Azure portal

Define backup policies

An Azure Backup policy defines how often backups occur and how long the backups are retained. The default policy accomplishes a daily backup at 05:30pm UTC and retains backups for 30 days and you can define custom Backup policies. In [Figure 5-45](#), a custom Backup Policy is configured. In the **Frequency** drop-down menu, **Daily** has been chosen, though options for **Weekly**, **Monthly**, and **Yearly** are also available.

Implement backup policies

To implement a backup policy, open the policy in the Azure portal and click the

Associated Items button, as shown in Figure 5-46.

Create policy

Policy name * ⓘ

ExamRefBackupPolicy ✓

Backup schedule

Frequency *

Time *

Timezone *

Daily ✓

6:30 AM ✓

(UTC-05:00) Eastern Time (US & Canada) ✓

Instant Restore ⓘ

Retain instant recovery snapshot(s) for

2 ✓ Day(s) ⓘ

Retention range

Retention of daily backup point.

At

For

6:30 AM ✓

180 ✓ Day(s)

Retention of weekly backup point.

On *

At

For

Sunday ✓

6:30 AM ✓

Week(s)

Retention of monthly backup point.

Week Based

Day Based

On *

Day *

At

For

First ✓

Sunday ✓

6:30 AM ✓

Month(s)

Retention of yearly backup point.

Week Based

Day Based

In *

On *

Day *

At

For

January ✓

First ✓

Sunday ✓

6:30 AM ✓

Year(s)

10 ✓



Azure Backup service creates a separate resource group to store the instant recovery points of managed virtual machines. The default naming format of resource group created by Azure Backup service is AzureBackupRG_{Geo}_{n}.

[Learn More](#)

Create

FIGURE 5-45 Configuring a custom backup policy

Home > Recovery Services vaults > ExamRef-RSV | Backup policies >

ExamRefBackupPolicy

Backup policy

 Associated items  Modify  Delete

[Learn more and get FAQs about Backup policy](#)

BACKUP FREQUENCY

Daily at 6:30 AM Eastern Standard Time

RETENTION RANGE

Retention of daily backup point

Retain backup taken every day at 6:30 AM for 180 Day(s)

Retention of weekly backup point

Retain backup taken every week on Sunday at 6:30 AM for 52 Week(s)

Retention of monthly backup point

Retain backup taken every month on First Sunday at 6:30 AM for 60 Month(s)

Retention of yearly backup point

Retain backup taken every year in January on First Sunday at 6:30 AM for 10 Year(s)

FIGURE 5-46 The associated items link for the Azure policy

The **Associated Items** blade in [Figure 5-47](#) shows all the resources currently associated with the policy.

The screenshot shows the 'Associated items' blade for an Azure backup policy named 'ExamRefBackupPolicy'. The top navigation bar includes 'Home > Recovery Services vaults > ExamRef-RSV > Backup policies > ExamRefBackupPolicy >'. Below the title 'Associated items' is a breadcrumb trail 'ExamRefBackupPolicy'. A toolbar at the top right includes 'Refresh' (blue), 'Add' (green plus), and 'Filter' (blue magnifying glass). A message 'Fetching data from service completed.' with an info icon is displayed. A search bar 'Filter items...' is present. A table lists one item: 'ExamRef-VM1' under 'Name', 'ExamRef-RG' under 'Resource Group', 'Passed' under 'Backup Pre-Check' (green circle icon), 'Success' under 'Last Backup Status' (green circle icon), and '8/17/2020, 11:35:34 AM' under 'Latest restore point'. The table has columns for Name, Resource Group, Backup Pre-Check, Last Backup Status, Latest restore point, and an ellipsis column.

Name	Resource Group	Backup Pre-Check	Last Backup Status	Latest restore point	
ExamRef-VM1	ExamRef-RG	Passed	Success	8/17/2020, 11:35:34 AM	...

FIGURE 5-47 The associated items for the backup policy

Clicking **Add** will launch the **Backup Goal** blade, where you can add other virtual machines or file shares to be backed up using the goals defined in the policy.

Perform backup and restore operations by using Azure Backup Service

Azure Backup service can be used to back up and restore various cloud as well as on-prem resources. Recovery Services Vault is used to enable Azure Backup and to configure the backup policies.

Backup and restore with Azure workloads

For Azure workloads, the Azure Backup service can back up the following resources:

- Virtual machine/s
- SAP HANA databases running in an Azure VM
- Azure file share
- SQL Server databases running in an Azure VM

When you back up an Azure virtual machine, you can restore an entire virtual machine or you can restore individual files from the virtual machine and it is quite easy to set up. To back up a VM in Azure with Azure Backup, open the Recovery Services Vault and click **Backup** under **Getting Started**. From the **Where Is Your Workload Running?** drop-down menu, select **Azure**, and from the **What Do You Want To Backup?** drop-down menu, select **Vrtual Machine**. After making these selections, click **Backup**, as shown in [Figure 5-48](#).



ExamRef-RSV | Backup

Recovery Services vault

Search (Ctrl+ /)

«



The storage replication is set to Geo-redundant. This option cannot be changed later. Before proceeding further, click here. →



Overview



Activity log



Access control (IAM)



Tags



Diagnose and solve problems



Settings



Identity



Private endpoint connections



Properties



Locks



Export template

Getting started



Backup



Site Recovery

Where is your workload running?

Azure

▼

What do you want to backup?

Virtual machine

▼

Step: Configure Backup

Backup

FIGURE 5-48 Configuring Azure Backup to protect virtual machines

Next, choose a backup policy from the **Policy** drop-down menu, or click the **Create New Policy** link to create your own policy. Next, choose the VMs to back up by clicking the **Add** button on the lower left and then select the virtual machines from the **Select Virtual Machines** list as shown in [Figure 5-49](#). Only VMs located in the same region as the Recovery Services Vault are available for backup.

Home > Recovery Services vaults > ExamRef-RSV | Backup >

Backup

ExamRef-RSV

Policy	DefaultPolicy	V
Create a new policy		
BACKUP FREQUENCY		
Daily at 5:30 PM UTC		
Instant Restore		
Retain instant recovery snapshot(s) for 2 day(s)		
RETENTION RANGE		
Retention of daily backup point		
Retain backup taken every day at 5:30 PM for 30 Day(s)		
Virtual Machines		
Virtual machine name	Resource Group	OS Disk Only
No Virtual Machines Selected		
Add <p>! OS Disk only backup option allows you to backup Azure Virtual Machine with only OS disk and exclude all the data disks. You can use Selective Disk Backup feature through PowerShell limitation and pricing- Learn more.</p>		
Enable Backup		OK

Select virtual machines

X

Filter items ...

	Virtual machine name	Resource Group
<input checked="" type="checkbox"/>	ExamRef-VM	ExamRef-RG
<input checked="" type="checkbox"/>	ExamRef-VMs	ExamRef-RG

FIGURE 5-49 Configuring Azure Backup to backup virtual machines and using the default policy

After the VMs are selected, click the **Enable Backup** button.

Note Azure VM Protection and Vault Storage Redundancy Type

When protecting IaaS VMs by using Azure Backup, only VMs in the same region as the Recovery Services Vault are available for backup. Because of this, it is a best practice to choose Geo-Redundant storage or Read Access Geo-Redundant storage to be associated with the Recovery Services Vault. This ensures that if a regional outage affects VM access, there is a replicated copy of the backup in another region.

When you click the **Enable Backup** button, behind the scenes, the `vMSnapshot` (for Windows) or `vMSnapshotLinux` (for Linux) extension is automatically deployed by the Azure fabric controller to the VMs. This allows for snapshot-based backups to occur, which means a snapshot of the VM is taken first, and then this snapshot is streamed to the Azure Storage associated with the Recovery Services Vault. The initial backup is not taken until the day/time configured in the backup policy, though an ad-hoc backup can be initiated at any time. To do so, navigate to the **Protected Items** section of the Recovery Services Vault properties, click **Backup Items**, and click **Azure Virtual Machine** under **Backup Management Type**. The VMs that are enabled for backup are listed here. To begin an ad-hoc backup, right-click a VM and select **Backup Now**, as shown in [Figure 5-50](#).

Backup Items (Azure Virtual Machine)

ExamRef-RSV

⟳ Refresh + Add ⚡ Filter



Fetching data from service completed.

🔍 Filter items ...

Name

↑↓ Resource Group

↑↓ Backup Pre-Check

ExamRef-VM

ExamRef-RG

✓ Passed

ExamRef-VMs

✓ Passed

Pin to dashboard



Backup now

Restore VM

File Recovery

Stop backup

Delete backup data

Undelete

FIGURE 5-50 Starting an ad-hock backup

Azure Backup also directly supports the ability to back up and restore data from Azure Files, SQL Server databases, and SAP HANA databases on Azure virtual machines. It is a good idea to have a basic understanding of the capabilities because they might appear on the exam.

More Info Azure Files and SQL Server in an Azure VM

Learn about the current capabilities of Azure Backup support for Azure

Files here: <https://docs.microsoft.com/azure/backup/backup-azure-files>,

SQL Server on Azure VM here:

<https://docs.microsoft.com/azure/backup/backup-azure-sql-database> and

SAP HANA on Azure VM here:

<https://docs.microsoft.com/azure/backup/sap-hana-db-about>.

After backing up a virtual machine using Azure Backup there are two methods to restore data: Restore VM and File Recovery.

To restore a recovery point as a new virtual machine, open the Recovery Services vault, and click on **Backup Items**, then click **Azure Virtual Machine**, and then click the virtual machine you want to restore from the list. The next screen will list all the restore points available for restoration, as shown in [Figure 5-51](#).

ExamRef-VM



Backup Item



Backup now



Restore VM



File Recovery



Stop backup



Resume backup



Delete backup data



Restore to Secondary Region



Undelete

Alerts and Jobs

[View all Alerts](#) (last 24 hours)

Backup status

Backup Pre-Check

Passed

Summary

Recovery services vault [ExamRef-RSV](#)

[View all Jobs](#) (last 24 hours)

Last backup status

Success 8/17/2020, 6:08:51 AM

Backup policy

[DefaultPolicy](#)

Oldest restore point 8/17/2020, 6:08:55 AM (2 hour(s) ago)

Restore points (1)

This list is filtered for last 30 days of restore points. To recover from restore point older than 30 days, [click here](#).



Time

↑↓ Consistency

↑↓ Recovery Type

8/17/2020, 6:08:55 AM

Crash Consistent

Snapshot and Vault

FIGURE 5-51 Available restore points for a virtual machine.

Right-click the desired restore point and select **Restore VM** (see Figure 5-52), or click the **Restore VM** link at the top of the page (see Figure 5-53).



FIGURE 5-52 Restore VM option 1

Home > Recovery Services vaults > ExamRef-RSV | Backup items > Backup Items (Azure Virtual Machine) > ExamRef-VM

Restore Virtual Machine

examref-vm

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point* [Select](#)

Select restore point

x

examref-vm

Start Date	End Date	Recovery point type
08/03/2020	08/17/2020	All restore points

CRASH CONSISTENT APPLICATION CONSISTENT FILE-SYSTEM CONSISTENT

Time	Consistency ↑	Recovery Type ↑↓
8/17/2020, 6:08:55 AM	Crash Consistent	Snapshot and Vault

[Restore](#) [OK](#) [Cancel](#)

FIGURE 5-53 Restore VM option 2

From there, you can then restore to a new virtual machine by selecting **Create New**, or you can restore over an existing virtual machine by selecting **Replace Existing**.

Figure 5-54 shows the **Restore Virtual Machine** blade with the **Create New** option selected. Here, you can specify the virtual machine name, resource group, virtual network, subnet, and storage account.

Restore Virtual Machine

examref-vm

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point *

8/17/2020, 6:08:55 AM

Select

Restore Configuration

Create new Replace existing

i To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type * ⓘ

Create new virtual machine

Virtual machine name * ⓘ

ExamRef-VMNew



Resource group * ⓘ

ExamRef-RG



Virtual network * ⓘ

ExamRef-RG-vnet (ExamRef-RG)



Subnet * ⓘ

default



Staging Location * ⓘ

examrefrgdiag690 (StandardLRS)



[Can't find your storage account ?](#)

Restore

FIGURE 5-54 Restore to a new virtual machine

If you just need access to files from the virtual machine, choose the **File Recovery** option at the top of the page shown previously in [Figure 5-51](#) instead. From there, you can select the recovery point and then download a script that will mount the selected recovery point to another computer as local disks (see [Figure 5-55](#)). The disks will remain mounted for 12 hours so you can recover the needed data.

File Recovery

examref-vm

✓ Step 1: Select recovery point

8/17/2020, 6:08:55 AM [Latest] (Cras... ▾)

✓ Step 2: Download script to browse and recover files

This script will mount the disks from the selected recovery point **as local drives on the machine where it is run**. These drives will remain mounted for 12 hours.

[Download Script *](#)

✓ Password generated successfully.

Password to run the script

bc734769bce3a5b 

→ Step 3: Unmount the disks after recovery

Unmount disks and close the connection to the recovery point.

[Unmount Disks](#)

* Run this script on the machine where you want to copy the files

* To restore files larger than 10GB, [restore entire VM](#) to an alternate location or [restore disks using PowerShell](#)

* Data transfer rate: up to 1GB/Hr

If you have trouble finding your files,
[click here](#)

FIGURE 5-55 Restore to a new virtual machine



Exam Tip

To restore a virtual machine that has encrypted disks, you also need to provide the Azure Backup Service access to the Key Vault holding the keys. See <https://docs.microsoft.com/azure/backup/backup-azure-vms-encryption>.

More Info More details about restoring virtual machines and files

You can learn more about recovering virtual machines with the Azure Backup service at <https://docs.microsoft.com/azure/backup/backup-azure-arm-restore-vms>. For more information about file-level recovery, see <https://docs.microsoft.com/azure/backup/backup-azure-restore-files-from-vm>.

Backup and restore with on-premises workloads

Azure Backup Server is a standalone service that you install on a Windows Server operating system that stores the backed-up data in a Recovery Services Vault. Azure Backup Server inherits much of the workload backup functionality from Data Protection Manager (DPM). Though Azure Backup Server shares much of the same functionality as DPM, Azure Backup Server does not back up to tape, and it does not integrate with System Center.

You should consider using Azure Backup server when you need to back up the following supported workloads:

- Windows Client
- Windows Server

- Linux Servers (running on Hyper-V or VMWare)
- VMWare VMs
- Exchange
- SharePoint
- SQL Server
- System State and Bare Metal Recovery

More Info Azure Backup Server Protection Matrix

The entire list of supported workloads and the versions supported for Azure Backup Server can be found at

<https://docs.microsoft.com/azure/backup/backup-mabs-protection-matrix>.

Follow these steps to install Azure Backup Manager:

1. To see the instructions to prepare the infrastructure, navigate to the Recovery Services Vault and under **Getting Started**, click **Backup**.
2. From the **Where Is Your Workload Running?** drop-down menu, select **On-Premises** as the location where the workload is running.
3. From the **What Do You Want To Backup?** drop-down menu, choose **Hyper-V Virtual Machine** as the workload to backup.
4. Finally, click **Prepare Infrastructure**, as shown in [Figure 5-56](#).



ExamRef-RSV | Backup

Recovery Services vault

Search (Ctrl+ /)

«

Where is your workload running?

On-Premises

▼

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Identity

Private endpoint connections

Properties

Locks

Export template

Getting started

Backup

Site Recovery

What do you want to backup?

