



Microsoft Azure Security Technologies

Exam Ref AZ-500

Yuri Diogenes
Orin Thomas

Contents

- 1. Cover Page**
- 2. Title Page**
- 3. Copyright Page**
- 4. Dedication Page**
- 5. Contents at a glance**
- 6. Contents**
- 7. About the Authors**
- 8. Acknowledgments**
- 9. Introduction**
 - 1. Organization of this book**
 - 2. Preparing for the exam**
 - 3. Microsoft certifications**
 - 4. Quick access to online references**
 - 5. Errata, updates, & book support**
 - 6. Stay in touch**
- 10. Chapter 1. Manage identity and access**
 - 1. Skill 1.1: Manage Azure Active Directory identities**
 - 2. Skill 1.2: Configure secure access by using Azure AD**
 - 3. Skill 1.3: Manage application access**
 - 4. Skill 1.4: Manage access control**
 - 5. Thought experiment answers**
 - 6. Chapter summary**
- 11. Chapter 2. Implement platform protection**
 - 1. Skill 2.1: Implement advanced network security**
 - 2. Skill 2.2: Configure advanced security for compute**
 - 3. Thought experiment answers**
 - 4. Chapter summary**
- 12. Chapter 3. Manage security operations**
 - 1. Skill 3.1: Configure security services**
 - 2. Skill 3.2: Monitor security by using Azure Security Center**
 - 3. Skill 3.3: Monitor security by using Azure Sentinel**
 - 4. Skill 3.4: Configure security policies**
 - 5. Thought experiment answers**
 - 6. Chapter summary**
- 13. Chapter 4. Secure data and applications**
 - 1. Skill 4.1: Configure security for storage**
 - 2. Skill 4.2: Configure security for databases**
 - 3. Skill 4.3: Configure and manage Key Vault**
 - 4. Thought experiment answers**
 - 5. Chapter summary**
- 14. Index**
- 15. Exam Ref AZ-500: Microsoft Azure Security Technologies**

1. List of URLs

16. Code Snippets

1. [i](#)
2. [ii](#)
3. [iii](#)
4. [iv](#)
5. [v](#)
6. [vi](#)
7. [vii](#)
8. [viii](#)
9. [ix](#)
10. [x](#)
11. [xi](#)
12. [xii](#)
13. [xiii](#)
14. [xiv](#)
15. [xv](#)
16. [xvi](#)
17. [1](#)
18. [2](#)
19. [3](#)
20. [4](#)
21. [5](#)
22. [6](#)
23. [7](#)
24. [8](#)
25. [9](#)
26. [10](#)
27. [11](#)
28. [12](#)
29. [13](#)
30. [14](#)
31. [15](#)
32. [16](#)
33. [17](#)
34. [18](#)
35. [19](#)
36. [20](#)
37. [21](#)
38. [22](#)
39. [23](#)
40. [24](#)
41. [25](#)
42. [26](#)
43. [27](#)
44. [28](#)
45. [29](#)
46. [30](#)
47. [31](#)
48. [32](#)
49. [33](#)
50. [34](#)
51. [35](#)

52. **36**
53. **37**
54. **38**
55. **39**
56. **40**
57. **41**
58. **42**
59. **43**
60. **44**
61. **45**
62. **46**
63. **47**
64. **48**
65. **49**
66. **50**
67. **51**
68. **52**
69. **53**
70. **54**
71. **55**
72. **56**
73. **57**
74. **58**
75. **59**
76. **60**
77. **61**
78. **62**
79. **63**
80. **64**
81. **65**
82. **66**
83. **67**
84. **68**
85. **69**
86. **70**
87. **71**
88. **72**
89. **73**
90. **74**
91. **75**
92. **76**
93. **77**
94. **78**
95. **79**
96. **80**
97. **81**
98. **82**
99. **83**
100. **84**
101. **85**
102. **86**
103. **87**
104. **88**
105. **89**

106. 90
107. 91
108. 92
109. 93
110. 94
111. 95
112. 96
113. 97
114. 98
115. 99
116. 100
117. 101
118. 102
119. 103
120. 104
121. 105
122. 106
123. 107
124. 108
125. 109
126. 110
127. 111
128. 112
129. 113
130. 114
131. 115
132. 116
133. 117
134. 118
135. 119
136. 120
137. 121
138. 122
139. 123
140. 124
141. 125
142. 126
143. 127
144. 128
145. 129
146. 130
147. 131
148. 132
149. 133
150. 134
151. 135
152. 136
153. 137
154. 138
155. 139
156. 140
157. 141
158. 142
159. 143

- 160. 144**
- 161. 145**
- 162. 146**
- 163. 147**
- 164. 148**
- 165. 149**
- 166. 150**
- 167. 151**
- 168. 152**
- 169. 153**
- 170. 154**
- 171. 155**
- 172. 156**
- 173. 157**
- 174. 158**
- 175. 159**
- 176. 160**
- 177. 161**
- 178. 162**
- 179. 163**
- 180. 164**
- 181. 165**
- 182. 166**
- 183. 167**
- 184. 168**
- 185. 169**
- 186. 170**
- 187. 171**
- 188. 172**
- 189. 173**
- 190. 174**
- 191. 175**
- 192. 176**
- 193. 177**
- 194. 178**
- 195. 179**
- 196. 180**
- 197. 181**
- 198. 182**
- 199. 183**
- 200. 184**
- 201. 185**
- 202. 186**
- 203. 187**
- 204. 188**
- 205. 189**
- 206. 190**
- 207. 191**
- 208. 192**
- 209. 193**
- 210. 194**
- 211. 195**
- 212. 196**
- 213. 197**

214. 198
215. 199
216. 200
217. 201
218. 202
219. 203
220. 204
221. 205
222. 206
223. 207
224. 208
225. 209
226. 210
227. 211
228. 212
229. 213
230. 214
231. 215
232. 216
233. 217
234. 218
235. 219
236. 220
237. 221
238. 222
239. 223
240. 224
241. 225
242. 226
243. 227
244. 228
245. 229
246. 230
247. 231
248. 232
249. 233
250. 234
251. 235
252. 236
253. 237
254. 238
255. 239
256. 240
257. 241
258. 242
259. 243
260. 244
261. 245
262. 246
263. 247
264. 248
265. 249
266. 250
267. 251

268. 252
269. 253
270. 254
271. 255
272. 256
273. 257
274. 258
275. 259
276. 260
277. 261
278. 262
279. 263
280. 264
281. 265
282. 266
283. 267
284. 268
285. 269
286. 270
287. 271
288. 272
289. 273
290. 274
291. 275
292. 276
293. 277
294. 278
295. 279
296. 280
297. 281
298. 282
299. 283
300. 284
301. 285
302. 286
303. 287
304. 288
305. 289
306. 290
307. 291
308. 292
309. 293
310. 294
311. 295
312. 296
313. 297
314. 298
315. 299
316. 300
317. 301
318. 302
319. 303
320. 304
321. 305

322. **306**
323. **307**
324. **308**
325. **309**
326. **310**
327. **311**
328. **312**
329. **313**
330. **314**
331. **315**
332. **316**
333. **317**
334. **318**
335. **319**
336. **320**
337. **u-1**
338. **u-2**
339. **u-3**
340. **u-4**
341. **u-5**
342. **u-6**

Exam Ref AZ-500 Microsoft Azure Security Technologies

Yuri Diogenes
Orin Thomas



Exam Ref AZ-500 Microsoft Azure

Security Technologies

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

Copyright © 2021 by Yuri Diogenes and Orin Thomas 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: 978-013-678893-5

ISBN-10: 0-136-78893-9

Library of Congress Control Number: 2020948249

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.

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

Assistant Sponsoring Editor

Charvi Arora

Development Editor

Rick Kughen

Managing Editor

Sandra Schroeder

Senior Project Editor

Tracey Croom

Copy Editor

Rick Kughen

Indexer

Cheryl Lenser

Proofreader

Charlotte Kughen

Technical Editor

Mike Martin

Editorial Assistant

Cindy Teeters

Interior Designer

Tricia Bronkella

Cover Designer

Twist Creative, Seattle

Graphics

Tammy Graham

In memory of Chris Jackson, Chief Awesomeologist at Microsoft. Chris was passionate about security, and he was always enthusiastic when he had to speak about this topic. Chris left us way too early, but his enthusiasm, leadership, and friendship will never be forgotten. Rest in peace, friend.