Hyper-V Virtual Machines

▼

Step: Prepare Infrastructure

Prepare Infrastructure...

FIGURE 5-56 Configuring Azure Backup to protect on-premises virtual machines

5. If you are not using any System Center environment, it is recommended that you download and install Microsoft Azure Backup Server. On the next screen, deselect the **Already Using System Center Data Protection Manager Or Any Other System Center Product** box. You will see instructions for downloading and installing Microsoft Azure Backup Server, as shown in [Figure 5-57](#). You might need to download the vault credentials to register the server to the Recovery Services Vault, which has expiry date of 2 days. After setting up the Azure Backup Server, use its UI to configure the on-premises backup.

Prepare infrastructure

X

- Already using System Center Data Protection Manager or any other System Center Product

Azure Backup Server

Please follow the steps mentioned below.

1. Install Microsoft Azure Backup Server

[Download](#)

2. Download vault credentials to register the server to the vault. Vault credentials will expire after 2 days.

- Already downloaded or using the latest Azure Backup Server installation



If you are downloading vault credentials for recovering data from an alternate server, ensure that your Azure Backup Server installation has the latest version of the Azure Recovery Services Agent. Click here to go to the latest release of the Azure Recovery Services Agent.



[Download](#)

3. Post infrastructure preparation, please use Microsoft Azure Backup Server UI(on-premises) to configure backup.

[Learn More](#)

FIGURE 5-57 Instructions to prepare infrastructure for on-premises backup using Backup server

6. Alternatively, if you are using any System Center environment, check the **Already Using System Center Data Protection Manager Or Any Other System Center Product** box. You will these a different set of instructions, which tell you how to download and install System Center DPM (Data Protection Manager), as shown in [Figure 5-58](#). You might need to download the vault credentials to register the server to the Recovery Services Vault, which has expiry date of two days. After setting up the DPM, use its UI to configure the on-premises backup.

Prepare infrastructure

- Already using [System Center Data Protection Manager](#) or any other [System Center](#) Product

SC Data Protection Manager

Please follow the steps mentioned below.

1. Download and install System Center DPM available with your System Center Subscription.
2. Install Microsoft Azure Recovery Services Agent.

[Download](#)

3. Download vault credentials to register the server to the vault. Vault credentials will expire after 2 days.

- Already downloaded or using the [latest Recovery Services Agent](#)

[Download](#)

4. Post infrastructure preparation, please use DPM UI (on-premises) to configure backup.

[Learn More](#)

FIGURE 5-58 Instructions to prepare infrastructure for on-premises backup using DPM

To back up files and folders from on-premises VMs, you need to use Microsoft Azure Recovery Services (MARS) agent. The MARS agent is available for installation from the Recovery Services Vault.

1. Click **Backup** under **Getting Started**.
2. From the **Where Is Your Workload Running?** drop-down menu, select **On-Premises**, and from the **What Do You Want To Backup?**, choose **Files And Folders**.
3. Next, click **Prepare Infrastructure**, and the Recovery Services agent is made available, as shown in [Figure 5-59](#). You can also download the MARS agent by visiting the MARS vault's Properties and choosing **Backup -> Download Recovery Services Agent**.
4. Notice there is only a Windows agent because the backing up of files and folders is only supported on Windows computers. Click the link to download the agent. Before initiating the installation of the MARS agent, you should also download the vault credentials file, which is right under Recovery Services agent download link. The vault credentials file is needed during the installation of the MARS agent.

Note Vault Credentials Expiration

The vault credentials are only valid for 48 hours from the time of download, so be sure to obtain them only when you are ready to install the MARS agent.

Prepare infrastructure

Recovery Services Agent

Please follow the steps mentioned below.

1. Install Recovery Services agent

[Download Agent for Windows Server or Windows Client](#)

2. Download vault credentials to register the server to the vault. Vault credentials will expire after 2 days.

Already downloaded or using the latest Recovery Services Agent

[Download](#)

3. Schedule backup using Recovery Services Agent UI. [Learn More](#)

4. Once the backups are scheduled, you can use backup jobs page to monitor the backups. [Browse jobs page](#)

5. You can also Configure Notifications from alerts page to receive email alerts for backup failures. [Browse alerts page](#)

[Learn More](#)

FIGURE 5-59 Downloading the MARS agent

5. During the MARS agent installation, a cache location must be specified. There must be free disk space within this cache location that is equal to or greater than five percent of the total amount of data to be protected. These configuration options are shown in [Figure 5-60](#).

Microsoft Azure Recovery Services Agent Setup Wizard

X



Installation Settings

Installation Stages

Installation Settings

Proxy Configuration

Installation

Installation Folder

Microsoft Azure Recovery Services Agent will be installed in the following folder. To choose a different installation folder, click Browse. The location specified must have at least 1 GB of free space.

C:\Program Files\Microsoft Azure Recovery Services Agent

Browse

Cache Location

Microsoft Azure Recovery Services Agent can use this to keep track of files being backed up from your computer. The location specified must have free space which is atleast 5% of the backup data.

C:\Program Files\Microsoft Azure Recovery Services Agent\Scratch

Browse

< Back

Next >

Cancel

FIGURE 5-60 Installing the MARS agent

6. The agent needs to communicate to the Azure Backup service on the Internet, so on the **Proxy Configuration** screen, configure any required proxy settings.
7. On the **Installation**, required Windows features are added to the system where the agent is being installed. After these features have been added, you can click **Install** as shown in Figure 5-61.

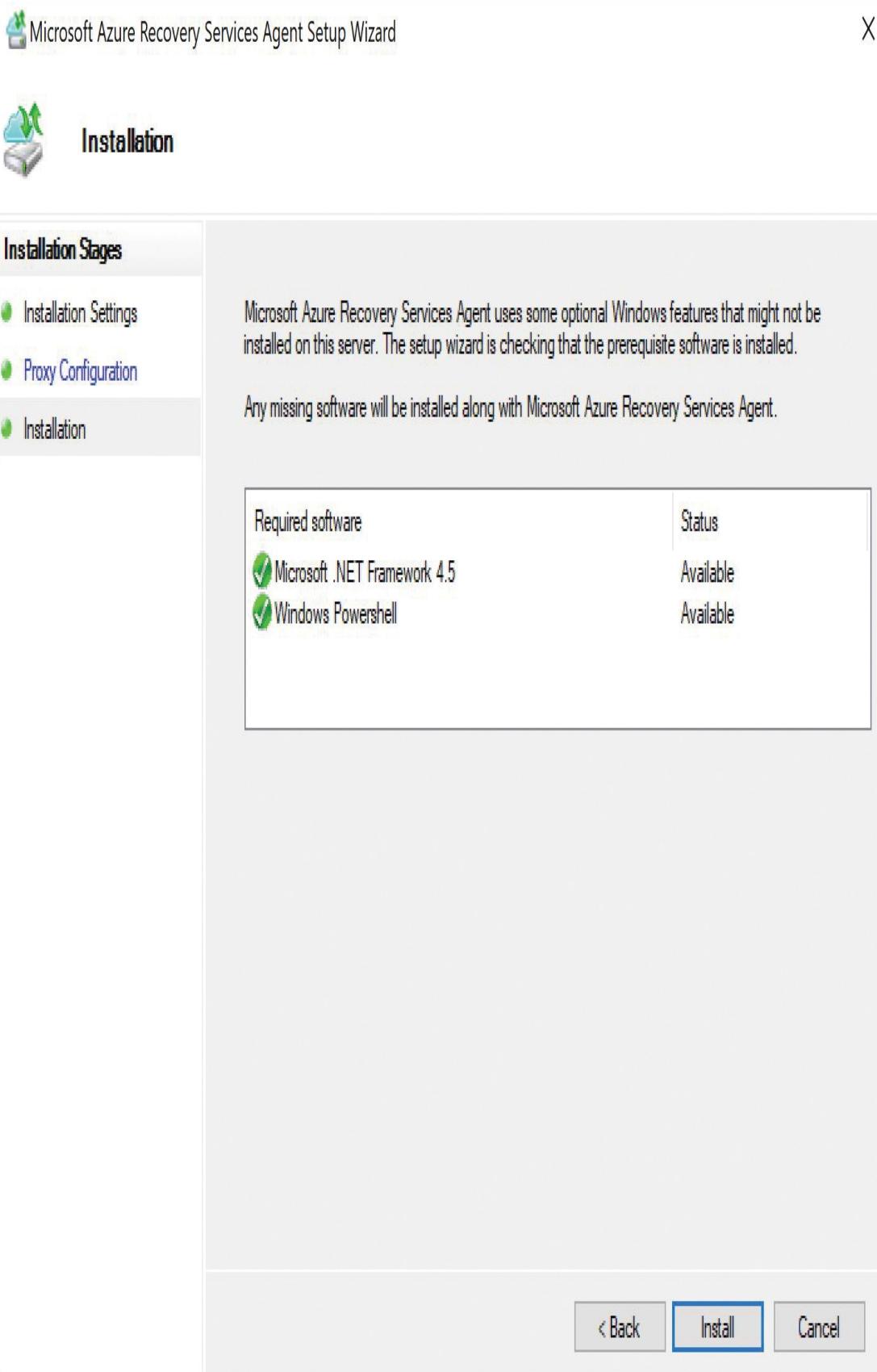


FIGURE 5-61 Final screen of the MARS agent installation

8. Click **Proceed To Registration** to open the **Agent Registration** dialog box. The vault credentials must be provided by browsing to the path of the downloaded file.
9. On the **Encryption Settings** page, it is very important that you either specify a passphrase or allow the installation program to generate one. Enter this passphrase twice, and then specify where the passphrase file should be saved. The passphrase file is a text file that contains the passphrase, so store this file securely.

Note Azure Backup Encryption Passphrase

Data protected by Azure Backup is encrypted using the supplied passphrase. If the passphrase is lost or forgotten, any data protected by Azure Backup is not able to be recovered and is lost.

10. After the agent is registered with the Azure Backup service, it can then be configured to begin protecting data.
11. In the last section, the MARS agent was installed and registered with the Azure Backup vault. Before data can be protected with the agent, it must be configured with settings, such as when the backups occur, how often they occur, how long the data is retained, and what data is protected. In the MARS agent interface, click **Schedule Backup** to begin this configuration process.
12. Click **Next** to move past the **Getting Started** screen and click **Add Items** to add files and folders. Clicking **Exclusion Settings** allows you to choose certain file types that are not protected, as shown in [Figure 5-62](#).

Schedule Backup Wizard

Select Items to Backup

Getting started

Select Items to Backup

Specify Backup Schedule ...

Select Retention Policy (F...)

Choose Initial Backup Typ...

Confirmation

Modify Backup Progress

Click Add Items to select the files and folders you want to backup.

Name

C:\Data\

Add Items Remove Items

Exclusion Settings

! Use "Exclusion Settings" for removing all items within a volume. Refer to <https://aka.ms/MarsUnselectVolume> to learn about the implications of not using "Exclusion Settings" for removing all items in a volume.

i If a backup location includes file types or sub-folders that you do not want to back up, click Exclusion Settings to remove those items from the backup.

< Previous Next > Finish Cancel

FIGURE 5-62 Configuring the MARS agent to protect data

13. Next, schedule how often backups should occur. The agent can be configured to back up daily or weekly, with a maximum of three backups taken per day.
14. Specify the retention you want, and the initial **Backup Type (Over The Network or Offline)**. Confirm the settings to complete the wizard. Backups are now scheduled to occur, but they can also be initiated at any time by clicking **Back Up Now** on the home screen of the agent. [Figure 5-63](#) shows an active backup.

Schedule Backup Wizard



Modify Backup Progress

Getting started

Select Items to Backup

Specify Backup Schedule ...

Select Retention Policy (F...

Choose Initial Backup Typ...

Confirmation

Modify Backup Progress

Status:

Creating Backup Schedule...



Action

Create backup schedule for Files and Folders

Status

In Progress

< Previous Next > Close Cancel

FIGURE 5-63 Creating Backup Schedule

Follow these steps to recover data:

1. Click the **Recover Data** option on the home screen of the MARS agent. This initiates the Recover Data Wizard.
2. Choose the computer to which you want to restore the data. Generally, this is the same computer the data was backed up from.
3. Next, choose the data to recover, the date on which the backup took place, and the time the backup occurred. These choices comprise the recovery point to restore.
4. Click **Mount** to mount the selected recovery point as a volume, and then choose the location to recover the data.
5. Confirming the selected options starts the recovery.

Perform site-to-site recovery by using Azure Site Recovery

Every organization will have its own business continuity and disaster recovery (BCDR) plans to handle the unpredictable circumstances with unexpected outages that occur. Azure Site Recovery service enables us to replicate, failover, and fallback virtual machines as needed. Azure Site Recovery solution allows us to address below major scenarios:

- Azure VMs from one region to another
- On-Premises VMs (VMware, Hyper-V, and physical servers) to Azure
- On-Premises VMs to another site

Let's take Azure VMs to replicate from one region to another as an example. Firstly, you would need to create a Recovery Services Vault. We will leverage the one created in earlier section named **ExamRef-RSV**. As a best practice, you should always validate the target subscription readiness by checking the appropriate VM SKU and major feature availability.

For the enterprise environments, you should also consider allow-listing the URLs for outbound connectivity to required Azure resources and service tag—

based NSG rules. You would also need minimum Site Recovery Contributor rights for configuring the replication and Site Recovery Operator rights for executing the failover and failback operations.

To enable replication from source VM, follow these steps:

1. Open the Recovery Services Vault and select + **Replicate**.
2. On the Source page, you need to provide the source details, such as the location, deployment model, subscription, source resource group, as shown in [Figure 5-64](#).

Enable replication

ExamRef-RSV

«

Source

ExamRef-RSV

1 Source

Configure

»

i Select your source environment

Source

Azure

2 Virtual machines

Select

»

Source location * ⓘ

East US

3 Replication settings

Configure replication settings

»

Azure virtual machine deployment model * ⓘ

Resource Manager

Source subscription *

Visual Studio Ultimate with MSDN

Source resource group *

ExamRef-RGSR

Disaster Recovery between Availability Zones? * ⓘ

No

Availability Zones ⓘ

Select

FIGURE 5-64 Enable replication – Source configuration

3. On the next blade, select the Source VM for replication (see [Figure 5-65](#)).

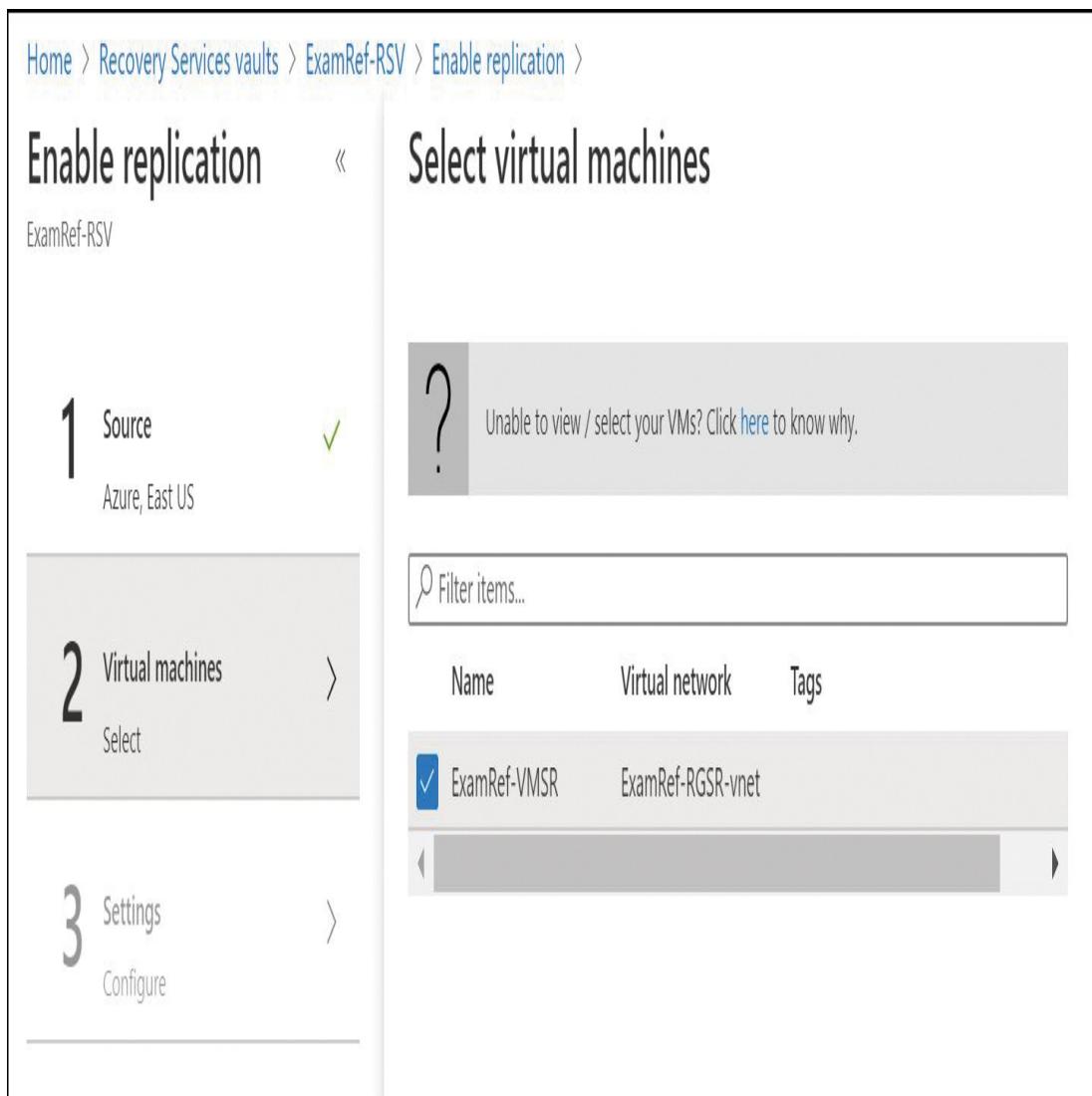


FIGURE 5-65 Enable replication – Source VM selection

4. Next, you must configure target environment settings, as shown in [Figure 5-66](#), and then click **Create Target Resources**.

Enable replication

ExamRef-RSV

- 1** Source ✓
Azure, East US
- 2** Virtual machines ✓
1 Selected
- 3** Settings >
Configure

Configure settings

Target location * ⓘ

Target subscription ⓘ Customize

Visual Studio Ultimate with MSDN



If you are choosing General Purpose v2 storage accounts, ensure that operations and data transfer prices are understood clearly before you proceed. [Learn more](#)

Resource group, Network, Storage and Availability ⓘ Customize

By default Site Recovery will mirror the source site configuration to target site by creating/using the required resource groups, storage accounts, virtual network and availability sets as below. Click 'Customize' above to change the configuration. The resources created are appended with "asr" suffix.

Target resource group ⓘ

ExamRef-RG

Target virtual network ⓘ

ExamRef-RG-vnet

Cache storage accounts ⓘ

(new) m1rtnexamrefrsasrcache

Replica managed disks ⓘ

(new) 1 premium disk(s), 0 standard disk(s)

Target availability sets ⓘ

Not Applicable

Replication Policy ⓘ Customize

Name: 24-hour-retention-policy**Recovery point retention:** 24 hour(s)**App consistent snapshot frequency:** 4 hour(s)**Replication group:** None

Extension settings [-] Hide details

Update settings

ⓘ

Automation account

ⓘ

Enable replication

Create target resources

FIGURE 5-66 Enable replication – Target settings

- Finally, click **Enable Replication** (see Figure 5-67).

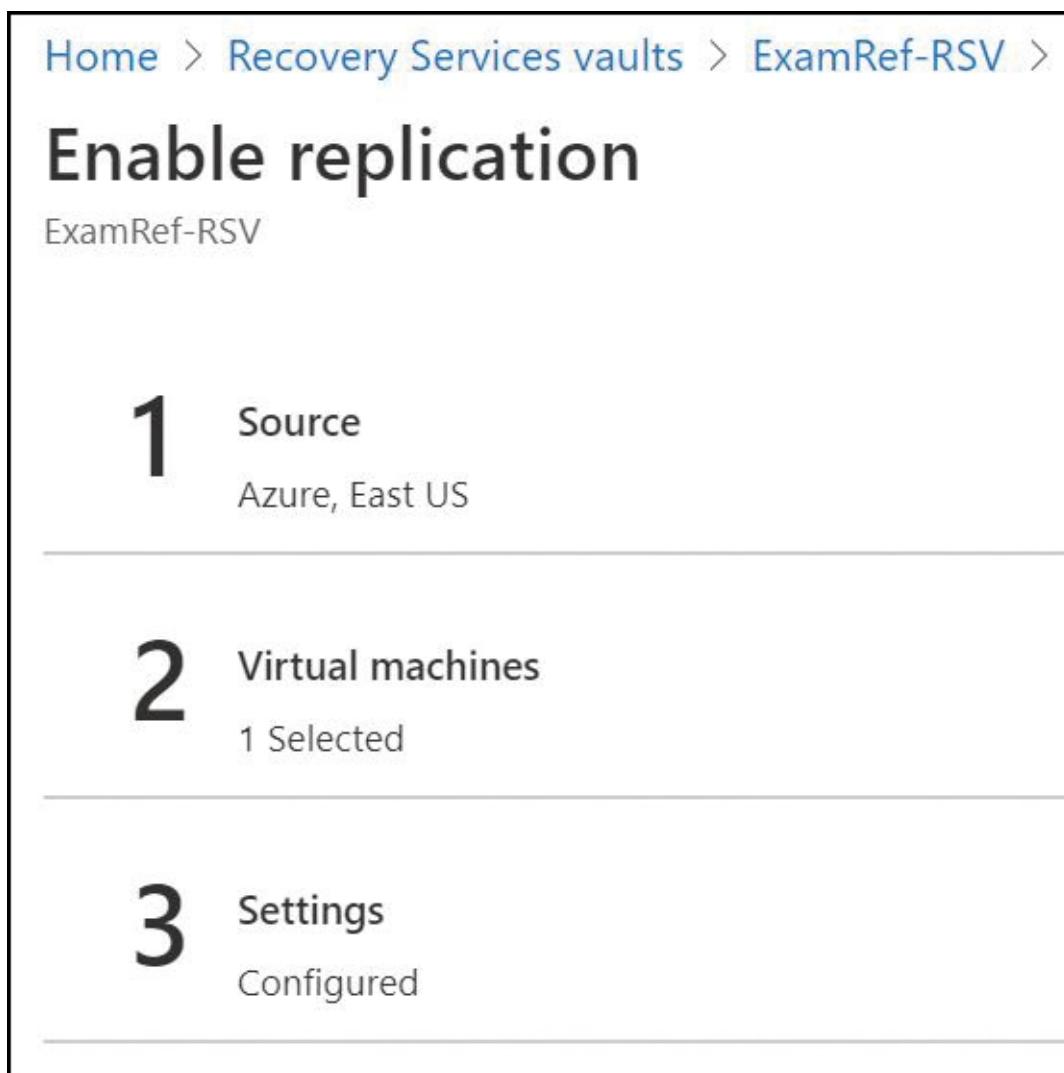


FIGURE 5-67 Enable replication—Target settings

You can track the replication progress by selecting Site Recovery Jobs (see Figure 5-68). It takes a while to get the replication and synchronization completed. You cannot proceed with further steps without replication of the VM.

[Home >](#)

Site Recovery jobs

X

[ExamRef-RSV](#)[Filter](#) [Export jobs](#) Filter items...

Name	Status	Type	Item	Start time	Duration
Protection configuration	Successful	Cloud	asr-a2a-default-eastus-container	8/17/2020, 4:14:02 PM	00:00:02
Protection configuration	Successful	Cloud	asr-a2a-default-canadacentral-container	8/17/2020, 4:14:01 PM	00:00:01
Enable replication	Successful	Protected item	examref-vmsr	8/17/2020, 4:13:38 PM	00:11:38
Associate replication policy	Successful	Replication policy	24-hour-retention-policy	8/17/2020, 4:11:47 PM	00:01:11
Associate replication policy	Successful	Replication policy	24-hour-retention-policy	8/17/2020, 4:10:26 PM	00:01:05
Map Networks	Successful	Network	examref-rgs-vnet	8/17/2020, 4:10:25 PM	00:00:02
Create protection container	Successful	Cloud	asr-a2a-default-canadacentral-container	8/17/2020, 4:09:54 PM	00:00:00
Map Networks	Successful	Network	examref-rgs-vnet	8/17/2020, 4:09:54 PM	00:00:01
Create protection container	Successful	Cloud	asr-a2a-default-eastus-container	8/17/2020, 4:08:52 PM	00:00:00
Create a site	Successful	Server	asr-a2a-default-canadacentral	8/17/2020, 4:08:47 PM	00:00:05
Create a site	Successful	Server	asr-a2a-default-eastus	8/17/2020, 4:07:25 PM	00:00:04
Create replication policy	Successful	Replication policy	24-hour-retention-policy	8/17/2020, 4:07:25 PM	00:00:00

FIGURE 5-68 Site Recovery jobs

Once replicated, now you can see the Source VM listed in the Recovery Services Vault under **Replicated Items**. The overview is shown in [Figure 5-69](#).

ExamRef-VMR

Replicated items

[Search \(Ctrl+ /\)](#) « Failover Test Failover ✓ Cleanup test failover Commit Resynchronize ⚡ Change recovery point Re-protect ⚡ Disable Replication Error Details ⚡ Refresh

[Overview](#) Essentials

General

Health and status		Failover readiness	
Replication Health	Healthy	Last successful Test Failover	Never performed successfully
Status	Protected	Configuration issues	No issues
RPO	1 min [As on 8/17/2020, 5:43:08 PM]	Agent version	9.36.5690.1
		Agent status	Healthy

Latest recovery points
 Click above to see the latest recovery points.

Errors(0) [Open in new page](#) **Events - Last 72 hours(0)** [Open in new page](#)

No errors No events

[Infrastructure view](#) [Table view](#)

```

graph LR
    VM[Virtual machine ExamRef-VMR] --- ASR[Azure Site Recovery]
    ASR --- Cache[Cache storage account(s)]
    ASR --- Disk[Managed disk(s) 1]
    Cache --- Disk
  
```

FIGURE 5-69 Replicated Items—Source VM

Now it's time to do test failover:

1. Right-click the VM under **Replicated Items** and click **Test Failover** (see [Figure 5-70](#)).

Home >

ExamRef-RSV | Replicated items

Recovery Services vault

Search (Ctrl+F) Refresh + Replicate Columns Filter

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Last refreshed at: 8/17/2020, 5:03:32 PM

0 Finished loading data from service.

Filter items...

Name	Replication Health	Status	Active location
ExamRef-VMSP	Healthy	Protected	East US

Pin to dashboard

Failover
Test Failover
Cleanup test failover
Change recovery point
Commit
Re-protect
Resynchronize
Error Details
Disable Replication

Identity Private endpoint connections Properties Locks Export template

Getting started

Backup Site Recovery

Protected items

Backup items Replicated items

The screenshot shows the 'Replicated items' blade in the Azure Recovery Services vault. The main table displays a single replicated item named 'ExamRef-VMSP'. The item is in a healthy state, marked as protected, and its active location is listed as 'East US'. A context menu is open over this item, providing options for failover, cleanup, commit, re-protection, synchronization, and error details. The left sidebar features a navigation menu with sections like Overview, Activity log, Access control (IAM), Tags, and Diagnose and solve problems. The 'Replicated items' section is currently selected. The top of the blade includes standard navigation controls such as search, refresh, and filter buttons.

FIGURE 5-70 Replicated Items—Test Failover

2. On the **Test Failover** blade, select a recovery point from the **Choose A Recovery Point** drop-down menu and choose a virtual network from the **Azure Virtual Network** drop-down menu, as shown in [Figure 5-71](#) and click **OK**.

Test failover

ExamRef-VMSR

Failover direction

From ⓘ

East US

To ⓘ

Canada Central

Recovery Point

Choose a recovery point ⓘ

Latest processed (low RTO) (1 out of 1 ... ▾)

Azure virtual network * ⓘ

ExamRef-RG-vnet ▾



It is recommended that for a test failover you use a network different from production network (as specified under Compute and Network settings of the virtual machine). [Learn more](#).

FIGURE 5-71 Test failover blade

3. You can track the progress of the test failover by using Site Recovery jobs, as shown in [Figure 5-72](#). Now, you will be able to see the test VM created in the target resource group.

Home >

Test failover

X

Site Recovery job

Export job Environment Details

Properties

Vault examref-svr

Protected item ExamRef-VMSR

Job id e65bb0a5-1a4f-4d15-99aa-92a312b94f2020-08-17T21:54:17Z/bz4chmlyid0cdk227-7fb3-4459-8283-d5d62cf627

Job

Name	Status	Start time	Duration
Prerequisites check for test failover	Successful	8/17/2020, 5:54:24 PM	00:00:00
Create test virtual machine	Successful	8/17/2020, 5:54:24 PM	00:01:32
Preparing the virtual machine	Successful	8/17/2020, 5:55:57 PM	00:00:35
Start the virtual machine	Successful	8/17/2020, 5:56:32 PM	00:00:00

FIGURE 5-72 Test failover –Jobs

4. You can delete the test VM after verifying the VM and network details. To delete the VM and other resources, select **Cleanup Test Failover**, as shown in [Figure 5-73](#).

Home > Resource groups > ExamRef-RG >

ExamRef-RSV | Replicated items

Recovery Services vault

Search (Ctrl+I) Refresh + Replicate Columns Filter

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Last refreshed at: 8/17/2020, 5:59:55 PM

Finished loading data from service.

Filter items...

Name	Replication Health	Status	Active location
ExamRef-VMSP	Healthy	Cleanup test failover pending	East US

Pin to dashboard ↗

Fallover

Test Fallover

Cleanup test failover

Change recovery point

Commit

Re-protect

Resynchronize

Error Details

Disable Replication

FIGURE 5-73 Cleanup test failover

5. You can now run an actual failover. Select **Failover** from the options, as shown in [Figure 5-74](#).

Home >

ExamRef-RSV | Replicated items

Recovery Services vault

Search (Ctrl+I) Refresh + Replicate Columns Filter

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Last refreshed at: 8/17/2020, 6:04:03 PM

Finished loading data from service.

Filter items...

Name	Replication Health	Status	Active location
ExamRef-VMSR	Healthy	Protected	East US

Pin to dashboard

Failover

Test Failover

Cleanup test failover

Change recovery point

Commit

Re-protect

Resynchronize

Error Details

Disable Replication

FIGURE 5-74 Failover option

6. On the **Failover** blade, select the recovery point and verify the **Failover Direction**, as shown in [Figure 5-75](#). Click **OK**.