Contents at a glance

Introduction

Chapter 1 Manage identity and access

Chapter 2 Implement platform protection

Chapter 3 Manage security operations

Chapter 4 Secure data and applications

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 identity and access

Skill 1.1: Manage Azure Active Directory identities

Configure security for service principals

Manage Azure AD directory groups

Manage Azure AD users

Configure password writeback

Configure authentication methods including password hash and Pass

Through Authentication (PTA), OATH, and passwordless authentication

Transfer Azure subscriptions between Azure AD tenants

Skill 1.2: Configure secure access by using Azure AD

Monitor privileged access for Azure AD Privileged Identity Management (PIM)

Configure access reviews

Activate and configure PIM

Implement conditional access policies
including multifactor authentication

Administer MFA users

Configure Azure AD Identity Protection

Skill 1.3: Manage application access

Create app registrations

Configure app registration permission
scopes

Manage app registration permission
consent

Manage API access to Azure
subscriptions and resources

Skill 1.4: Manage access control

Configure subscription and resource
permissions

Configure resource group permissions

Identify the appropriate role

Apply the principle of least privilege

Configure custom RBAC roles

Interpret permissions

Check access

Thought experiment answers

Chapter summary

Chapter 2 Implement platform protection

Skill 2.1: Implement advanced network security

Overview of Azure network
components

Secure the connectivity of virtual
networks

Configure network security groups and
Application Security Groups

Create and configure Azure Firewall

Configure Azure Front Door service as an application gateway

Configure Web Application Firewall (WAF) on Azure Application Gateway

Configure Azure Bastion

Configure resource firewall

Implement service endpoint

Implement DDoS

Skill 2.2: Configure advanced security for compute

Configure endpoint security within the VM

Configure system updates for VMs in Azure

Configure authentication for containers

Configure security for different types of containers

Implement vulnerability management

Configure isolation for AKS

Configure security for container registry

Implement Azure disk encryption

Configure security for Azure App Service

Thought experiment answers

Chapter summary

Chapter 3 Manage security operations

Skill 3.1: Configure security services

Configure Azure Monitor

Create and customize alerts

Configure diagnostic logging and log retention

Monitoring security logs by using Azure Monitor

Skill 3.2: Monitor security by using Azure Security Center

Evaluate vulnerability scans from Azure Security Center

Configure Just-In-Time VM access by using Azure Security Center

Configure centralized policy management by using Azure Security Center

Configure compliance policies and evaluate for compliance by using Azure Security Center

Skill 3.3: Monitor security by using Azure Sentinel

Introduction to Azure Sentinel's architecture

Configure Data Sources to Azure Sentinel

Create and customize alerts

Configure a Playbook for a security event by using Azure Sentinel

Evaluate results from Azure Sentinel

Skill 3.4: Configure security policies

Configure security settings by using Azure Policy

Configure security settings by using Azure Blueprint

Thought experiment answers

Chapter summary

Chapter 4 Secure data and applications

Skill 4.1: Configure security for storage

Configure access control for storage accounts

Configure key management for storage accounts

Create and manage Shared Access Signatures (SAS)

Create a stored access policy for a blob or blob containers

Configure Azure AD authentication for Azure Storage

Configure Azure AD Domain Services authentication for Azure Files

Configure Storage Service Encryption

Advanced Threat Protection for Azure Storage

Skill 4.2: Configure security for databases

Enable database authentication

Enable database auditing

Configure Azure SQL Database

Advanced Threat Protection

Implement database encryption

Implement Azure SQL Database

Always Encrypted

Skill 4.3: Configure and manage Key Vault

Manage access to Key Vault

Key Vault firewalls and virtual networks

Manage permissions to secrets, certificates, and keys

Configure RBAC usage in Azure Key Vault

Manage certificates

Manage secrets

Configure key rotation

[Backup and restore of Key Vault items](#)

[Thought experiment answers](#)

[Chapter summary](#)

[Index](#)

About the Authors

Yuri Diogenes, MsC has a Master's of Science in cybersecurity intelligence and forensics investigation (UTICA College) and is a Principal Program Manager for the Microsoft CxE Azure Security Center Team. Primarily, Yuri helps customers onboard and deploy Azure Security Center and works with the ASC Engineering Team for continuous improvement of the product. Yuri has been working for Microsoft since 2006 in different positions, including five years as Senior Support Escalation Engineer for the CSS Forefront Edge Team, and from 2011 to 2017 as a member of the content development team, where he also helped create the Azure Security Center content experience after its launch in 2016. Yuri has published a total of 23 books, mostly about information security and Microsoft technologies. Yuri also holds an MBA and many IT/Security industry certifications, such as CISSP, E|CND, E|CEH, E|CSA, E|CHFI, CompTIA Security+, CySA+, Cloud Essentials Certified, Mobility+, Network+, CASP, CyberSec First Responder, MCSE, and MCTS. You can follow Yuri on Twitter at [@yuridiogenes](https://twitter.com/yuridiogenes).

Orin Thomas is a Principal Cloud Operations Advocate at Microsoft and has written more than three dozen books for Microsoft Press on topics including Windows Server, Windows Client, Azure, Microsoft 365, Office 365, System Center, Exchange Server, Security, and SQL Server. He has authored Azure Architecture courses at Pluralsight, has authored multiple Microsoft Official Curriculum and EdX courses on a variety of IT Pro topics, and is completing a Doctorate of Information Technology on cloud computing security and compliance.

at Charles Sturt University. You can follow him on
twitter at [@orinthomas](https://twitter.com/orinthomas).

Acknowledgments

The authors would like to thank Loretta Yates and the entire Microsoft Press/Pearson team for their support in this project. We would also like to thank Mike Martin (Microsoft MVP) for reviewing this book and Rick Kughen for the editorial review.

From Yuri: Thanks to my wife and daughters for their endless support; my great God for giving me strength and guiding my path on each step of the way; my friend and co-author Orin Thomas for the great partnership on this project; my manager Rebecca Halla for always encourage me to go above and beyond; and my teammates Safeena, Kerinne, Fernanda, Future, Tom, and Lior. Last but not least, thanks to my parents for working hard to give me an education, which is the foundation I use every day to keep moving forward in my career.

From Orin: Thanks to Yuri for being very supportive in this project and forgiving when life has gotten in the way of my writing schedule. I'd like to thank my son, Rooslan, for keeping his nose to the grindstone and not giving up under extraordinary conditions. I'd also like to thank the usual suspects for their support (Rick Claus, Donovan Brown, Sonia Cuff, Anthony Bartolo, Pierre Roman, Phoummala Schmitt, Sarah Lean, Thomas Maurer, and the cat that Thomas will have (or should be) buying Isidora Katanic.

Introduction

The AZ-500 exam deals with advanced topics that require candidates to have an excellent working knowledge of Azure security technologies. Portions of the exam cover topics that even experienced Azure security administrators might rarely encounter unless they work with all aspects of Azure on a regular basis. To be successful in taking this exam, candidates not only need to understand how to manage Azure identity and access, they need to understand how to implement Azure platform protection, manage Azure security operations, and secure Azure data and applications. Candidates also need to be able to keep up to date with new developments in Azure security technologies, including expanded features and changes to the interface.

Candidates for this exam should have subject matter expertise with implementing security controls and threat protection; managing identity and access; and protecting data, applications, and networks in cloud and hybrid environments as part of an end-to-end infrastructure.

Responsibilities for an Azure Security Engineer include maintaining the security posture, identifying and remediating vulnerabilities by using a variety of security tools, implementing threat protection, and responding to security incident escalations. Azure Security Engineers often serve as part of a larger team dedicated to cloud-based management and security of hybrid environments as part of an end-to-end infrastructure.

A candidate for this exam should be familiar with scripting and automation and should have a deep understanding of networking and virtualization. A candidate should also have a strong familiarity with

cloud capabilities, Azure products and services, and other Microsoft products and services. To pass, candidates require a thorough theoretical understanding of the technologies involved, as well as meaningful practical experience implementing the same.

This edition of this book covers Azure and the AZ-500 exam objectives as of late 2020. As Azure's security functionality evolves, so do the AZ-500 exam objectives, so you should check carefully to determine whether any changes have occurred since this edition of the book was authored, and you should study accordingly.

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 “Need more review?” links you’ll find in the text to find more information and take the time to research and study the topic. Great information is available on docs.microsoft.com and in blogs and forums.

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 Learn website: <http://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/ExamRefAZ500/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/ExamRefAZ500/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

[*http://www.MicrosoftPressStore.com/Support*](http://www.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*](http://support.microsoft.com)

STAY IN TOUCH

Let's keep the conversation going! We're on Twitter:

[*http://twitter.com/MicrosoftPress.*](http://twitter.com/MicrosoftPress)

Chapter 1

Manage identity and access

An important step when securing workloads is determining what traffic you'll allow and what traffic you'll block. In the past, you might use the network location and traffic type to make this determination. For example, you might allow traffic that came from a particular IP address and on a particular port and deny that traffic if it didn't meet those specific conditions. Over time, clever attackers have learned to spoof IP address information, allowing them to bypass these traditional barriers. Today, you will hear security practitioners utter the aphorism "identity is the new control plane." What the phrase means is that when the network location or traffic properties are not a great signifier of whether a host or traffic is trustworthy, the identity that is used to interact with the resource you are trying to protect might be a better guide; this is especially true if those identities are hardened with technologies such as multifactor authentication. In this chapter, you'll learn about managing identities in the cloud, securing access to resources and applications in the cloud, and managing access control to cloud administrative tools.