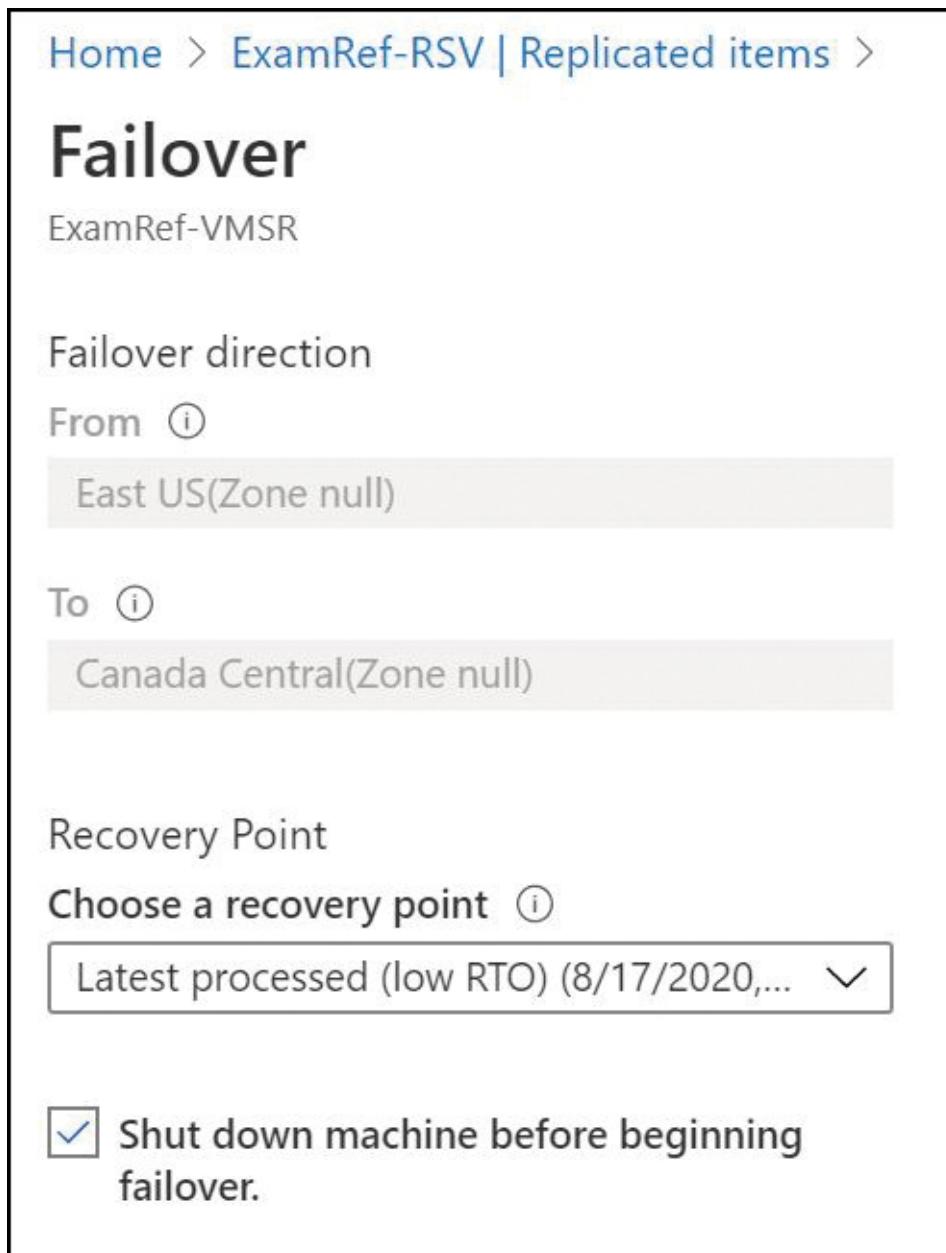


FIGURE 5-75 Failover blade

7. You can track the failover progress by following the site recovery jobs (see [Figure 5-76](#)). You must see a target Azure VM will be created with the same configuration and target settings provided earlier.
8. You should validate by logging into the VM.
9. Select **Commit** to complete the failover process.

Home >

Failover

Site Recovery Job

Export job Environment Details

X

Properties

Vault examref-rsi

Protected item ExamRef-VMR

Job id eab50a5-1af4-4f15-99aa-92a5121d85-2020-08-17T22:09:05Z-02 Activity/d6430d4f-f228-47b1-9b65-e2d5199b02e

Job

Name	Status	Start time	Duration
Prerequisites check for failover	Successful	8/17/2020, 6:09:10 PM	00:00:00
Shut down the virtual machine	Successful	8/17/2020, 6:09:11 PM	00:01:01
Synchronizing the latest changes	Skipped		00:00:00
Start failover	Successful	8/17/2020, 6:10:13 PM	00:00:11
Start the replica virtual machine	Successful	8/17/2020, 6:12:25 PM	00:00:00

FIGURE 5-76 Failover – Jobs

10. You should also consider protecting your VM again by clicking **Re-Protect**, which will reverse the process (see [Figure 5-77](#)).

More Info Site Recovery scenarios

Learn about the VMware site recovery to Azure at

<https://docs.microsoft.com/azure/site-recovery/tutorial-prepare-azure>.

You can learn about Hyper-V VM site recovery to Azure at

<https://docs.microsoft.com/azure/site-recovery/tutorial-prepare-azure-for-hyperv>.

11. Once the VM is protected again, you can perform a failback to get to the original state. Similarly, you can use Site Recovery for other scenarios as well.

Re-protect

canadacentral to eastus 



If you are choosing General Purpose v2 storage accounts, ensure that operations and data transfer prices are understood clearly before you proceed. [Learn more](#)

Resource group, Network, Storage and Availability  [Customize](#)

By default, Site Recovery will pick the original source resource group, virtual network, storage accounts and availability sets as below. Click 'Customize' above to change the configuration.
The resources created are appended with "asr" suffix.

Target resource group

ExamRef-RGSR

Target virtual network

ExamRef-RGSR-vnet

Cache storage accounts

(new) h2uxl0examrefrsasrcache

Replica managed disks

(new) 1 premium disk(s), 0 standard disk(s)

Target availability sets

Not Applicable

FIGURE 5-77 Re-protect option

More Info Azure Migrate

If you want to migrate an on-premises workload to Azure, see <https://docs.microsoft.com/azure/migrate/migrate-services-overview>.

Configure and review backup reports

Azure Backup Reports provide data visualization across your Recovery Services Vaults and Azure subscriptions to provide insight into your backup activity. This reporting solution is currently widely supported for Azure virtual machine backup and file and folder backup scenarios when using the MARS (Microsoft Azure Recovery Services) agent. For other supported scenarios, see <https://docs.microsoft.com/azure/backup/backup-azure-configure-reports#supported-scenarios>.

In order to configure the backup reports, you need to create or use an existing Log Analytics Workspace to store the backup reporting data. Also, you need a Recovery Services Vault, which records all the backup operations as diagnostic data. Creating a Recovery Services Vault is discussed earlier in the chapter (see “[Create a Recovery Services Vault](#)”). To configure diagnostics for the Recovery Service Vault, open the **Recovery Services Vault** and then choose **Diagnostic Settings > + Add Diagnostic Setting** (see [Figure 5-78](#)).

ExamRef-RSV | Diagnostic settings

Recovery Services vault

Search (Ctrl+ /) Refresh Provide feedback

Export template

Getting started

Backup

Site Recovery

Protected items

Backup items

Replicated items

Manage

Backup policies

Backup infrastructure

Site Recovery infrastructure

Recovery Plans (Site Recovery)

Backup Reports

Monitoring

Alerts

Diagnostic settings

No diagnostic settings defined

+ Add diagnostic setting

Click 'Add Diagnostic setting' above to configure the collection of the following data:

- AzureBackupReport
- CoreAzureBackup
- AddonAzureBackupJobs
- AddonAzureBackupAlerts
- AddonAzureBackupPolicy
- AddonAzureBackupStorage
- AddonAzureBackupProtectedInstance
- AzureSiteRecoveryJobs
- AzureSiteRecoveryEvents
- AzureSiteRecoveryReplicatedItems
- AzureSiteRecoveryReplicationStats
- AzureSiteRecoveryRecoveryPoints
- AzureSiteRecoveryReplicationDataUploadRate
- AzureSiteRecoveryProtectedDiskDataChurn

FIGURE 5-78 Diagnostic settings for the Azure Recovery Vault

In this example, we have chosen the log categories shown in [Figure 5-79](#), and data is configured to send it to the Log Analytics Workspace with the retention set to the default 30-day setting. If you want to retain data for more than 30 days, then you need to update the **Retention** setting in the Log Analytics Workspace.

Diagnostics setting

 Save  Discard  Delete  Provide feedback

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)

Diagnostic setting name *

ExamRef-BackupReports 

Category details

Destination details

log

Send to Log Analytics

AzureBackupReport

CoreAzureBackup

AddonAzureBackupJobs

AddonAzureBackupAlerts

AddonAzureBackupPolicy

AddonAzureBackupStorage

AddonAzureBackupProtectedInstance

AzureSiteRecoveryJobs

AzureSiteRecoveryEvents

AzureSiteRecoveryReplicatedItems

AzureSiteRecoveryReplicationStats

AzureSiteRecoveryRecoveryPoints

Subscription

Visual Studio Ultimate with MSDN 

Log Analytics workspace

ExamRef-LAWorkspace (canadacentral) 

Destination table 

Azure diagnostics Resource specific

 You need to create separate diagnostics settings for Azure Backup and Azure Site Recovery events to prevent potential data loss. For Azure Backup events, if you choose the 'Resource specific' mode, you must select the following events only - CoreAzureBackup, AddonAzureBackupJobs, AddonAzureBackupAlerts, AddonAzureBackupPolicy, AddonAzureBackupStorage, AddonAzureBackupProtectedInstance. The AzureBackupReport event works only in 'Azure diagnostics' mode. [Learn more](#)

Archive to a storage account

Stream to an event hub

FIGURE 5-79 Diagnostic settings for the Azure Recovery Vault

Once diagnostic settings are configured, you can view the backup report data in the **Recovery Services Vault** by clicking **Backup Reports** under **Manage**, as shown in [Figure 5-80](#).

ExamRef-RSV | Backup Reports

Recovery Services vault

Search (Ctrl+ /) Refresh

Identity Private endpoint connections Properties Locks Export template

Getting started

Backup Site Recovery Protected items

Backup items Replicated items

Manage

Backup policies Backup Infrastructure Site Recovery infrastructure Recovery Plans (Site Recovery)

Backup Reports

Monitoring

Alerts Diagnostic settings Backup Jobs

Overview Summary Backup Items Usage Jobs Policies Optimize

Select Log Analytics Workspace

How to configure vault diagnostic settings for backup reports

What happened to the Power BI reports?

Subscriptions Workspaces

All ExamRef-LAWorkspace

Report Filters

How to use backup reports?

Filters are applied left to right and top to bottom on each page. [Learn More](#)

Time Range

Last 7 days

All datetimes are in UTC. Data for the current partial day is not shown in the reports. [Learn More](#)

Note - There is no backup data in the selected Log Analytics workspace(s) for the selected time range. [Learn More](#)

Welcome to Azure Backup Reports!

Select your Workspace(s) and click the Summary Tab above to get started

Salient Features

View reports across workload types, subscriptions and regions

View reports across your tenants using Azure Lighthouse

Track and forecast usage both at a vault and backup item level

Analyze trends on jobs, backup items, policies, and more

FIGURE 5-80 Backup reports for the Azure Recovery Vault

Thought experiment

In this thought experiment, apply what you have learned. You can find answers to these questions in the next section.

You are the administrator for Trey Research Pharmaceuticals. As a leader in the design and manufacturing of cutting-edge treatments for cancer patients, Trey Research needs to ensure that the users data within the organization are protected as they handle sensitive data and can't accommodate any data loss. Users have their own assigned VMs in Azure that are deployed in Canada Central region.

Trey Research needs to ensure backup solution has the following features:

1. All users' data must be backed up daily at 6 PM Eastern time. The data should be retained for one year from the date it is backed up.
2. If any user's backup data is accidentally deleted, then they should be able to restore it back within two weeks of time.
3. Users should be able to restore their VMs as well as files and folders from the backup data.

Thought experiment answers

This section contains the answers to the thought experiment for the chapter.

1. Create a backup policy with the schedule to execute the backup at 6pm Eastern time with the retention of daily backup point for 365 days.
2. Enable Soft delete feature under Security Settings by visiting the Properties of Recovery Services Vault.
3. Leverage Restore VM and File Recovery options for restoring the VM and restoring files and folders respectively.

Chapter summary

Below are some of the key takeaways from this chapter:

- Azure Monitor is a single pane of glass for accessing Azure metrics,

tenant and resource diagnostic logs, Log Analytics, service health, and alerts.

- You can configure alerts based on metric alerts (captured from Azure Metrics) to Activity Log alerts that can notify by email, webhook, SMS, Logic Apps, or even an Azure Automation Runbook.
- Azure Log Analytics can consolidate machine data from on-premises and cloud-based workloads and this data is indexed and categorized for quick searching. Data can be collected from both Windows and Linux machines.
- Azure Log Analytics has many management solutions that help administrators gain value out of complex machine data. These solutions contain pre-built visualizations and queries that help surface insights quickly.
- Queries in Log Analytics can be saved for quick access and visualized and shared using Azure Dashboards. To analyze data outside of Log Analytics you can export the data to Excel and Power BI.
- Recovery Services Vault is used for configuration and management of both Backup and Site Recovery.
- An Azure Backup policy defines how often backups occur and how long the backups are retained.
- The Azure Backup service can backup and restore an entire virtual machine and you can also use it for just file recovery to restore files from a recovery point without recreating the entire virtual machine.
- Azure Backup can be used to protect files and folders, applications, and IaaS virtual machines. This cloud-based data protection service helps organizations by providing offsite backups of on-premises servers and protection of VM workloads they have already moved to the cloud.
- The backup data is retained for 14 days after deletion by using Soft delete feature.
- Azure Site Recovery service enables us to replicate, failover, and fallback virtual machines as needed.