Skills in this chapter:

- Manage Azure Active Directory identities
- Configure secure access by using Azure AD
- Manage application access
- Manage access control

SKILL 1.1: MANAGE AZURE ACTIVE DIRECTORY IDENTITIES

This objective deals with identities within Azure Active Directory. In Azure Active Directory, identities are

represented as users, service principals, managed identities, or groups. Azure Active Directory allows you to use a variety of authentication methods including one-time passwords and multifactor authentication to secure these identities. This section covers the following topics:

- [Configure security for service principals](#)
- [Manage Azure AD directory groups](#)
- [Manage Azure AD users](#)
- [Configure password writeback](#)
- [Configure authentication methods including password hash and Pass Through Authentication \(PTA\), OATH, and passwordless authentication](#)
- [Transfer Azure subscriptions between Azure AD tenants](#)

Configure security for service principals

You configure security for a service principal when you want to control what access an application has to resources within Azure. When you register an Azure Active Directory application, the following objects will be created in your Azure Active Directory tenancy:

- **An application object** Application objects are stored within the Azure AD instance and define the application. The schema for an application object's properties is defined by the Microsoft Graph application entity resource type. Application objects are a global representation of an application across all Azure AD tenancies. The application object functions as a template from which common and default properties are determined when Azure AD creates the corresponding service principal object. Application objects have a one-to-one relationship with the software application and a one-to-many relationship with corresponding service principal objects.
- **A service principal object** A user principal in Azure AD is an object that represents a user. A service principal is an Azure AD object that represents an application. The `ServicePrincipal` object allows you to specify the access policy and permissions for the application and the user of that application within your organization's Azure AD tenant. A service principal is required for each tenancy where the application is used. A single-tenant application will only have one service principal, and a multitenant application will have a service principal for each tenancy where a user from that tenancy has consented to the application's use. The Microsoft Graph service principal entity defines the schema used for a `ServicePrincipal` object's properties. The service principal is the representation of the application in a specific Azure AD tenancy.

Registering an application with Azure AD allows you to leverage the Microsoft identity platform's secure sign-in and authorization features for use with that application. Registering an application with Azure AD requires that you provide information including the URL where the application can be accessed, the URL to forward replies after authentication occurs, and the URI that identifies your application. You will learn more about registering applications with Azure AD later in this chapter.

More Info Application and Service Principal Objects

You can learn more about application and service principal objects at <https://docs.microsoft.com/en-us/azure/active-directory/develop/app-objects-and-service-principals>.

Service principals are analogous to an on-premises Active Directory service account in that both allow an application to have an identity and security context. Service principals in Azure AD can include the following:

- A reference to an application object through the application ID property
- Local user and group application-role assignment properties
- Local user and admin application permissions
- Local policy data, including information about conditional access policies
- Data about alternate local application settings, including
 - Claims transformation rules
 - Attribute mappings (user provisioning)
 - Directory-specific app roles (when the application supports custom roles)
 - Directory-specific name or logo

Creating a service principal

As you have already learned, Azure AD will create a service principal when you register an application with an Azure AD instance. This is the way most Azure AD service principals will be created. It is possible to create a service principal with the New-

`AzADServicePrincipal` cmdlet from an Azure PowerShell session. The simplest way to run Azure

PowerShell is through a Cloud Shell session. For example, to create a new service principal named ExampleServiceprincipal, run the following command from an Azure PowerShell session.

[Click here to view code image](#)

```
$servicePrincipal = New-AzADServicePrincipal -  
    DisplayName "ExampleServiceprincipal"
```

Service principals can use two different types of authentication: password-based authentication and certificate-based authentication. If you don't specify a type of sign-in authentication when creating a service principal, password-based authentication will be used, and a random password will be assigned to the service principal account.

To view a list of service principals associated with an Azure AD instance, run the following command from an Azure PowerShell session:

[Click here to view code image](#)

```
Get-AzAdServicePrincipal | format-table
```

More Info Create Service Principal

See <https://docs.microsoft.com/en-us/powershell/azure/create-azure-service-principal-azuresps> to learn more about creating service principals.

Assigning permissions to service principals through roles

To provide access within a subscription to an application, you assign a set of permissions to the service principal associated with the application. The most straightforward way to accomplish this goal is to assign a particular role to the application. For example, if you want to give an application read access to resources within a particular resource group, you could assign the Reader role to the service principal associated with the application.

To assign a role to an application that is already registered with an Azure AD instance, perform the following steps:

1. In the Azure portal, select the subscription that the application is associated with and then from the **Subscriptions** page, select the **Access Control (IAM)** node, as shown in Figure 1-1.

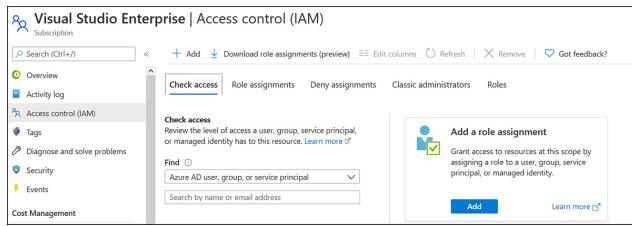


Figure 1-1 Access control (IAM) for a subscription

2. On the **Access Control (IAM)** page, select **Add A Role Assignment**, choose the role that you want to assign to the application, and choose **Azure AD User, Group, Or Service Principal** from the **Assign Access To** drop-down menu, as shown in Figure 1-2, and then in the **Select** text box, specify the name of the application.

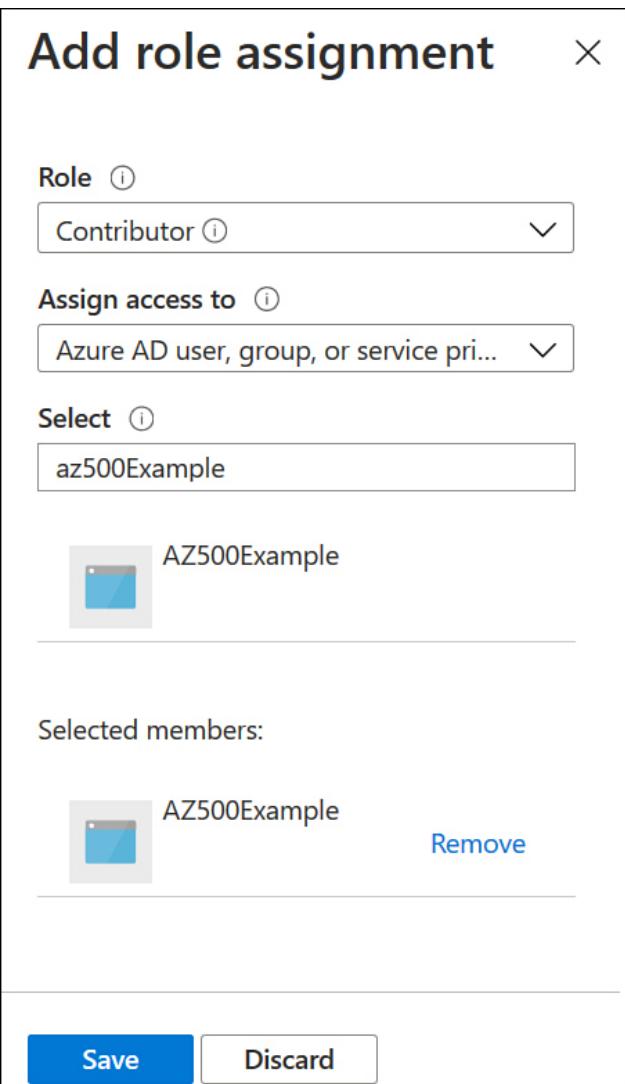


Figure 1-2 Assign a role to an application

3. Click **Save** to assign the role to the service principal.

More Info Azure Roles

You can learn more about the roles that you can assign to service principals at
<https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles>.

Just as you can assign permissions through a role through the Access Control (IAM) node at the subscription level, you can use the Access Control (IAM) node at the resource group or the resource level to assign a role to a service principal. When assigning permissions to a service principal, you should assign those

permissions in the most restrictive way possible. This means that you should only assign roles at the appropriate scope level and only assign the role needed by the application. If the application only requires reader access to a resource group, don't assign the Contributor role at the subscription level to the application's service principal.

You can use the `New-AzRoleAssignment` PowerShell cmdlet to assign a role to a service principal. For example, to create a new service principal and assign reader permissions at the subscription level to the service principal, enact the following PowerShell commands:

[Click here to view code image](#)

```
$servicePrincipal = New-AzADServicePrincipal -  
    DisplayName "ExampleServiceprincipal"  
    New-AzRoleAssignment -RoleDefinitionName "Reader"  
        -ApplicationId $servicePrincipal.  
        ApplicationId
```

Working with service principals in command-line environments requires you to use application IDs rather than the display name of the service principal. This is why the `ApplicationId` is specified in the second command in the previous example, which assigns the role to the service principal created in the first command.