Index

A

- A records (DNS), [249](#)
- AAAA records (DNS), [249](#)
- accelerated networking, [177–178](#)
- access control for storage accounts, [64–67](#). *See also* RBAC (role-based access control)
 - Azure Files
 - Azure AD DS authentication, [86–89](#)
 - configuring, [84–89](#)
 - on-premises AD DS authentication, [85–86](#)
 - Blob Storage network access levels, [66–67](#)
 - firewalls, [64–65](#)
 - virtual network service endpoints, [65–66](#)
- access keys (Azure Storage), [79–80](#)
- access tiers (Blob Storage), [71](#)
 - configuring, [117–121](#)
- account-level SAS, [77](#)
- account-level tiers (Blob Storage), [117–118](#)
- accounts (Azure AD). *See* [users \(Azure AD\)](#)
- accounts (Azure Storage). *See also* [blobs \(Azure Storage\)](#)
 - access key management, [79–80](#)
 - Azure AD authentication, [80–84](#)
 - configuring, [67–73](#)
 - account types, [69](#)
 - Azure AD authentication, [82–84](#)

Blob Storage access tiers, 71
naming requirements, 68
performance tiers, 68
replication options, 69–70, 98–103
connecting to Azure Storage Explorer, 93–95
copying data with AzCopy, 96–98
 async blob copy service, 97
 sync blob copy service, 98
 uploading/downloading data, 97
creating, 71–73
network access configuration, 64–67
 Blob Storage access levels, 66–67
 firewalls, 64–65
 virtual network service endpoints, 65–66
purpose of, 63
SAS token creation, 73–78
service types, 67–68
ACI (Azure Container Instances), 129, 185–187
 container groups, 185–186
 sizing and scaling, 185–186
Action Groups (Azure Monitor), 356–359
Activity Log, 345
AD (Active Directory), purpose of, 1
AD DS (Active Directory Domain Services), configuring, 85–86
adding
 data disks to VMs, 173–175
 server endpoints in Azure File Sync, 111–112
ADFS (Active Directory Federation Services), purpose of, 1
administrative roles (Azure AD), roles versus, 17

AKS (Azure Kubernetes Service), [129](#), [187–191](#)

clusters

- connecting to, [189–190](#)
- scaling, [188–189](#)
- storage configuration, [187–188](#)
- upgrading, [190–191](#)

Alert Rules (Azure Monitor), [353](#)

alerts (Azure Monitor), [336](#), [352–363](#)

- analyzing across subscriptions, [361–363](#)
- configuring, [353–359](#)
- purpose of, [352–353](#)
- states, [361](#)
- viewing, [359–361](#)

Alias records (Azure DNS), [249–250](#)

aligned availability sets, [153–154](#)

allocating public IP addresses, [229](#)

App Service

- domain registration, [246](#)
- networking, [203–206](#)
- plans

- creating, [192](#)
- scaling, [193–196](#)

purpose of, [191](#)

web apps

- backing up, [201–203](#)
- creating, [197–198](#)
- custom domain names, [199–201](#)
- deploying, [206–209](#)
- security, [198–199](#)

Append Blobs, 68, 114

Application Insights, configuring, 363–365

application rule collection in firewalls, 274–275

application security groups (ASGs), 177, 262–263

applying network routes, 236–237

Archive access tier (Blob Storage), 71, 118

ARM (Azure Resource Manager)

- limitations, 43
- operations available, 20
- tags in, 52
- templates
 - creating custom, 139–144
 - modifying, 137–138
 - network interface creation, 133
 - public IP address addition, 134–135
 - saving deployment as, 144–145
 - schema, 137
 - structure of, 130–137
 - virtual network creation, 131–132
- VM resource creation, 135–137

ASGs (application security groups), 177, 262–263

assigning roles (Azure AD), 16–17, 19

- managing assignments, 25–28, 47–49

associating

- devices (Azure AD), 11–12

ExpressRoute circuits with Azure Virtual WAN, 324–325

NSGs with subnets, 265–266

policies with management groups (Azure AD), 32

route tables with firewalls, 273–274

async blob copy service, 95–97, 99–100
authentication
 for AzCopy, 96–97
 for Azure Files
 Azure AD DS authentication, 86–89
 on-premises AD DS authentication, 85–86
 for storage accounts, 80–84
 for VMs, 179
 for web apps, 198–199
authoritative DNS servers, 244
automating VM configuration, 130–148
 ARM template modification, 137–138
 ARM template structure, 130–137
 network interface creation, 133
 public IP address addition, 134–135
 schema, 137
 virtual network creation, 131–132
 VM resource creation, 135–137
 Custom Script Extension, 145–148
 custom template creation, 139–144
 saving deployment as ARM template, 144–145
 VHD template configuration, 138–139
availability sets for VMs, 151–154
 configurations, 151–152
 creating, 152–153
 managed disks and, 153–154
availability zones
 for VMs, 149–151
 for VMSS, 155

Az module (PowerShell), 53

AzCopy, 96–98

- async blob copy service, 97
- authentication, 96–97
- platform support, 97
- sync blob copy service, 98
- uploading/downloading data, 97

Azure Active Directory Domain Services (Azure AD DS), configuring, 86–89

Azure Activity Log, 345

Azure AD (Azure Active Directory)

- devices
 - configuring Azure AD Join, 11–13
 - managing, 7–8
- governance
 - cost management configuration, 52–59
 - management group configuration, 49–51
 - policy configuration, 30–37
 - resource group management, 41–47
 - resource lock configuration, 37–38
 - resource tag configuration, 38–41
 - subscription management, 47–49
- groups
 - creating, 3–6
 - managing, 6–7
 - types of, 4
- hierarchy of, 30
- purpose of, 1

RBAC (role-based access control)

- creating custom roles, 19–25

management groups and, 51
managing role assignments, 25–28, 49
operational overview, 16–19
for storage accounts, 80–84

resource groups
 creating, 41
 deleting, 45–47
 hierarchy of, 30
 managing, 41–47
 moving resources across, 42–45
 purpose of, 29–30

resources
 hierarchy of, 30
 purpose of, 28–29

subscriptions
 hierarchy of, 30
 purpose of, 28

users
 bulk updating, 8–9
 creating, 3–4
 guest accounts, 9–11
 managing, 6–7
 SSPR (self-service password reset), 14–15
 types of, 3

Azure AD B2B (Business-to-Business), purpose of, 1
Azure AD B2C (Business-to-Customer), purpose of, 1
Azure AD Connect, purpose of, 1
Azure AD DS (Azure Active Directory Domain Services), configuring, 86–89
Azure AD Join, configuring, 11–13

Azure App Service
domain registration, 246

networking, 203–206

plans

creating, 192

scaling, 193–196

purpose of, 191

web apps

backing up, 201–203

creating, 197–198

custom domain names, 199–201

deploying, 206–209

security, 198–199

Azure Application Gateway

configuring, 283–287

documentation, 282

purpose of, 282

Azure Backup

Azure workload backups, 371–373

Azure workload restoration, 374–377

backup report configuration, 390–392

on-premises workload backups, 374–383

on-premises workload restoration, 383

purpose of, 334, 365

Azure Backup Server

installing, 377–379

purpose of, 377

Azure Bastion Service, configuring, 279–282

Azure Blob Storage. *See* [Blob Storage](#)

Azure CLI

ARM template parameters, [144](#)

Azure Bastion Service, deploying, [282](#)

Azure Firewall, deploying, [279](#)

blob management, [115](#)

storage accounts

access key generation, [79](#)

async blob copy service, [100](#)

creating, [73](#)

Azure Cloud Shell, [185](#)

Azure Container Instances. *See* [ACI \(Azure Container Instances\)](#)

Azure Container Networking Interface (CNI), [189](#)

Azure Disk Encryption, [161–170](#)

cost of, [162](#)

disabling, [169–170](#)

enabling

with CMK (customer-managed keys), [166–168](#)

on existing VMs, [162–166](#)

on new data disks, [168–169](#)

Azure DNS

Alias records, [249–250](#)

configuring

custom DNS settings, [253–255](#)

private DNS zones, [255–257](#)

DNS records

creating, [250–253](#)

managing, [248–249](#)

DNS zones

creating, [250–253](#)

delegating, 247–248
purpose of, 246

Azure File Sync
configuring, 108–113
agent deployment, 109–111
health monitoring, 112–113
server endpoint addition, 111–112
sync group creation, 108
purpose of, 108
troubleshooting, 112

Azure Files
access control
Azure AD DS authentication, 86–89
configuring, 84–89
on-premises AD DS authentication, 85–86
account registration, 87
disaster recovery, 374
file shares
creating, 104–105
Linux connections, 107
non-Azure connections, 105
Windows connections, 105–107
purpose of, 104

Azure Firewall, configuring, 268–279
application rule collection, 274–275
deployment, 271–272
DNAT rule creation, 278–279
network rule collection, 275–277
route table creation and association, 273–274

testing, 277–278
VM creation, 270–271
VNet and subnet creation, 269

Azure Import/Export
 exporting data, 89–90
 importing data, 91–93
 purpose of, 89

Azure Key Vault, 80
 cost of, 162

Azure Kubernetes Service. *See AKS (Azure Kubernetes Service)*

Azure Load Balancer, 287–290
 configuring, 290–294
 backend configuration, 289
 frontend IP configuration, 288–289
 health probes, 289–290
 pricing tiers, 288
 purpose of, 283
 troubleshooting, 294–295

Azure Monitor
 alerts, 352–363
 analyzing across subscriptions, 361–363
 configuring, 353–359
 purpose of, 352–353
 states, 361
 viewing, 359–361
 log queries and analysis, 347–352
 metrics configuration, 336–340
 purpose of, 334–336

Azure Monitor for Containers, 335

Azure Monitor for VMs, 335

Azure Policy

configuring, 30–37

scope, 50–51

Azure Site Recovery, 384–390

Azure Storage. *See also* Blob Storage

access key management, 79–80

account creation and configuration, 67–73

account types, 69

Blob Storage access tiers, 71

naming requirements, 68

performance tiers, 68

replication options, 69–70, 98–103

AzCopy, 96–98

async blob copy service, 97

authentication, 96–97

platform support, 97

sync blob copy service, 98

uploading/downloading data, 97

Azure AD authentication, 80–84

exporting data, 89–90

importing data, 91–93

network access configuration, 64–67

Blob Storage access levels, 66–67

firewalls, 64–65

virtual network service endpoints, 65–66

purpose of, 63

SAS token creation, 73–78

service types, 67–68

Azure Storage Explorer, 93–96
 async blob copy service, 95–96
 AzCopy and, 96
 blob management, 116
 connecting to storage accounts, 93–95
 installing, 93
 supported operations, 95

Azure Traffic Manager, purpose of, 246

Azure Virtual Networks (VNets). *See* [virtual networks](#)

Azure Virtual WAN, configuring, 320–325
 creating in Azure portal, 320–321
 ExpressRoute association, 324–325
 point-to-site connections, 323–324
 site-to-site connections, 322–323

Azure VPN Gateway
 configuring, 311–315
 BGP (Border Gateway Protocol), 312
 creating in Azure portal, 313–315
 high availability, 312–313
 subnets, 311, 313–315
 pricing tiers, 311–312

AzureCloud service tag, 260

AzureLoadBalancer server tag, 260

AzureRm module (PowerShell), 53

AzureTrafficManager service tag, 260

B

backend configuration in Azure Load Balancer, 289
backing up web apps, 201–203

backup and recovery. *See* [disaster recovery](#)

backup policies, configuring, [368–371](#)

backup reports, configuring, [390–392](#)

BGP (Border Gateway Protocol), [312](#)

Blob Storage. *See also* [blobs \(Azure Storage\)](#)

- access tiers, [71](#)
 - configuring, [117–121](#)
- account types, [69](#)
- configuring, [113–117](#)
 - Azure portal management, [114–115](#)
 - blob containers, [113–114](#)
 - soft delete, [116–117](#)
 - Storage Explorer management, [116](#)
 - types of blobs, [114](#)
- lifecycle management configuration, [121–125](#)
- network access levels, [66–67](#)
- object replication configuration, [100–103](#)
- purpose of, [113](#)
- uploading/downloading data, [97](#)

blob-level tiers (Blob Storage), [118–119](#)

blobs (Azure Storage). *See also* [Blob Storage](#)

- Azure Storage Explorer operations, [95](#)
- change feed, [100](#)
- containers, [113–114](#)
 - Azure portal management, [114–115](#)
 - Storage Explorer management, [116](#)
- exporting, [89–90](#)
- purpose of, [67](#)
- RBAC roles, [81](#)

scope, 82
soft delete, 116–117
types of, 68, 114
versioning, 100

Block Blobs, 68, 114

Border Gateway Protocol (BGP), 312

budgets (Azure Cost Management), 53–55

built-in roles (Azure AD), 17

cloning, 20–25

bulk updating users (Azure AD), 8–9

C

CAA records (DNS), 249

change feed for blobs, 100

changing

- access tiers (Blob Storage), 119–121
- storage account replication mode, 99

child DNS zones, delegating, 247–248

CIDR (classless inter-domain routing) notation, 214

circuits (ExpressRoute)

- associating with Azure Virtual Wan, 324–325
- cost of, 319
- creating, 318–319
- peering, 316–317

cloning roles (Azure AD), 20–25

Cloud Shell, 185

cloud tiering, 111

cluster autoscaler, 189

clusters (AKS)

connecting to, 189–190
scaling, 188–189
storage configuration, 187–188
upgrading, 190–191

cmdlets (PowerShell), referencing, 53

CMK (customer-managed keys), 166–168

CNAME records (DNS), 249

CNI (Azure Container Networking Interface), 189

Compute Optimized size type (VMs), 172

compute resources. *See ACI (Azure Container Instances); AKS (Azure Kubernetes Service); VMs (virtual machines)*

configuring

- access control (Azure Files), 84–89
 - Azure AD DS authentication, 86–89
 - on-premises AD DS authentication, 85–86
- access tiers (Blob Storage), 117–121
- accounts (Azure Storage), 67–73
 - account types, 69
 - Azure AD authentication, 82–84
 - Blob Storage access tiers, 71
 - naming requirements, 68
 - network access, 64–67
 - performance tiers, 68
 - replication options, 69–70, 98–103
 - SAS token creation, 73–78
- AKS (Azure Kubernetes Service)
 - scaling, 188–189
 - storage, 187–188
- alerts (Azure Monitor), 353–359

Application Insights, 363–365

Azure AD Join, 11–13

Azure Application Gateway, 283–287

Azure Bastion Service, 279–282

Azure DNS

- custom DNS settings, 253–255
- private DNS zones, 255–257

Azure File Sync, 108–113

- agent deployment, 109–111
- health monitoring, 112–113
- server endpoint addition, 111–112
- sync group creation, 108

Azure Firewall, 268–279

- application rule collection, 274–275
- deployment, 271–272
- DNAT rule creation, 278–279
- network rule collection, 275–277
- route table creation and association, 273–274
- testing, 277–278
- VM creation, 270–271

VNet and subnet creation, 269

Azure Load Balancer, 290–294

- backend configuration, 289
- frontend IP configuration, 288–289

Azure Policy, 30–37

Azure Virtual WAN, 320–325

- creating in Azure portal, 320–321
- ExpressRoute association, 324–325
- point-to-site connections, 323–324

site-to-site connections, 322–323

Azure VPN Gateway, 311–315

- BGP (Border Gateway Protocol), 312
- creating in Azure portal, 313–315
- high availability, 312–313
- subnets, 311, 313–315

backup policies, 368–371

backup reports, 390–392

Blob Storage, 113–117

- Azure portal management, 114–115
- blob containers, 113–114
- soft delete, 116–117
- Storage Explorer management, 116
- types of blobs, 114

cost management (Azure AD), 52–59

- cost center quotas, 53–55
- monitoring and reporting spend, 56–59
- resource quotas, 52–53

ExpressRoute, 315–320

- circuit creation, 318–319
- circuit peering, 316–317
- connectivity models, 315–316
- global availability, 317–318
- peering, 316–317
- virtual network connections, 320

ExpressRoute Monitor, 301–302

lifecycle management (Blob Storage), 121–125

Log Analytics, 340–347

- agent installation, 344

agent ports and protocols, 344
diagnostic settings, 344–347
workspace implementation, 340–344

management groups (Azure AD), 49–51
metrics, 336–340

NPM (Network Performance Monitor), 296–299

object replication (Blob Storage), 100–103

Performance Monitor, 299–300

resource locks (Azure AD), 37–38

resource tags (Azure AD), 38–41

Service Connectivity Monitor, 300–301

SSPR (self-service password reset), 14–15

virtual networks

- creating in Azure portal, 217–219
- IP ranges, 214–215
- network interfaces, 225–226
- network routes, 232–239
- peering, 220–225
- private endpoints, 241–243
- private IP addresses, 226–228
- properties, 215–216
- public IP addresses, 228–232
- service endpoints, 239–241
- subnets, 215

VMs (virtual machines)

- ARM template modification, 137–138
- ARM template structure, 130–137
- automating configuration, 130–148
- Azure Disk Encryption, 161–170

Custom Script Extension, 145–148
custom template creation, 139–144
high availability, 148–154
networking, 175–183, 225
saving deployment as ARM template, 144–145
scalability, 154–161
VHD template configuration, 138–139

connecting

to AKS (Azure Kubernetes Service), 189–190

to Azure Files

Linux connections, 107

non-Azure connections, 105

Windows connections, 105–107

storage accounts to Azure Storage Explorer, 93–95

to VMs

authentication, 179

Linux VM connections with SSH, 182–183

network interface creation, 179–181

options for, 179

Windows VM connections with Remote Desktop, 181–182

Connection Monitor tool, 309–310

Connection Troubleshoot tool, 307–309

connectivity models (ExpressRoute), 315–316

container groups, ACI (Azure Container Instances), 185–186

containers

ACI (Azure Container Instances), 185–187

container groups, 185–186

sizing and scaling, 185–186

advantages of, 184

AKS (Azure Kubernetes Service), [187–191](#)
 cluster upgrades, [190–191](#)
 connecting to, [189–190](#)
 scaling, [188–189](#)
 storage configuration, [187–188](#)

Azure Monitor for Containers, [335](#)

Blob Storage, [113–114](#)
 Azure portal management, [114–115](#)
 Storage Explorer management, [116](#)

Cool access tier (Blob Storage), [71](#)

copying with AzCopy, [96–98](#)

cost center quotas (Azure AD), [53–55](#)

cost management (Azure AD), configuring, [52–59](#)
 cost center quotas, [53–55](#)
 monitoring and reporting spend, [56–59](#)
 resource quotas, [52–53](#)

custom ARM templates, creating, [139–144](#)

custom DNS settings, configuring, [253–255](#)

custom domain names for web apps, [199–201](#)

custom roles (Azure AD), creating, [19–25](#)

Custom Script Extension, [145–148](#)

customer-managed keys (CMK), [166–168](#)

D

data disks, adding to VMs, [173–175](#)

default NSG rules, [261](#)

delegating
 DNS domains, [245](#)
 DNS zones, [247–248](#)

deleting

devices (Azure AD), 8

resource groups (Azure AD), 45–47

role assignments (Azure AD), 28

deny assignments (RBAC), 19, 26

deploying web apps, 206–209

deployment slots, 206–208

devices (Azure AD)

configuring Azure AD Join, 11–13

managing, 7–8

diagnostic logs, 302, 344–347

disabling

Azure Disk Encryption, 169–170

devices (Azure AD), 8

SMB (Server Message Block) v1, 105

disaster recovery

Azure Backup

Azure workload backups, 371–373

Azure workload restoration, 374–377

backup report configuration, 390–392

on-premises workload backups, 374–383

on-premises workload restoration, 383

purpose of, 334, 365

Azure Site Recovery, 384–390

Recovery Services Vault

backup policy configuration, 368–371

creating, 366

Soft Delete option, 366–368

disks (Azure Storage), 68

DNAT rules, creating, 278–279

DNS (Domain Name System). *See also Azure DNS*

in Azure, 246

labels, 230–231

operational overview, 243–246

records

creating, 250–253

managing, 248–249

for web apps, 199–201

DNS resolvers, 244–245

DNS zones

creating, 250–253

delegating, 247–248

private zones, configuring, 255–257

purpose of, 244

Domain Name System. *See DNS (Domain Name System)*

domain names, 243–244

for web apps, 199–201

downloading with AzCopy, 97

dynamic groups (Azure AD), creating, 5–6

dynamic private IP addresses, 226–227

dynamic public IP addresses, 229

E

effective security rules, evaluating, 267–268

enabling

Azure Disk Encryption

with CMK (customer-managed keys), 166–168

on existing VMs, 162–166

on new data disks, 168–169
diagnostic logs, 345–346
IP forwarding, 236
Network Watcher, 302
static private IP addresses, 227–228
encryption. *See* Azure Disk Encryption
endpoints
 private endpoints, configuring, 241–243
 service endpoints, configuring, 239–241
evaluating effective security rules, 267–268
exporting Azure Storage data, 89–90
ExpressRoute
 associating with Azure Virtual Wan, 324–325
 configuring, 315–320
 circuit creation, 318–319
 circuit peering, 316–317
 connectivity models, 315–316
 global availability, 317–318
 peering, 316–317
 virtual network connections, 320
 purpose of, 315
 site-to-site VPNs versus, 315
ExpressRoute Monitor
 configuring, 301–302
 purpose of, 296

F

fault domains for availability sets, 152
file shares (Azure Files)

Azure File Sync, configuring, [108–113](#)
creating, [104–105](#)
Linux connections, [107](#)
non-Azure connections, [105](#)
purpose of, [104](#)
Windows connections, [105–107](#)

files (Azure Storage)
Azure Storage Explorer operations, [95](#)
purpose of, [68](#)

firewalls
Azure Firewall, configuring, [268–279](#)
for storage accounts, [64–65](#)

forced tunneling, [237](#)

FQDNs (fully qualified domain names), [245, 275](#)

frontend IP configuration in Azure Load Balancer, [288–289](#)

Function Apps, [358](#)

functions for ARM templates, [131](#)

G

General Purpose size type (VMs), [172](#)

geographically redundant storage (GRS), [70](#)

geographically zone redundant storage (GZRS), [70](#)

global availability of ExpressRoute, [317–318](#)

global VNet peering, [220](#)

glue records (DNS), [247](#)

governance (Azure AD)
cost management configuration, [52–59](#)
cost center quotas, [53–55](#)
monitoring and reporting spend, [56–59](#)

- resource quotas, 52–53
- management group configuration, 49–51
- policy configuration, 30–37
- resource group management, 41–47
- resource lock configuration, 37–38
- resource tag configuration, 38–41
- subscription management, 47–49

GPU Optimized size type (VMs), 172

graphs for queries, 350–352

groups (Azure AD)

- creating, 3–6
- managing, 6–7
- role assignment, 17
- types of, 4

GRS (geographically redundant storage), 70

guest accounts (Azure AD), managing, 9–11

GZRS (geographically zone redundant storage), 70

H

hardware security modules (HSMs), 80

health monitoring

- in Azure File Sync, 112–113
- in Azure Load Balancer, 289–290
- in VMSS, 158–159

high availability

- for VMs, 148–154
 - availability sets, 151–154
 - availability zones, 149–151
- for VPN gateways, 312–313

High Performance Compute size type (VMs), 172
horizontal pod autoscaler (HPA), 189
Hot access tier (Blob Storage), 71
HPA (horizontal pod autoscaler), 189
HSMs (hardware security modules), 80
HTTPS protocol, 67
hub-and-spoke networks, service chaining in, 222
hybrid joining devices (Azure AD), 11, 13
hybrid networks
 Azure Virtual WAN configuration, 320–325
 Azure VPN Gateway configuration, 311–315
 ExpressRoute configuration, 315–320
 purpose of, 310–311
 verifying and troubleshooting, 320

I

IaC (Infrastructure as Code), 137
importing Azure Storage data, 91–93
inbound rules, default, 261
infrastructure FQDNs, 275
inheritance
 of resource locks (Azure AD), 37
 of roles (Azure AD), 16, 18
installing
 Azure Backup Server, 377–379
 Azure File Sync agent, 109–111
 Azure Storage Explorer, 93
 MARS (Microsoft Azure Recovery Services) agent, 379–383
 NPM (Network Performance Monitor), 296–299

Internal DNS, [246](#)
internal Load Balancers, [288](#)
Internet default rule, [261](#)
Internet service tag, [260](#)
IP addresses
 private, configuring, [226–228](#)
 public, configuring, [228–232](#)
 types of, [225](#)
IP Flow Verify tool, [303](#)
IP forwarding, [236](#)
IP ranges, [214–215](#)
IPv4 public addresses, [232](#)
IPv6 public addresses, [232](#)
ITSM (IT Service Manager) actions, [358](#)

J

joining devices (Azure AD), [11, 13](#)
JSON (JavaScript Object Notation) files
 custom roles (Azure AD), creating, [25](#)
 schema files in ARM templates, [131](#)

K

Kubectl, [188–189](#)
kubenet, [189](#)
Kubernetes. *See AKS (Azure Kubernetes Service)*
Kusto, [348](#)

L

large scale sets (VMSS), [154](#)
LDNS (local DNS service), [244–245](#)

license requirements, SSPR (self-service password reset), 14
lifecycle management (Blob Storage), configuring, 121–125
Linux connections to Azure Files, 107
Linux VMs, SSH connections, 182–183
listings
 creating network interface, 133–134
 IP configurations, 135
 template structure for creating virtual network, 132
 variables for virtual network creation, 132
 virtual machine resource, 136
Load Balancer default rule, 261
load balancing, 282–295
 Azure Application Gateway
 configuring, 283–287
 documentation, 282
 purpose of, 282
 Azure Load Balancer, 287–290
 backend configuration, 289
 configuring, 290–294
 frontend IP configuration, 288–289
 health probes, 289–290
 pricing tiers, 288
 purpose of, 283
 troubleshooting, 294–295
local DNS service (LDNS), 244–245
locally redundant storage (LRS), 69
Log Analytics, 335
 configuring, 340–347
 agent installation, 344

agent ports and protocols, 344
diagnostic settings, 344–347
workspace implementation, 340–344
log queries and analysis, 347–352
purpose of, 340
Logic Apps, 358
logs
 diagnostic, 344–347
 metrics versus, 335
 purpose of, 347
 queries and analysis, 347–352
LRS (locally redundant storage), 69

M

managed disks, availability sets and, 153–154
management groups (Azure AD), 18
 associating policies with, 32
 configuring, 49–51
management locks (Azure AD), configuring, 37–38
managing
 blobs (Azure Storage)
 in Azure portal, 114–115
 in Azure Storage Explorer, 116
 devices (Azure AD), 7–8
 groups (Azure AD), 6–7
 guest accounts (Azure AD), 9–11
 records (DNS), 248–249
 resource groups (Azure AD), 41–47
 role assignments (Azure AD), 25–28, 47–49

subscriptions (Azure AD), 47–49
users (Azure AD), 6–7
VMs (virtual machines)
 Custom Script Extension, 145–148
 data disk addition, 173–175
 moving across subscriptions/resource groups, 170–171
 size types, 172–173
MARS (Microsoft Azure Recovery Services) agent, installing, 379–383
Memory Optimized size type (VMs), 172
metrics
 configuring, 336–340
 logs versus, 335
 purpose of, 347
Microsoft 365, 3
migrating on-premises workloads to Azure, 390
modifying ARM templates, 137–138
monitoring
 Application Insights, 363–365
 Azure Monitor
 alerts, 352–363
 purpose of, 334–336
 Log Analytics, configuring, 340–347
 logs
 metrics versus, 335
 queries and analysis, 347–352
metrics
 configuring, 336–340
 logs versus, 335
spend, 56–59

strategy development for, 333
synchronization, 112–113
virtual networks
 diagnostic logs, 302
 Network Watcher, 302–306
 NPM (Network Performance Monitor), 296–302
 VMSS (VM scale sets), 158–159
mounting. *See* connecting
moving resources (Azure AD) across resource groups, 42–45
MX records (DNS), 249
MySQL in-app, 201

N

name resolution. *See* Azure DNS; DNS (Domain Name System)
naming requirements for storage accounts, 68
net use command, 107
network access, configuring for storage accounts, 64–67
network interfaces
 associating NSGs with, 265–266
 configuring, 225–226
 creating, 133, 179–181
Network Performance Monitor. *See* NPM (Network Performance Monitor)
network routes, 232–239
 applying, 236–237
 forced tunneling, 237
 IP forwarding, 236
 purpose of, 232
 system routes, 232–234
 user-defined routes, 234–239

network rule collection in firewalls, [275–277](#)

network security groups. *See* [NSGs \(network security groups\)](#)

Network Topology tool, [306](#)

Network Watcher, [302–306](#)

- Connection Monitor tool, [309–310](#)
- Connection Troubleshoot tool, [307–309](#)
- deploying, [302](#)
- IP Flow Verify tool, [303](#)
- Network Topology tool, [306](#)
- Next Hop tool, [304](#)
- Packet Capture tool, [305](#)
- purpose of, [302](#)
- VPN Troubleshoot tool, [307](#)

networking. *See also* [virtual networks](#)

- in Azure App Service, [203–206](#)
- configuring for VMs, [175–183](#)
 - accelerated networking, [177–178](#)
 - authentication, [179](#)
 - connection options, [179](#)
 - IP address types, [225](#)
 - Linux VM connections with SSH, [182–183](#)
 - network interface creation, [179–181](#)
 - Windows VM connections with Remote Desktop, [181–182](#)
- Next Hop tool, [304](#)
- next hops, types of, [234](#)
- notifications (Azure Monitor), [356–358](#)
- NPM (Network Performance Monitor), [296–302](#)
 - deploying, [296–299](#)
- ExpressRoute Monitor configuration, [301–302](#)

Performance Monitor configuration, 299–300
Service Connectivity Monitor configuration, 300–301
services in, 296
NS records (DNS), 245, 249
NSGs (network security groups), 176
 associating with subnets, 265–266
 creating in Azure portal, 263–265
 default rules, 261
 evaluating effective rules, 267–268
 priority of rules, 259
 properties, 258–259
 purpose of, 258
 service tags, 260

O

object replication (Blob Storage), configuring, 100–103
Office 365, 3
Office 365 groups (Azure AD), 4
on-premises AD DS authentication, configuring, 85–86
outbound Internet connections, 231
outbound rules, default, 261
outputs for ARM templates, 131

P

Packet Capture tool, 305
Page Blobs, 68, 114
parameters for ARM templates, 131
passwords (Azure AD), SSPR (self-service password reset), 14–15
peering virtual networks, 220–225

creating in Azure portal, 223–225
ExpressRoute circuits, 316–317
limitations, 220
purpose of, 220
service chaining in hub-and-spoke networks, 222
sharing virtual network gateways, 222–223

Performance Monitor
configuring, 299–300
purpose of, 296
performance tiers for storage accounts, 68
permissions. *See RBAC (role-based access control)*
persistent volumes, 188
placement groups (VMSS), 154
point-to-site VPNs, creating in Azure Virtual WAN, 323–324
policies (Azure AD)
configuring, 30–37
scope, 50–51
Policy definitions (Azure AD), creating, 30–37
PowerShell
ARM template parameters, 144
Azure Bastion Service, deploying, 282
Azure Firewall, deploying, 279
blob management, 115
cmdlets, referencing, 53
storage accounts
access key generation, 79
async blob copy service, 99
creating, 73
prefixes for public IP addresses, 230–231

Premium tier (Azure Storage)

account types, [69](#)

purpose of, [68](#)

replication options, [68](#)

pricing tiers

for Azure Load Balancer, [288](#)

for public IP addresses, [228–229](#)

for VPN gateways, [311–312](#)

priority of NSG rules, [259](#)

private DNS zones, configuring, [255–257](#)

private endpoints, configuring, [241–243](#)

private IP addresses, configuring, [226–228](#)

properties

of DNS records, [248](#)

of network interfaces, [226](#)

of NSG rules, [258–259](#)

of subnets, [215–216](#)

of virtual networks, [215–216](#)

proximity placement groups, [153](#)