You can determine what roles have been assigned to a service principal at the subscription, resource group, or resource levels by performing the following steps:

1. In the Azure portal, select the subscription, resource group, or resource to which the application is associated and then from the **Subscriptions** page, select the **Access Control (IAM)** node.
2. Select the **Role Assignments** section. This page lists all roles assigned at this scope. In the **Type** column, service principals are listed with the **App** type, as shown in Figure 1-3.

Name	Type	Role	Scope
AZ500Example	App	Contributor	This resource
Prime Admin	User	Owner	This resource

Figure 1-3 Checking Role assignments for service principals

Manage Azure AD directory groups

Groups allow you to group users and then assign them privileges and access to workloads or services. Rather than directly assigning privileges and access to workloads or services to users, you can assign these rights to a group and then indirectly assign them to users by adding the user accounts to the appropriate group. Using groups allows you to assign access and rights by adding and removing users from a group. While it's possible to assign access and rights on a per-user basis, this is administratively cumbersome and makes it challenging to determine which users have a specific right. Determining rights can be much easier to do if rights are only delegated to groups. If you only assign rights to group, if you need to determine rights, you just have to check the group membership.

You can use the Azure AD administrative console in the Azure portal to manage groups. You can access the Azure Active Directory admin center at

<https://aad.portal.azure.com> or through the Azure portal Azure AD blade. Azure AD supports two group types: security groups and Office 365 groups. Figure 1-4 shows how to select the group type when creating the group. Office 365 groups are used for collaboration between users where organizations use services such as

Microsoft 365 or Office 365. Users in groups can be internal or external to the organization.

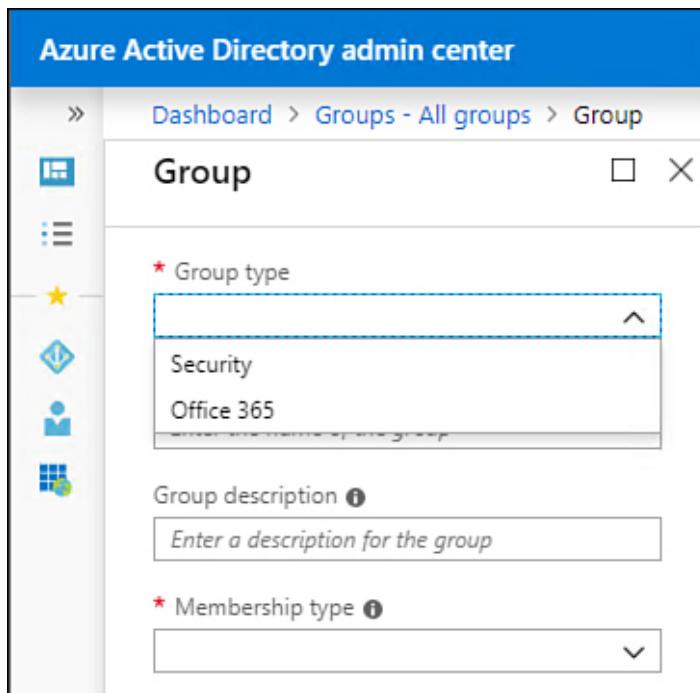


Figure 1-4 Create Azure AD Group

Group membership for security groups must be assigned and is not dynamic. When a group's membership is assigned, members are added and removed manually by administrators or other users who have the appropriate rights.

Office 365 group types can be configured as assigned or dynamic. When the dynamic option is selected, group membership is determined based on the results of a query against user or device attributes. For example, with Office 365 groups, you can have group membership determined by user attributes such as location or manager. Figure 1-5 shows an Office 365 group with dynamic membership, where users who have the department attribute set to Marketing will automatically be assigned membership of the group.

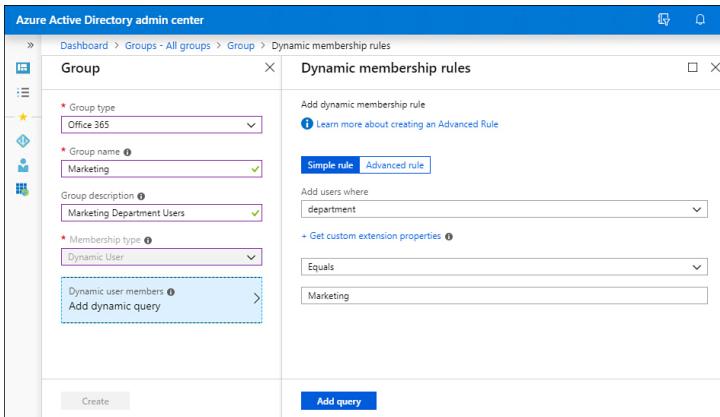


Figure 1-5 Office 365 dynamic group membership

You can use the following PowerShell commands from the Azure AD PowerShell module to manage Azure AD Groups:

- **Get-AzureADGroup** Provides information about Azure AD Groups.
- **New-AzureADGroup** Creates a new Azure AD Group.
- **Set-AzureADGroup** Configures the properties of an Azure AD Group.
- **Remove-AzureADGroup** Removes an Azure AD Group.
- **Add-AzureADGroupMember** Adds a user to an Azure AD Group.
- **Remove-AzureADGroupMember** Removes a user from an Azure AD Group.
- **Add-AzureADGroupOwner** Adds a user as an owner of an Azure AD Group. Gives the user limited group management privileges.
- **Remove-AzureADGroupOwner** Removes a user as owner of an Azure AD Group.

More Info Azure AD Groups

You can learn more about Azure AD Groups at <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-groups-view-azure-portal>.

Creating groups

To create an Azure AD group, perform the following steps:

1. In the Azure portal, select the **Azure Active Directory** menu blade.

2. Under **Manage** in the **Azure Active Directory** menu blade, select **Groups**, as shown in Figure 1-6.

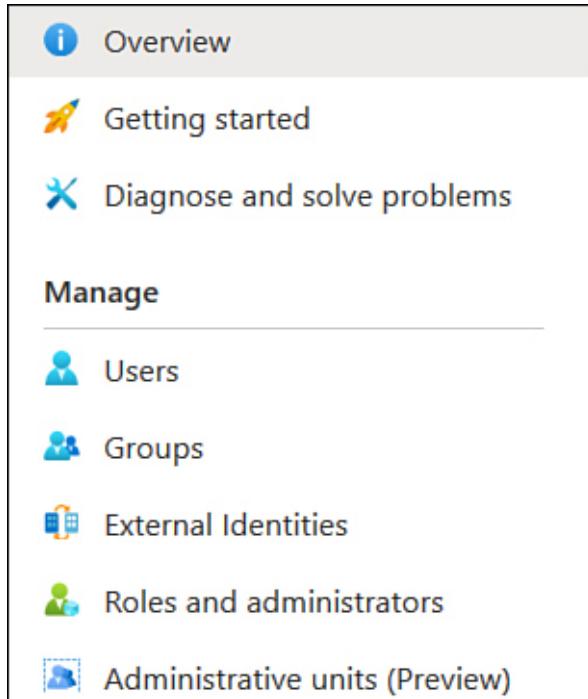


Figure 1-6 Azure Active Directory menu blade

3. On the **Groups** page control bar, click **New Group**.
4. On the **New Group** page shown in Figure 1-7, provide the following information and select **Create**:
 1. **Group Type** Choose between **Security** and **Office 365**.
 2. **Group Name** Provide a name for the group. It is often a good idea to come up with a system for naming groups, rather than naming the group based on whatever comes to mind when filling out the form. Use this system for all groups in the subscription. One strategy is to name groups in a way that indicates how they collect accounts, such as `Research Users` for user accounts related to research. Group names need to be unique within an Azure Active Directory instance.
 3. **Group Description** Provide a meaningful description for the group. This description should be meaningful enough that if you won the lottery and retired to Tahiti, the person who replaced you could understand the purpose of the group.
 4. **Membership Type** If you choose a **Security** group, group members must be added manually. If you choose the **Office 365** group type, you will have the following options:

- Owners** Users designated as group owners can modify the membership of the group.
- Members** Allows you to specify group membership. Can include users, groups, service principals, and managed identities.

The screenshot shows the 'New Group' page in the Azure portal. It includes fields for Group type (Security), Group name (Research Users), Group description (Research Users Group), Membership type (Assigned), Owners (1 owner selected), Members (3 members selected), and a Create button.

Figure 1-7 New Group page

You can create Azure Groups from a Cloud Shell session using the `az ad group create` command. For example, to create a group named *Accounting Users*, use the following command:

[Click here to view code image](#)

```
Az ad group create --display-name "Accounting Users" --mail-nickname "accounting.users"
```

More Info Creating Groups

You can learn more about topic at <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-manage-groups>.

Adding and removing group members

You can add members to an Azure AD group from a Cloud Shell session using the `az ad group member add` command. The challenge when using this command is that you must specify the member using the object ID of the member, rather than the member name. For

example, to add the user with the object ID ac5ebbf2-22c7-4381-b91d-12aeb3093413 to the group Accounting Users, use the following command from an Azure PowerShell session:

[Click here to view code image](#)

```
az ad group member add --group "Accounting Users"  
--member-id ac5ebbf2-22c7-4381-b91d-  
12aeb3093413
```

You can determine the object ID of a user by using the `az ad user show` command and specifying the user's user principal name with the `ID` parameter. For example, to determine the object ID of the user delta.user@tailwindtraders.net, run the following command in Cloud Shell:

[Click here to view code image](#)

```
az ad user show --id  
delta.user@tailwindtraders.net
```

Nested groups

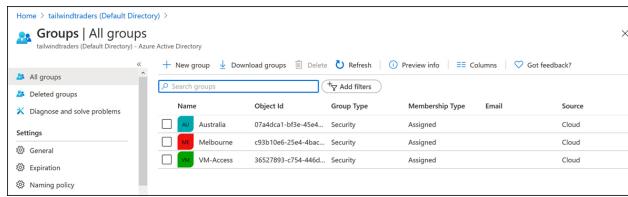
Azure AD allows you to add a security group as a member of another security group, which is known as a nested group. When you do this, the member group will inherit the attributes and properties of the parent group. Nesting groups allows you to further simplify management of large numbers of users. For example, you might have groups for the managers in Melbourne, Sydney, and Adelaide. You could add these three groups to an Australian Managers group and then assign top-level group rights and permissions to Australian Managers, rather than assigning those rights to each city-level Managers group. This also provides you with flexibility should you add additional city-level managers groups, such as Brisbane and Perth, at some point in the future because you'd just add these groups to the Australian Managers group to assign the same permissions.

At the time of writing, Azure AD does not support the following nesting scenarios:

- Adding an Azure AD group to a group synchronized from on-premises Active Directory
- Adding Azure AD security groups to Office 365 groups
- Adding Office 365 to groups other than other Office 365 groups
- Assigning apps to nested groups
- Assigning licenses to nested groups
- Nesting distribution groups

To nest groups using the Azure portal, perform the following steps:

1. On the **Groups – All Groups** page of the Azure Active Directory blade of the Azure portal, click the group that you want to nest. This will open the group's properties, as shown in Figure 1-8. In this example, the `Melbourne` group will be added to the `Australia` group.



The screenshot shows the 'Groups - All groups' page in the Azure Active Directory portal. The left sidebar includes 'All groups', 'Deleted groups', 'Diagnose and solve problems', 'Settings' (with options for General, Expiration, and Naming policy), and a 'New group' button. The main area displays a table with three rows:

Name	Object Id	Group Type	Membership Type	Email	Source
Australia	074ddca1-bf3e-45e4...	Security	Assigned		Cloud
APAC	c93b10ef-23ed-4bce...	Security	Assigned		Cloud
VM-Access	36527893-c754-446d...	Security	Assigned		Cloud

Figure 1-8 List of Azure AD groups

2. Click the Group Memberships item in the Manage section of the group's properties, as shown in Figure 1-9.

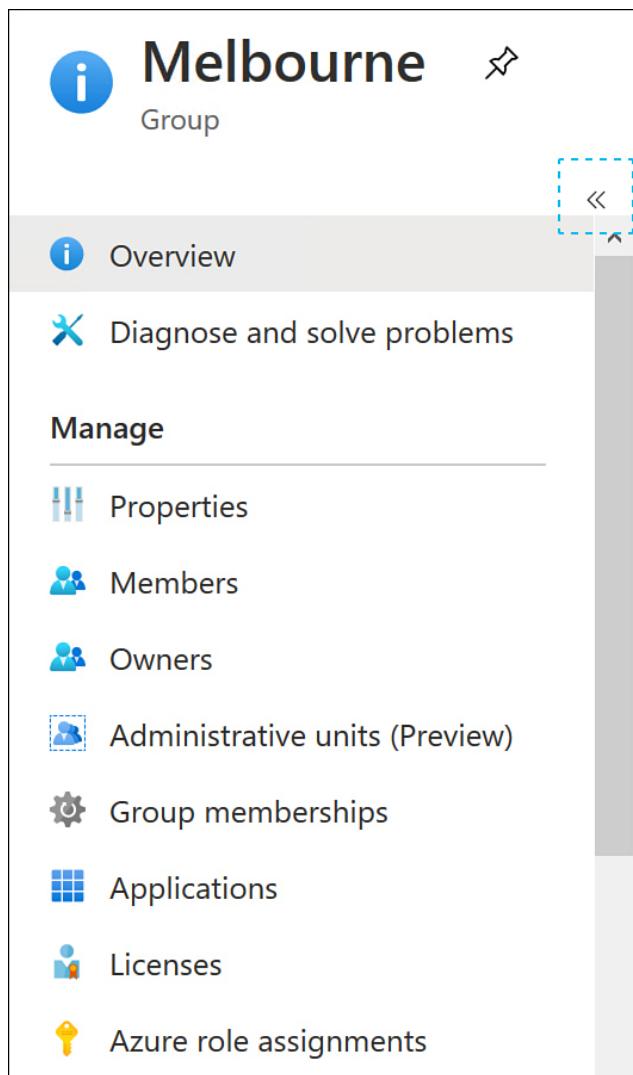


Figure 1-9 Group memberships listed in the Groups menu

3. On the **Group Memberships** page, click **Add Memberships**.
4. On the **Select Groups** page, select the group that you want to nest the group within. In this case, we will select the **Australia** group, as shown in Figure 1-10. Click **Select** to nest the group. A group can be nested within multiple groups.

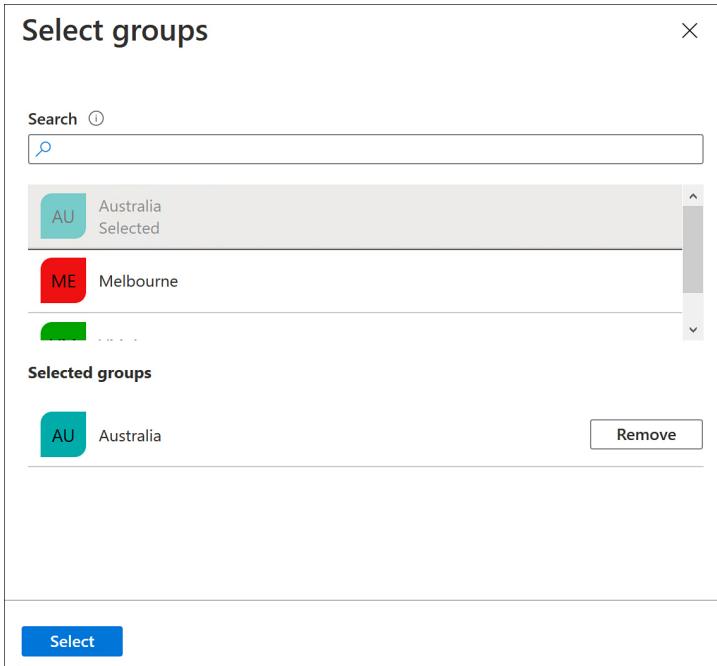


Figure 1-10 Selecting a group to nest

To remove a group from another group, open the parent group's group membership page and then remove the nested group by selecting that group and clicking **Remove Memberships**.

More Info Nesting Groups

You can learn more about this topic at <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-groups-membership-azure-portal>.

Manage Azure AD users

You can use the Azure AD Admin Center in the Azure portal, Azure PowerShell, or the Microsoft 365 admin center to manage Azure AD user accounts. The Azure AD admin center gives you a greater set of options for managing the properties of user accounts than does the Microsoft 365 admin center because you can edit extended user properties, as shown in [Figure 1-11](#).

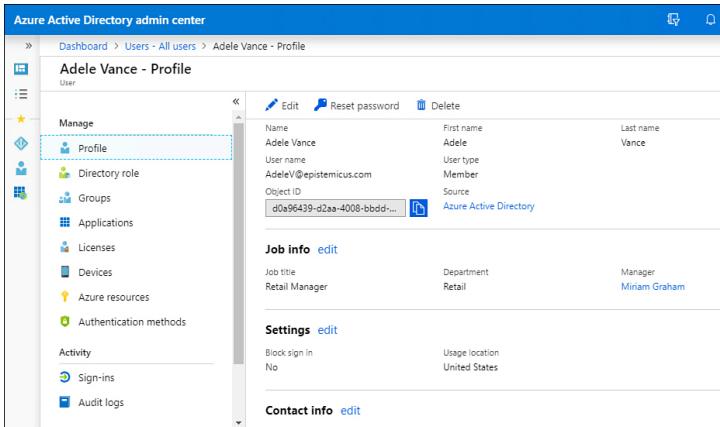


Figure 1-11 User properties page

To create a new Azure AD User, perform the following steps:

1. In the Azure AD console, select **Users—All Users** and then click **New User**.
2. On the **New User** blade shown in Figure 1-12, provide the following information:
 1. **Name** The user's actual name.
 2. **User Name** The user's sign-in name in UPN format.
 3. **Profile** The user's first name, last name, job title and department.
 4. **Properties** This specifies the source of authority for the user. By default, if you are creating the user using the Azure AD admin center or the Microsoft 365 admin center, the source of authority will be Azure Active Directory.
 5. **Groups** This defines which groups the user should be a member of.
 6. **Directory Role** Choose whether the account has User, Global Administrator, or a Limited Administrator role.
 7. **Password** This is the automatically generated password. With the **Show Password** option, you can transmit the password to the user through a secure channel.