PTR records (DNS), [249](#)

public IP addresses

adding to VMs, [134–135](#)

allocating, [229](#)

configuring, [228–232](#)

creating in Azure portal, [232](#)

IPv4 versus IPv6, [232](#)

outbound Internet connections, [231](#)

prefixes, [230–231](#)

pricing tiers, [228–229](#)

public Load Balancers, 288

Q

queries, 347–352

 creating, 348–350

 graphs for, 350–352

 saving to dashboard, 350

queues (Azure Storage)

 Azure Storage Explorer operations, 95

 purpose of, 68

 RBAC roles, 81

 scope, 82

R

RBAC (role-based access control)

 custom roles, creating, 19–25

 management groups and, 51

 operational overview, 16–19

 role assignments, managing, 25–28, 49

 for storage accounts, 80–84

reconnecting to Azure Files in Windows, 107

records (DNS)

 creating, 250–253

 managing, 248–249

 for web apps, 199–201

recovery. *See* [disaster recovery](#)

Recovery Services Vault

 backup policy configuration, 368–371

 creating, 366

Soft Delete option, [366–368](#)
recursive DNS servers, purpose of, [244–246](#)
redeploying VMs (virtual machines), [183–184](#)
referencing cmdlets (PowerShell), [53](#)
regions, availability zones in, [149](#), [151](#)
registering devices (Azure AD), [11–13](#)
registration VNets, [256](#)
Remote Desktop connections to Windows VMs, [181–182](#)
removing. *See* [deleting](#)
replication options
 object replication configuration, [100–103](#)
 for storage accounts, [68–70](#), [98–100](#)
reporting spend, [56–59](#)
resiliency. *See* [high availability](#)
resolution VNets, [256](#)
resource groups (Azure AD)
 creating, [41](#)
 deleting, [45–47](#)
 governance, [32](#)
 hierarchy of, [30](#)
 managing, [41–47](#)
 metrics analysis, [339–340](#)
 moving resources across, [42–45](#), [170–171](#)
 purpose of, [29–30](#)
resource locks (Azure AD), configuring, [37–38](#)
resource quotas (Azure AD), [52–53](#)
resource record sets (RRSets), [248](#)
resource tags (Azure AD), configuring, [38–41](#)
resources (ARM templates), [131](#)

resources (Azure AD)
 hierarchy of, 30
 moving across resource groups, 42–45, 170–171
 purpose of, 28–29
reverse DNS, purpose of, 245–246
role-based access control. *See* RBAC (role-based access control)
roles (Azure AD), 16
 administrative roles versus, 17
 assigning, 16–17, 19
 managing assignments, 25–28, 47–49
 built-in roles, 17
 cloning, 20–25
 creating, 19–25
 definitions, 17
 inheritance, 16, 18
 scope, 18
route tables
 associating with firewalls, 273–274
 creating, 235
routes. *See* network routes
routing loops, 236
RRSets (resource record sets), 248
Runbooks, 358

S

SAP HANA on Azure VM, disaster recovery, 374
SAS (shared access signature) tokens, 67, 73–78
saving
 deployment as ARM template, 144–145

queries to dashboard, 350

scalability for VMs, 154–161

scale sets. *See* [VMSS \(VM scale sets\)](#)

scaling

- ACI (Azure Container Instances), 185–186
- AKS (Azure Kubernetes Service), 188–189
- App Service plans, 193–196

schema files (JSON) in ARM templates, 131

scope

- in Azure Cost Management, 58
- in Azure Policy, 32, 50–51
- in RBAC, 18
- for storage accounts, 82

secure shell (SSH) protocol, Linux VM connections, 182–183

security

- in AKS (Azure Kubernetes Service), 190
- of virtual networks
 - Azure Bastion Service, 279–282
 - Azure Firewall, 268–279
 - effective security rule evaluation, 267–268
 - security rule association with subnets, 265–266
 - security rule creation, 258–265
- for web apps, 198–199

security groups (Azure AD), 4

security principals (Azure AD), 16

- role assignment, 16–17, 19, 25–28
- role definitions, 17
- role inheritance, 16, 18

security rules. *See* [NSGs \(network security groups\)](#)

self-service password reset (SSPR), 14–15

server endpoints, adding in Azure File Sync, 111–112

Server Message Block (SMB)

- access control (Azure Files), 84–89
- disabling, 105

service chaining in hub-and-spoke networks, 222

Service Connectivity Monitor

- configuring, 300–301
- purpose of, 296

service endpoints, configuring on subnets, 239–241

service tags, 260

shared access signature (SAS) tokens, 67, 73–78

sharing virtual network gateways, 222–223

site-to-site VPNs

- creating in Azure Virtual WAN, 322–323
- ExpressRoute versus, 315

size types for VMs, 172–173

sizing ACI (Azure Container Instances), 185–186

SMB (Server Message Block)

- access control (Azure Files), 84–89
- disabling, 105

SNAT (Source Network Address Translation), 231

SOA records (DNS), 249

soft delete for blobs, 116–117

Soft Delete option (Recovery Services Vault), 366–368

source control for ARM templates, 131

spending quotas (Azure AD), 52

SPF records (DNS), 249

spreading algorithm (VMSS), 160

SQL Server on Azure VM, disaster recovery, 374

Sql service tag, 260

SRV records (DNS), 249

SSH (secure shell) protocol, Linux VM connections, 182–183

SSPR (self-service password reset), 14–15

Standard tier (Azure Storage)

- account types, 69
- purpose of, 68

static private IP addresses

- in ARM templates, 133
- configuring, 227
- for DNS, 253
- enabling, 227–228
- purpose of, 227

static public IP addresses, 229

storage

- Azure Files
 - access control configuration, 84–89
 - account registration, 87
- Azure Storage
 - access key management, 79–80
 - account creation and configuration, 67–73
 - AzCopy, 96–98
 - Azure AD authentication, 80–84
 - exporting data, 89–90
 - importing data, 91–93
 - network access configuration, 64–67
 - purpose of, 63
 - replication implementation, 98–103

SAS token creation, 73–78
service types, 67–68

Azure Storage Explorer, 93–96
 async blob copy service, 95–96
 connecting to storage accounts, 93–95
 installing, 93
 supported operations, 95

Blob Storage
 access tier configuration, 117–121
 configuring, 113–117
 lifecycle management configuration, 121–125
 object replication configuration, 100–103
 configuring for AKS (Azure Kubernetes Service), 187–188

Storage Explorer. *See* [Azure Storage Explorer](#)

Storage Optimized size type (VMs), 172

Storage service tag, 260

stored access policies, 77–78

subnets
 associating NSGs with, 265–266
 configuring, 215
 service endpoints, 239–241
 creating, 217–219
 for firewalls, 269
 properties, 215–216
 purpose of, 213, 215
 for VPN gateways, 311, 313–315

subscriptions (Azure AD)
 administrator roles, 47–49
 alert analysis across, 361–363

hierarchy of, 30, 49–50
managing, 47–49
metrics analysis, 339–340
monitoring and reporting spend, 56–59
moving resources across, 42–43, 170–171
purpose of, 28
in RBAC, 18
transferring ownership, 42
types of, 47

sync blob copy service, 98

sync groups (Azure File Sync), creating, 108

synchronization. *See Azure File Sync*

system routes, 232–234

T

tables (Azure Storage)
 Azure Storage Explorer operations, 95
 purpose of, 68

tags (Azure AD)
 configuring, 38–41
 purpose of, 52

templates (ARM)
 creating custom, 139–144
 modifying, 137–138
 network interface creation, 133
 public IP address addition, 134–135
 saving deployment as, 144–145
 schema, 137
 structure of, 130–137

virtual network creation, [131–132](#)
VM resource creation, [135–137](#)
templates (VHD), configuring, [138–139](#)
testing Azure Firewall, [277–278](#)
transferring subscription ownership (Azure AD), [42](#)
troubleshooting
 Azure File Sync, [112](#)
 Custom Script Extension, [148](#)
 hybrid networks, [320](#)
 load balancing, [294–295](#)
 virtual networks, [306–310](#)
 Connection Monitor tool, [309–310](#)
 Connection Troubleshoot tool, [307–309](#)
 VPN Troubleshoot tool, [307](#)
TXT records (DNS), [249](#)

U

UDRs (user-defined routes)
 creating in Azure portal, [237–239](#)
 purpose of, [234–236](#)
update domains for availability sets, [152](#)
updating users (Azure AD), bulk updates, [8–9](#)
upgrading
 clusters (AKS), [190–191](#)
 VMSS (VM scale sets), [156–157](#)
uploading with AzCopy, [97](#)
URI (uniform resource identifier) for SAS tokens, [76](#)
user delegation SAS, [77](#)
users (Azure AD)

- bulk updating, 8–9
- creating, 3–4
- guest accounts, 9–11
- managing, 6–7
- SSPR (self-service password reset), 14–15
- types of, 3

V

- validating ARM templates, 141
- variables for ARM templates, 131
- verifying hybrid networks, 320
- versioning blobs (Azure Storage), 100
- VHD templates, configuring, 138–139
- viewing alerts (Azure Monitor), 359–361
- virtual machine resources, creating, 135–137
- virtual machines. *See* VMs (virtual machines)
- virtual network appliances, 236
- Virtual Network default rule, 261
- virtual network gateways
 - Azure VPN Gateway configuration, 311–315
 - sharing, 222–223
- virtual network service endpoints for storage accounts, 65–66
- virtual networks
 - configuring
 - creating in Azure portal, 217–219
 - IP ranges, 214–215
 - network interfaces, 225–226
 - network routes, 232–239
 - peering, 220–225

private endpoints, 241–243
private IP addresses, 226–228
properties, 215–216
public IP addresses, 228–232
service endpoints, 239–241
subnets, 215
creating, 131–132
hybrid networks
 Azure Virtual WAN configuration, 320–325
 Azure VPN Gateway configuration, 311–315
 ExpressRoute configuration, 315–320
 purpose of, 310–311
 verifying and troubleshooting, 320
load balancing, 282–295
 Azure Application Gateway, 282–287
 Azure Load Balancer, 283, 287–290
 troubleshooting, 294–295
monitoring
 diagnostic logs, 302
 Network Watcher, 302–306
 NPM (Network Performance Monitor), 296–302
name resolution. *See* Azure DNS
peering, 220–225
 creating in Azure portal, 223–225
 limitations, 220
 purpose of, 220
 service chaining in hub-and-spoke networks, 222
 sharing virtual network gateways, 222–223
purpose of, 213

security

Azure Bastion Service, [279–282](#)

Azure Firewall, [268–279](#)

effective security rule evaluation, [267–268](#)

security rule association with subnets, [265–266](#)

security rule creation, [258–265](#)

troubleshooting, [306–310](#)

Connection Monitor tool, [309–310](#)

Connection Troubleshoot tool, [307–309](#)

VPN Troubleshoot tool, [307](#)

VirtualNetwork service tag, [260](#)

VMs (virtual machines)

Azure Monitor for VMs, [335](#)

backups, [371–373](#)

configuring

ARM template modification, [137–138](#)

ARM template structure, [130–137](#)

automating configuration, [130–148](#)

Azure Disk Encryption, [161–170](#)

Custom Script Extension, [145–148](#)

custom template creation, [139–144](#)

high availability, [148–154](#)

networking, [175–183, 225](#)

saving deployment as ARM template, [144–145](#)

scalability, [154–161](#)

VHD template configuration, [138–139](#)

creating for firewalls, [270–271](#)

managing

data disk addition, [173–175](#)

moving across subscriptions/resource groups, 170–171
size types, 172–173
purpose of, 129
redeploying, 183–184
restoration, 374–377
 with Azure Site Recovery, 384–390
 with Soft Delete, 366–368
VMSS (VM scale sets)
 configuring, 154–161
 purpose of, 129
 upgrading, 156–157
VNets (Azure Virtual Networks). *See* [virtual networks](#)
VPN Gateway. *See* [Azure VPN Gateway](#)
VPN Troubleshoot tool, 307
VPNs (virtual private networks). *See also* [Azure VPN Gateway](#)
 point-to-site, creating in Azure Virtual WAN, 323–324
 site-to-site
 creating in Azure Virtual WAN, 322–323
 ExpressRoute versus, 315

W

WAImportExport tool, 91–93
web apps. *See also* [App Service](#)
 backing up, 201–203
 creating, 197–198
 custom domain names, 199–201
 deploying, 206–209
 security, 198–199
webhooks, 358

Windows connections to Azure Files, [105–107](#)
Windows PowerShell Desired State Configuration (DSC) extension, [145–146](#)
Windows Subsystem for Linux (WSL), [183](#)
Windows VMs, Remote Desktop connections, [181–182](#)
workloads
 in Azure
 backups, [371–373](#)
 restoration, [374–377](#)
 on-premises
 backups, [374–383](#)
 migrating to Azure, [390](#)
 restoration, [383](#)
workspaces (Log Analytics), [340–344](#)
WSL (Windows Subsystem for Linux), [183](#)

Z

zonal services, [151](#)
zone-redundant services, [151](#)
ZRS (zone redundant storage), [70](#)

Plug into learning at

MicrosoftPressStore.com

The Microsoft Press Store by Pearson offers:

- Free U.S. shipping
- Buy an eBook, get three formats – Includes PDF, EPUB, and MOBI to use with your computer, tablet, and mobile devices
- Print & eBook Best Value Packs
- eBook Deal of the Week – Save up to 50% on featured title
- Newsletter – Be the first to hear about new releases, announcements, special offers, and more
- Register your book – Find companion files, errata, and product updates, plus receive a special coupon* to save on your next purchase

Discounts are applied to the list price of a product. Some products are not eligible to receive additional discounts, so your discount code may not be applied to all items in your cart. Discount codes cannot be applied to products that are already discounted, such as eBook Deal of the Week, eBooks that are part of a book + eBook pack, and products with special discounts applied as part of a promotional offering. Only one coupon can be used per order.



Hear about it first.

Since 1984, Microsoft Press has helped IT professionals, developers, and home office users advance their technical skills and knowledge with books and learning resources.

Sign up today to deliver exclusive offers directly to your inbox.

- New products and announcements
- Free sample chapters
- Special promotions and discounts
- ... and more!

MicrosoftPressStore.com/newsletters



Code Snippets

Many titles include programming code or configuration examples. To optimize the presentation of these elements, view the eBook in single-column, landscape mode and adjust the font size to the smallest setting. In addition to presenting code and configurations in the reflowable text format, we have included images of the code that mimic the presentation found in the print book; therefore, where the reflowable format may compromise the presentation of the code listing, you will see a “Click here to view code image” link. Click the link to view the print-fidelity code image. To return to the previous page viewed, click the Back button on your device or app.

```
$tags = @{"Environment"="Production"; "Application"="ABC123"}  
$resource = Get-AzResource -Name prodServer -ResourceGroup prodRG  
New-AzTag -ResourceId $resource.id -Tag $tags
```

```
$tags = @{"Environment"="Non-Production"; "Application"="ABC123"}  
$resource = Get-AzResource -Name prodServer -ResourceGroup prodRG  
Update-AzTag -ResourceId $resource.id -Tag $tags -Operation Replace
```

<https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{sourceResourceGroupName}/validateMoveResources?api-version=2018-05-01>

```
{  
  "resources": ["<resource-id-1>", "<resource-id-2>"],  
  "targetResourceGroup": "/subscriptions/<subscription-id>/resourceGroups/<target-group>"  
}
```

Response Code: 202
cache-control: no-cache
pragma: no-cache
expires: -1
location: <https://management.azure.com/subscriptions/<subscription-id>/operationresults/<operation-id>?api-version=2018-02-01>
retry-after: 15
...