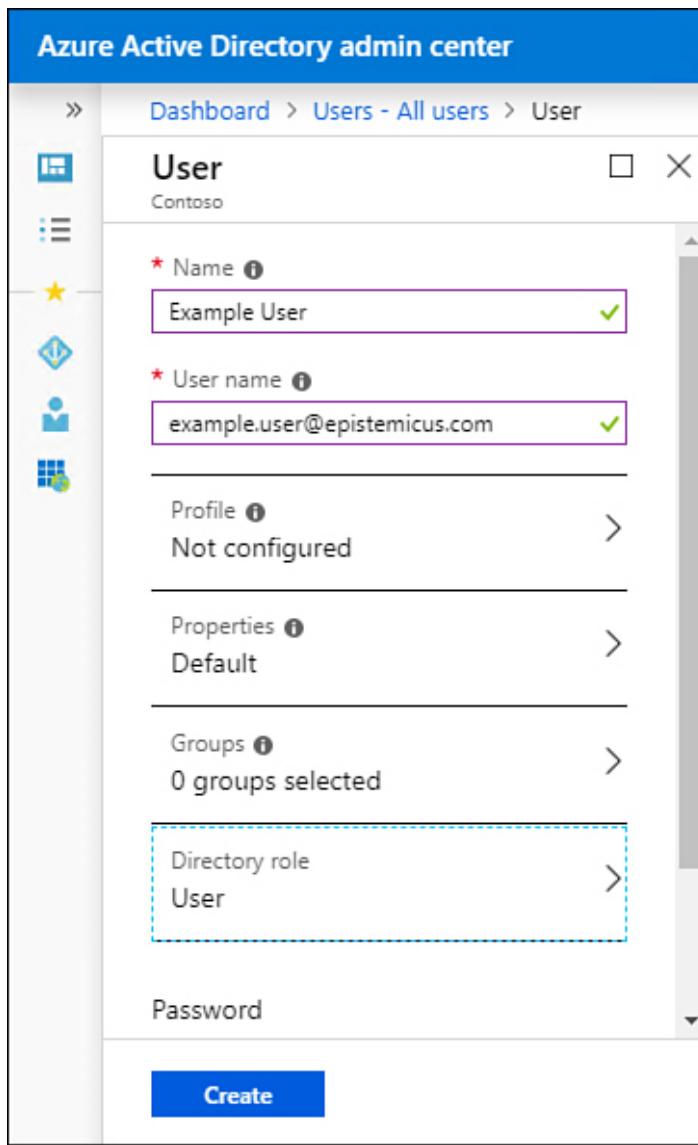


Figure 1-12 New User properties page

You can also use the Azure AD admin center to perform the following user administration tasks:

- Update profile information
- Assign directory roles
- Manage group membership
- Manage licenses
- Manage devices
- Manage access to Azure resources
- Manage authentication methods

When you delete a user from Azure AD, the account remains in the Azure Active Directory Recycle Bin for 30 days. This means that you can recover the account online should it be necessary to do so. If you delete a user from your on-premises Active Directory environment but have enabled the on-premises Active Directory Recycle Bin, recovering the user from the on-premises Active Directory Recycle Bin will recover the user account in Microsoft 365. If you don't have the Active Directory Recycle Bin enabled, you will need to create another account with a new GUID.

More Info Creating Azure AD Users

You can learn more about Azure AD PowerShell cmdlets for managing users at <https://docs.microsoft.com/en-us/powershell/azure/active-directory/new-user-sample>.

Configure password writeback

Password writeback occurs when a user uses self-service password (SSPR) functionality to update his or her password in Azure and that updated password is then written to an on-premises Active Directory Domain Services instance. Azure AD also supports SSPR on Azure AD native accounts where no writeback to an on-premises instance is necessary. To implement SSPR for organizations with on-premises Active Directory Domain Services, you need to first install Azure AD Connect to synchronize on-premises identities to Azure.

More Info Password Writeback

You can learn more about password writeback at <https://docs.microsoft.com/en-us/azure/active-directory/authentication/tutorial-enable-sspr-writeback>.

Install and configure Azure AD Connect

Azure AD Connect allows you to connect your on-premises Active Directory accounts with an Azure AD instance. This is useful not only for applications running in Azure, but it allows you to implement single sign-on if your organization is using Microsoft 365 or Office 365.

Single sign-on allows you to use one identity to access on-premises and cloud resources. In many scenarios, the user won't even be required to reauthenticate.

Azure AD Connect is software that you install on a computer that manages the process of synchronizing objects between the on-premises Active Directory and the Azure Active Directory instance. You can install Azure AD Connect on computers running the Windows Server 2012 or later operating systems:

Azure AD Connect has the following requirements:

- It must be installed on a Windows Server instance that has the GUI version of the operating system installed. You cannot install Azure AD connect on a computer running the Server Core operating system.
- You can deploy Azure AD Connect on a computer that is either a domain controller or a member server. If you use the custom options, a standalone server can be used.
- The server hosting Azure AD Connect requires .NET Framework 4.5.1 or later.
- The server hosting Azure AD Connect requires Microsoft PowerShell 3.0 or later.
- The server hosting Azure AD Connect must not have PowerShell Transcription enabled through Group Policy.
- If you are deploying Azure AD Connect with Active Directory Federation Services, you must use Windows Server 2012 R2 or later for the Web Application Proxy, and Windows remote management must be enabled on the servers that will host AD FS roles.
- If global administrators will have multifactor authentication enabled (MFA), then the URL <https://secure.aadcdn.microsoftonline-p.com> must be configured as a trusted site.

Connectivity requirements

The computer with Azure AD Connect installed must be a member of a domain in the forest that you want to synchronize, and it must have connectivity to a writable domain controller in each domain of the forest you want to synchronize on the following ports:

- **DNS** TCP/UDP port 53
- **Kerberos** TCP/UDP port 88

- **RPC** TCP port 135
- **LDAP** TCP/UDP port 389
- **TLS/SSL** TCP port 443
- **SMB** TCP port 445

The computer with Azure AD Connect installed must be able to establish communication with the Microsoft Azure servers on the Internet over TCP port 443. The computer with Azure AD Connect installed can be located on an internal network as long as it can initiate communication on TCP port 443. The computer hosting Azure AD Connect does not need a publicly routable IP address. The computer hosting Azure AD Connect always initiates synchronization communication to Microsoft Azure. Microsoft Azure Active Directory does not initiate synchronization communication to the computer hosting Azure AD Connect on the on-premises network.

Because the Azure AD Connect instance requires access to the Internet, you should not install Azure AD Connect on a domain controller. If you are going to be replicating more than 50,000 objects, Microsoft recommends that you deploy SQL Server on a computer that is separate from the computer that will host Azure AD Connect. If you plan to host the SQL Server instance on a separate computer, ensure that communication is possible between the computer hosting Azure AD Connect and the computer hosting the SQL Instance on TCP port 1433.

If you are going to use a separate SQL Server instance, ensure that the account used to install and configure Azure AD Connect has systems administrator rights on the SQL instance and that the service account used for Azure AD Connect has public permissions on the Azure AD Connect database.

SQL Server requirements

When you deploy Azure AD connect, you have the option of having Azure AD Connect install an SQL Server Express instance, or you can choose to have Azure AD

Connect leverage a full instance of SQL Server. SQL Server Express is limited to a maximum database size of 10 GB. In terms of Azure AD Connect, this means that Azure AD Connect is only able to manage 100,000 objects. This is likely to be adequate for all but the largest environments.

For environments that require Azure AD Connect to manage more than 100,000 objects, you'll need to have Azure AD Connect leverage a full instance of SQL Server. Azure AD Connect can use all versions of Microsoft SQL Server, from Microsoft SQL Server 2012 with the most recent service pack to SQL Server 2019. It is important to note that SQL Azure is not supported as a database for Azure AD Connect. If you are deploying a full instance of SQL Server to support Azure AD Connect, ensure that the following prerequisites are met:

- **Use a case-insensitive SQL collation** Case insensitive collations have the _CI_ identifier included in their names. Case sensitive collations (those that use the _CS_ designation) are not supported for use with Azure AD Connect.
- **You can only use one sync engine per SQL instance** If you have an additional Azure AD Connect sync engine or if you are using Microsoft Identity Manager in your environment, each sync engine requires its own separate SQL instance.

Requirements for deployment accounts

You use two accounts when configuring Azure AD Connect. One account must have specific Azure AD permissions; the other account must have specific on-premises Active Directory permissions. The accounts that you use to install and configure Azure AD Connect have the following requirements:

- The account used to configure Azure AD Connect must have Global Administrator privileges in the Azure AD tenancy. You should create a separate account for this task and configure the account with a complex password that does not expire. This account is used for the synchronization process between on-premises AD and Azure AD.
- The account used to install and configure Azure AD Connect must have Enterprise Administrator permissions within the on-premises Active Directory forest if you will be using Express installation settings. This account is only required during

installation and configuration. Once Azure AD Connect is installed and configured, this account no longer needs Enterprise Administrator permissions. The best practice is to create a separate account for Azure AD Connect installation and configuration and to temporarily add this account to the Enterprise Admins group during the installation and configuration process. Once Azure AD Connect is installed and configured, this account can be removed from the Enterprise Admins group. You should not attempt to change the account used after Azure AD Connect is set up and configured because Azure AD Connect always attempts to run using the original account.

- The account used to install and configure Azure AD Connect must be a member of the local Administrators group on the computer on which Azure AD Connect is installed.

Installing Azure AD Connect

Installing Azure AD Connect with Express settings is appropriate if your organization has a single Active Directory forest and you want to use password synchronization for authentication. The Azure AD Connect Express settings are appropriate for most organizations. You can download the Azure AD Connect installation files from Microsoft's download center website.

To install Azure AD Connect with Express settings, perform the following steps:

1. Double click the `AzureADConnect.msi` file that you've downloaded from the Microsoft download center. You will be prompted with a security warning. After clicking **Run**, Azure AD Connect will be installed on your computer. When the installation is complete, you will be presented with a splash screen detailing the license terms and displaying a privacy notice. You'll need to agree to these terms before clicking **Continue**.
2. If your organization has an internal nonroutable domain, it will be necessary for you to use custom settings. The best practice is to use domain synchronization when your on-premises Active Directory instance and your Azure Active Directory instance use the same routable domain name. Click **Continue**.
3. On the **Install Required Components** page, shown in Figure 1-13, choose between the following options:

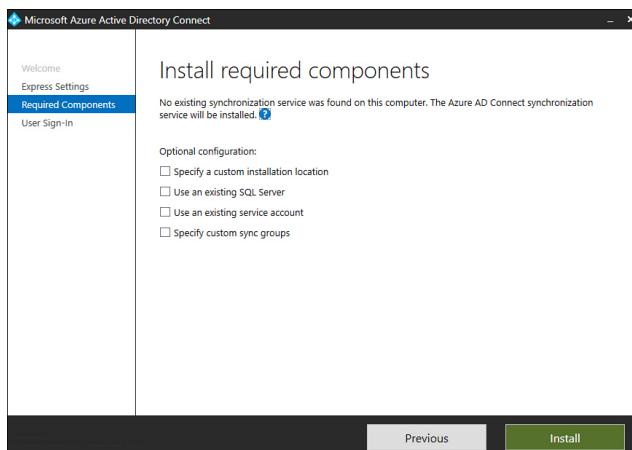


Figure 1-13 Install Required Components page

1. **Specify A Custom Installation Location** Choose this option if you want to install Azure AD Connect in a separate location, such as on another volume.
2. **Specify An Existing SQL Server** Choose this option if you want to specify an alternate SQL server instance. By default, Azure AD Connect will install an SQL Server Express instance.
3. **Use An Existing Service Account** You can configure Azure AD Connect to use an existing service account. By default, Azure AD Connect will create a service account. You can configure Azure AD Connect to use a Group Managed Service account. You'll need to use an existing service account if you are using Azure AD Connect with a remote SQL Server instance or if communication with Azure will occur through a proxy server that requires authentication.
4. **Specify Custom Sync Groups** When you deploy Azure AD Connect, it will create four local groups on the server that hosts the Azure AD Connect Instance. These groups are the Administrators group, Operators group, Password Reset group, and the Browse group. If you want to use your own set of groups, you can specify them here. These groups must be local to the host server and not a member of the domain.
4. Once you have specified which custom options you require—and you aren't required to choose any—click **Install**.
5. On the **User Sign-In** page shown in Figure 1-14, specify what type of sign in you want to allow. You can choose between the following options, the details of which were covered earlier in this chapter:
 1. Password Synchronization
 2. Pass-Through Authentication
 3. Federation With AD FS
 4. Federation With PingFederate

5. Do Not Configure

6. Enable Single Sign-On

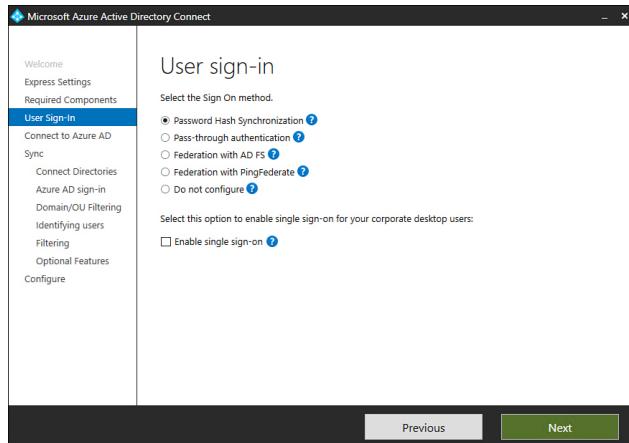


Figure 1-14 User Sign-In options page

Most organizations will choose **Password Synchronization** because this is the most straightforward option.

- 6.** On the **Connect To Azure AD** page, provide the credentials of an account with Global Administrator privileges in Azure AD. Microsoft recommends you use an account in the default `onmicrosoft.com` domain associated with the Azure AD instance to which you will be connecting. If you choose the **Federation With AD FS** option, ensure that you do not sign in using an account in a domain that you will enable for federation. Figure 1-15 shows a sign-in with a **Password Synchronization** scenario.

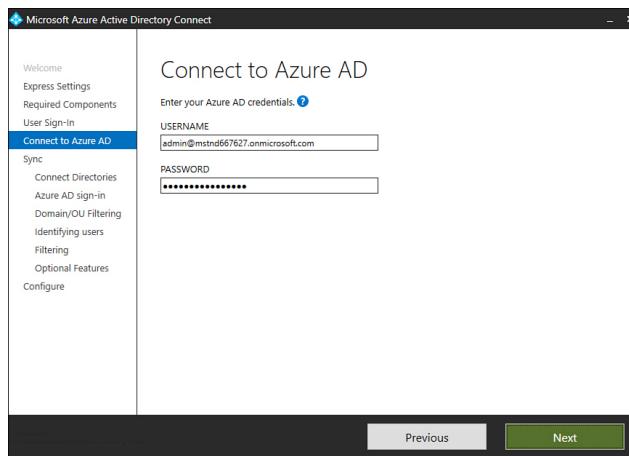


Figure 1-15 Connect to Azure AD page

- 7.** Once Azure AD Connect has connected to Azure AD, you will be able to specify the directory type to synchronize, as well as the forest. Click **Add Directory** to add a specific forest. When you add a forest by clicking **Add Directory**, you will need to specify

the credentials of an account that will perform periodic synchronization. Unless you are certain that you have applied the minimum necessary privileges to an account, you should provide Enterprise Administrator credentials and allow Azure AD Connect to create the account, as shown in Figure 1-16. This will ensure that the account is only assigned the privileges necessary to perform synchronization tasks.

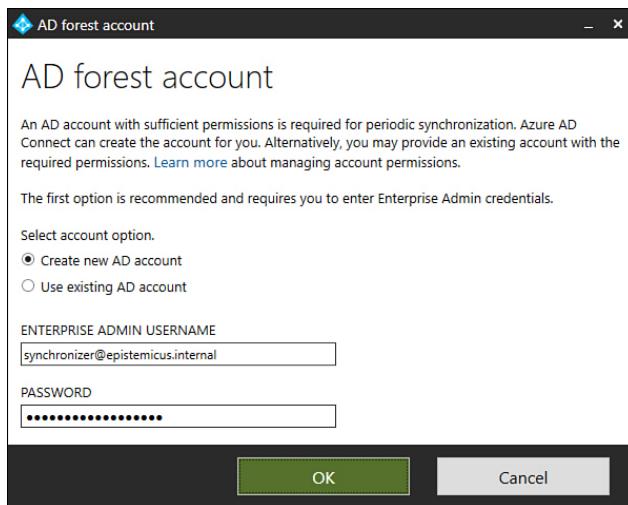


Figure 1-16 AD Forest Account page

8. Once the credentials have been verified, as shown in Figure 1-17, click **Next**.

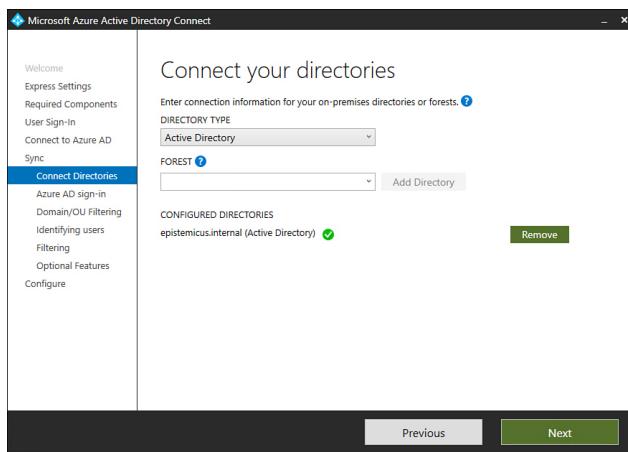


Figure 1-17 Connect Your Directories page

9. On the **Azure AD Sign-In Configuration** page, shown in Figure 1-18, review the UPN suffix and then inspect the on-premises attribute to use as the Azure AD username. You'll need to ensure that accounts use a routable Azure AD username.