<https://examrefstorage.blob.core.windows.net/examrefcontainer/sample-file.png>

<https://examrefstorage.blob.core.windows.net/examrefcontainer/sample-file.png?sv=2019-10-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2020-05-08T08:50:14Z&st=2020-05-08T00:50:14Z&spr=https&sig=65tNhZtj2lu0tih8HQtK7aEL9YCIpGGprZocXjiQ%2Fko%3D>

```
Set-AzStorageAccount ` 
    -ResourceGroupName "<your-resource-group-name-here>" ` 
    -Name "<your-storage-account-name-here>" ` 
    -EnableActiveDirectoryDomainServicesForFile $true ` 
    -ActiveDirectoryDomainName "<your-domain-name-here>" ` 
    -ActiveDirectoryNetBiosDomainName "<your-netbios-domain-name-here>" ` 
    -ActiveDirectoryForestName "<your-forest-name-here>" ` 
    -ActiveDirectoryDomainGuid "<your-guid-here>" ` 
    -ActiveDirectoryDomainSid "<your-domain-sid-here>" ` 
    -ActiveDirectoryAzureStorageSid "<your-storage-account-sid>"
```

```
net use <drive-letter>: \\<storage-account>.file.core.windows.net\<fileshare>
```

```
net use <drive-letter>: \\<storage-account>.file.core.windows.net\<fileshare> /  
user:Azure\<storage-account-name> <storage-account-key>
```

```
icacls <mounted-drive-letter>: /grant <user-email>:(f)
```

```
Update-AzStorageAccountADObjectPassword -RotateToKerbKey kerb2 -ResourceGroupName  
"<resource-group-name>" -StorageAccountName "<storage-account-name>"
```

```
WAImportExport.exe PrepImport /j:<JournalFile> /id:<SessionId> [/logdir:<LogDirectory>]  
[/sk:<StorageAccountKey>] /InitialDriveSet: <driveset> /DataSet:<driveset>
```

```
azcopy login --service-principal --application-id <application-id>
--tenant-id=<tenant-id>
```

```
azcopy copy "CreateUserTemplate.csv" "https://examref.blob.core.windows.net/  
destcontainer/"
```

```
azcopy copy "CreateUserTemplate.csv" "https://examref.blob.core.windows.net/  
destcontainer/?<sas token>"
```

```
azcopy copy "CreateUserTemplate.csv" "https://examref.blob.core.windows.net/  
destcontainer/?<sas token>"
```

```
azcopy copy "https://examref.blob.core.windows.net/srccontainer/" "CreateUserTemplate.csv"
```

```
AzCopy copy "https://examref.blob.core.windows.net/ srccontainer/[blob-path]?<sas token>"  
"https://examrefdest.blob.core.windows.net/destcontainer/[blob-path]?<sas token>"
```

```
azcopy sync "https://examref.blob.core.windows.net/srccontainer/?<sas token>"  
"https://examref.blob.core.windows.net/destcontainer/"
```

```
net use x \\erstandard01.file.core.windows.net\logs /u:AZURE\erstandard01  
r21Dk4qgY1HpcbriySWrBxnXnbedZLmnRK3N49Pfail1t3ragpQaIB7FqK5zbez/sMnDEzEu/dgA9Nq/W7IF4A==
```

```
cmdkey /add:<storage-account-name>.file.core.windows.net /user:AZURE\<storage-account-name> /pass:<storage-account-key>  
net use Z: \\<storage-account-name>.file.core.windows.net\<file-share-name> /persistent:yes
```

```
sudo mount -t cifs //<storage-account-name>.file.core.windows.net/logs /logs -o  
vers=3.0,username=<storage-account-name>,password=<storage-account-key>,  
dir_mode=0777,file_mode=0777,sec=ntlmssp
```

```
{  
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/  
deploymentTemplate.json#",  
    "contentVersion": "1.0.0.0",  
    "parameters": { },  
    "variables": { },  
    "functions": [ ],  
    "resources": [ ],  
    "outputs": { }  
}
```

<https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#>

<https://schema.management.azure.com/schemas/2019-04-01/subscriptionDeploymentTemplate.json#>

```
"ExamRefRGPrefix": "10.0.0.0/16",
"ExamRefRGSubnet1Name": "FrontEndSubnet",
"ExamRefRGSubnet1Prefix": "10.0.0.0/24",
"ExamRefRGSubnet2Name": "BackEndSubnet",
"ExamRefRGSubnet2Prefix": "10.0.1.0/24",
"ExamRefRGSubnet1Ref": "[concat(variables('vnetId'), '/subnets/',
variables('ExamRefRGSubnet1Name'))]",
"VNetId": "[resourceId('Microsoft.Network/virtualNetworks', variables('VirtualNetwork
Name'))]",
"VirtualNetworkName": "ExamRefVNET",
```

```
{  
    "name": "[variables('VirtualNetworkName')]",  
    "type": "Microsoft.Network/virtualNetworks",  
    "location": "[resourceGroup().location]",  
    "apiVersion": "2019-12-01",  
    "dependsOn": [],  
    "properties": {  
        "addressSpace": {  
            "addressPrefixes": [  
                "[variables('ExamRefRGPrefix')]"  
            ]  
        },  
        "subnets": [  
            {  
                "name": "[variables('ExamRefRGSubnet1Name')]",  
                "properties": {  
                    "addressPrefix": "[variables('ExamRefRGSubnet1Prefix')]"  
                }  
            },  
            {  
                "name": "[variables('ExamRefRGSubnet2Name')]",  
                "properties": {  
                    "addressPrefix": "[variables('ExamRefRGSubnet2Prefix')]"  
                }  
            }  
        ]  
    }  
}
```

```
{  
    "name": "[variables('VMNicName')]",  
    "type": "Microsoft.Network/networkInterfaces",  
    "location": "[resourceGroup().location]",  
    "apiVersion": "2019-12-01",  
    "dependsOn": [  
        "[resourceId('Microsoft.Network/virtualNetworks', 'ExamRefVNET')]"  
    ],  
    "properties": {  
        "ipConfigurations": [  
            {  
                "name": "ipconfig1",  
                "properties": {  
                    "privateIPAllocationMethod": "Dynamic",  
                    "subnet": {  
                        "id": "[variables('ExamRefRGSubnet1Ref')]"  
                    }  
                }  
            }  
        ]  
    }  
}
```

```
"privateIpAddress": "10.0.0.10",  
"privateIpAllocationMethod": "Static",
```

```
"VMPublicIPDnsName": {  
    "type": "string",  
    "minLength": 1  
}
```

"VMPublicIPName": "VMPublicIP"

```
{  
    "name": "[variables('VMPublicIPName')]",  
    "type": "Microsoft.Network/publicIPAddresses",  
    "location": "[resourceGroup().location]",  
    "apiVersion": "2019-12-01",  
    "dependsOn": [ ],  
    "properties": {  
        "publicIPAllocationMethod": "Dynamic",  
        "dnsSettings": {  
            "domainNameLabel": "[parameters('VMPublicIPDnsName')]"  
        }  
    }  
}
```

```
"dependsOn": [  
    "[resourceId('Microsoft.Network/virtualNetworks', 'ExamRefVNET')]",  
    "[resourceId('Microsoft.Network/publicIPAddresses',  
    variables('VMPublicIPName'))]"  
,
```

```
"ipConfigurations": [
    {
        "name": "ipconfig1",
        "properties": {
            "privateIPAllocationMethod": "Dynamic",
            "subnet": {
                "id": "[variables('ExamRefRGSubnet1Name')]"
            },
            "publicIPAddress": {
                "id": "[resourceId('Microsoft.Network/publicIPAddresses',
                    variables('VMPublicIPName'))]"
            },
        }
    }
]
```

```
"VMAdminUserName": {  
    "type": "string",  
    "minLength": 1  
},  
"VMAdminPassword": {  
    "type": "string",  
    "minLength": 1  
}
```

```
"VMName": "MyVM",
"VMIImagePublisher": "MicrosoftWindowsServer",
"VMIImageOffer": "WindowsServer",
"VMOSVersion": "WS2019-Datacenter",
"VMOSDiskName": "VM20SDisk",
"VMSize": "Standard_D2_v2",
"VM2IImagePublisher": "MicrosoftWindowsServer",
"VM2IImageOffer": "WindowsServer",
"VM20SDiskName": "VM20SDisk",
"VMSize": "Standard_D2_v2"
```

```
{
  "name": "[parameters('VMName')]",
  "type": "Microsoft.Compute/virtualMachines",
  "location": "[resourceGroup().location]",
  "apiVersion": "2019-12-01",
  "dependsOn": [
    "[resourceId('Microsoft.Network/networkInterfaces', variables('VMNicName'))]"
  ],
  "properties": {
    "hardwareProfile": {
      "vmSize": "[variables('vmSize')]"
    },
    "osProfile": {
      "computerName": "[variables('VMName')]",
      "adminUsername": "[parameters('VMAAdminUsername')]",
      "adminPassword": "[parameters('VMAAdminPassword')]"
    },
    "storageProfile": {
      "imageReference": {
        "publisher": "[variables('VMImagePublisher')]",
        "offer": "[variables('VMImageOffer')]",
        "sku": "[variables('VMOSVersion')]",
        "version": "latest"
      },
      "osDisk": {
        "createOption": "FromImage"
      }
    },
    "networkProfile": {
      "networkInterfaces": [
        {
          "id": "[resourceId('Microsoft.Network/networkInterfaces',
variables('VMNicName'))]"
        }
      ]
    }
  }
}
```

```
New-AzResourceGroupDeployment '  
    -Mode Complete '  
    -Name simpleVMDeployment '  
    -ResourceGroupName ExamRefRG '  
    -TemplateFile C:\ARMTemplates\deploy.json
```

```
az group deployment create \
--name simpleVMDeployment \
--mode Complete \
--resource-group ExamRefRG \
--template-file deploy.json
```

```
"imageReference": {  
    "publisher": "[variables('VMImagePublisher')]",  
    "offer": "[variables('VMImageOffer')]",  
    "sku": "[parameters('VMOSVersion')]",  
    "version": "latest"  
}
```

```
"storageProfile": {  
    "osDisk": {  
        "name": "[concat(variables('vmName'), '-osDisk')]",  
        "osType": "[parameters('osType')]",  
        "caching": "ReadWrite",  
        "image": {  
            "uri": "[parameters('vhndl')]"  
        },  
        "vhdl": {  
            "uri": "[variables('osDiskVhdName')]"  
        },  
        "createOption": "FromImage"  
    }  
}
```

```
"vhdUrl": {  
    "type": "string",  
    "metadata": {  
        "description": "VHD Url..."  
    }  
}  
"osDiskVhdName": "[concat('http://',parameters('userStorageAccountName'),  
.blob.core.windows.net/',parameters('userStorageContainerName'), '/',  
parameters('vmName'), 'osDisk.vhd')]"
```

```
$rgName = "ExamRefRG"  
$vmName = "ExamRefVM"  
Get-AzRemoteDesktopFile -ResourceGroupName $rgName -Name $vmName -Launch
```

```
$rgName = "ExamRefRG"  
$vmName = "ExamRefVM"  
$Path = "C:\Scratch\ExamRefVM.rdp"  
Get-AzRemoteDesktopFile -ResourceGroupName $rgName -Name $vmName -LocalPath $path
```

```
Set-AzVM -Redeploy -ResourceGroupName ExamRefRG -Name ExamRefVM
```

```
az vm redeploy --resource-group ExamRefRG --name ExamRefVM
```

```
az container create -g AZ104 --name az104container --image mcr.microsoft.com/azuredocs/
aci-helloworld --dns-name-label az104-test --ports 80
```

```
New-AzContainerGroup -ResourceGroupName AZ104 -Name az104container -Image mcr.microsoft.com/azuredocs/aci-helloworld -DnsNameLabel az104-test
```

```
az disk create --resource-group AZ-104 --name 104disk --size-gb 50 --query id --output tsv
```

```
az aks get-credentials --resource-group aks_rg --name aks_cluster
```

```
kubectl scale --replicas=3 deployment/az104
```

```
az aks scale --resource-group aks_rg --name aks_cluster --node-count 3
```

```
Get-AzAksCluter -ResourceGroupName aks_rg -Name aks_cluster | Set-AzAksCluster  
-NodeCount 3
```

```
kubectl autoscale deployment az104 --cpu-percent=60 --max=5 --min=2
```

```
az aks update --resource-group aks_rg --name aks_cluster --enable-cluster-autoscaler  
--max-count 5 --min-count 1
```

```
az aks get-upgrades --resource-group aks_rg --name aks_cluster
```

```
az aks upgrade --resource-group aks_rg --name aks_cluster --kubernetes-version 1.21.1
```

```
// Chart the number of reporting computers each hour
Heartbeat
| summarize dcount(ComputerIP) by bin(TimeGenerated, 1h)
| render timechart
```

Contents

Cover Page
Title Page
Copyright Page
Contents at a glance
Contents
Acknowledgments
About the Author
Introduction

Organization of this book
Preparing for the exam
Microsoft certifications
Quick access to online references
Errata, updates, & book support
Stay in touch

Chapter 1. Manage Azure identities and governance

Skill 1.1: Manage Azure Active Directory (Azure AD) objects
Skill 1.2: Manage role-based access control (RBAC)
Skill 1.3: Manage subscriptions and governance
Thought experiment
Thought experiment answers
Chapter summary

Chapter 2. Implement and manage storage

Skill 2.1: Secure Storage
Skill 2.2: Manage storage
Skill 2.3: Configure Azure Files and Azure Blob Storage
Thought experiment
Thought experiment answers
Chapter summary

Chapter 3. Deploy and manage Azure compute resources

Skill 3.1: Automate deployment of virtual machines (VMs) by using Azure Resource Manager templates

Skill 3.2: Configure VMs for high availability and scalability
Skill 3.3: Configure VMs
Skill 3.4: Create and configure containers
Skill 3.5: Create and configure Azure App Service
Thought experiment
Thought experiment answers
Chapter summary

Chapter 4. Configure and manage virtual networking

Skill 4.1: Implement and manage virtual networking
Skill 4.2: Secure access to virtual networks
Skill 4.3: Configure load balancing
Skill 4.4: Monitor and troubleshoot virtual networking
Skill 4.5: Integrate an on-premises network with an Azure virtual network
Thought experiment
Thought experiment answers
Chapter summary

Chapter 5. Monitor and back up Azure resources

Skill 5.1: Monitor resources by using Azure Monitor
Skill 5.2: Implement backup and recovery
Thought experiment
Thought experiment answers
Chapter summary

Index

Code Snippets

- i
- ii
- iii
- iv
- v
- vi
- vii
- viii
- ix
- x
- xi

xii

xiii

xiv

xv

xvi

xvii

xviii

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71

72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110

[111](#)
[112](#)
[113](#)
[114](#)
[115](#)
[116](#)
[117](#)
[118](#)
[119](#)
[120](#)
[121](#)
[122](#)
[123](#)
[124](#)
[125](#)
[126](#)
[127](#)
[128](#)
[129](#)
[130](#)
[131](#)
[132](#)
[133](#)
[134](#)
[135](#)
[136](#)
[137](#)
[138](#)
[139](#)
[140](#)
[141](#)
[142](#)
[143](#)
[144](#)
[145](#)
[146](#)
[147](#)
[148](#)
[149](#)

150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188

189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227

228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266

267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305

306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344

345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383

384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414