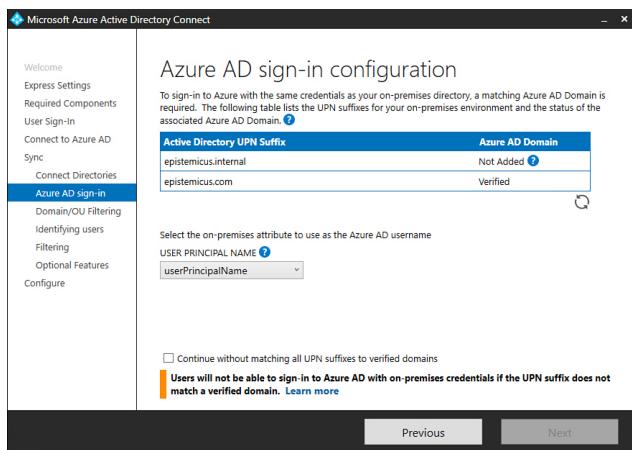


Figure 1-18 Azure AD Sign-In Configuration page

10. On the **Domain And OU Filtering** page, select whether you want to sync all objects or just objects in specific domains and OUs.
11. On the **Uniquely Identifying Users** page shown in Figure 1-19, specify how users are to be identified. By default, users should only have one representation across all directories. If users exist in multiple directories, you can have matches identified by a specific active directory attribute, with the default being the **Mail Attribute**.

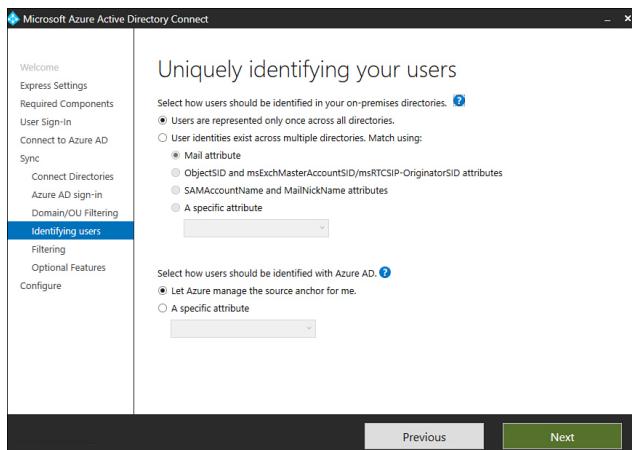


Figure 1-19 Uniquely Identifying Your Users page

12. On the **Filter Users And Devices** page, specify whether you want to synchronize all users and devices or only members of a specific group. Figure 1-20 shows members of the Microsoft 365-Pilot-Users group being configured so that their accounts will be synchronized with Azure.

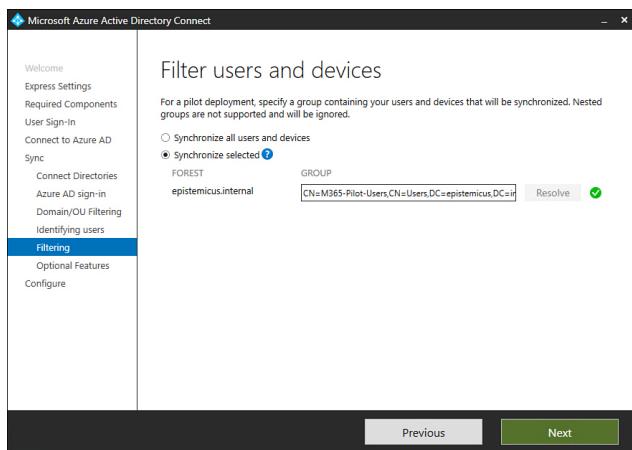


Figure 1-20 Filter Users And Devices page

13. On the **Optional Features** page shown in Figure 1-21, select any optional features that you want to configure. These features include the following:

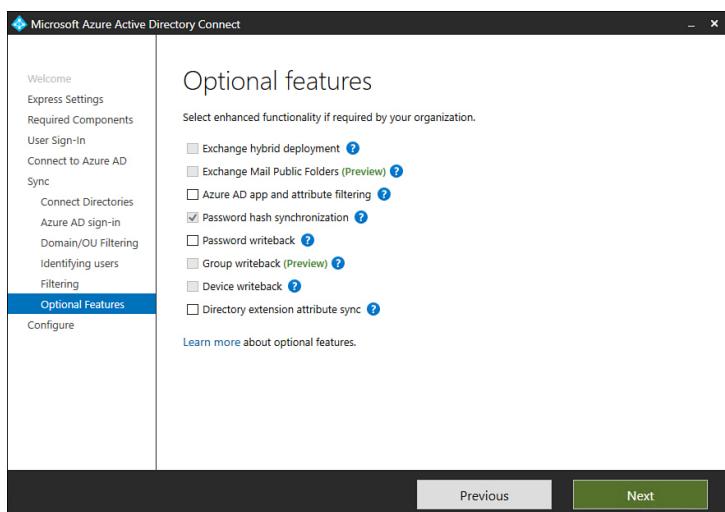


Figure 1-21 Optional Features page

- **Exchange Hybrid Deployment** This option is suitable for organizations that have an Office 365 deployment and where there are mailboxes hosted both on-premises and in the cloud.
- **Exchange Mail Public Folders** This feature allows organizations to synchronize mail-enabled public folder objects from an on-premises Active Directory environment to Microsoft 365.
- **Azure AD App And Attribute Filtering** Selecting this option gives you the ability to be more selective about which attributes are synchronized between the on-premises environment and Azure AD.
- **Password Synchronization** Synchronizes a hash of the user's on-premises password Azure AD. When the user authenticates to Azure AD, the submitted password is hashed using the same

process, and if the hashes match, the user is authenticated. Each time a user updates his or her password on-premises, the updated password hash synchronizes to Azure AD.

- **Password Writeback** Password writeback allows users to change their passwords in the cloud and have the changed password written back to the on-premises Active Directory instance.
- **Group Writeback** Changes made to groups in Azure AD are written back to the on-premises AD instance.
- **Device Writeback** Information about devices registered by the user in Azure AD is written back to the on-premises AD instance.
- **Directory Extension Attribute Sync** Allows you to extend Azure AD schema based on extensions made to your organization's on-premises Active Directory instance.

On the **Ready To Configure** page, you can choose to start synchronization or enable staging mode. When you configure staging mode, Azure AD Connect will prepare the synchronization process, but it will not synchronize any data with Azure AD.

Using UPN suffixes and nonroutable domains

Prior to performing synchronization between an on-premises Active Directory environment and an Azure Active Directory instance, you must ensure that all user account objects in the on-premises Active Directory environment are configured with a value for the UPN suffix that can function for both the on-premises environment and any application that you want to use it with in the cloud. This is not a problem when an organization's internal Active Directory domain suffix is a publicly routable domain. For example, a domain name, such as `contoso.com` or `adatum.com`, which is resolvable by public DNS servers, will suffice. Things become more complicated when the organization's internal Active Directory domain suffix is not publicly routable.

If a domain is nonroutable, the default Azure AD instance domain, such as `adatum2020.onmicrosoft.com`, should be used for the UPN suffix. This requires modifying the UPN suffix of accounts stored in the on-premises Active Directory

instance. Modification of UPN after initial synchronization has occurred is not supported. So, you need to ensure that on-premises Active Directory UPNs are properly configured prior to performing initial synchronization using Azure AD Connect. Perform the following steps to add a UPN suffix to the on-premises Active Directory if the Active Directory domain uses a nonroutable namespace:

1. Open the **Active Directory Domains And Trust** console and select **Active Directory Domains And Trusts**.
2. On the **Action** menu, click **Properties**.
3. On the **UPN Suffixes** tab, enter the UPN suffix to be used with Azure Active Directory. Figure 1-22 shows the UPN suffix of `epistemicus.com`.

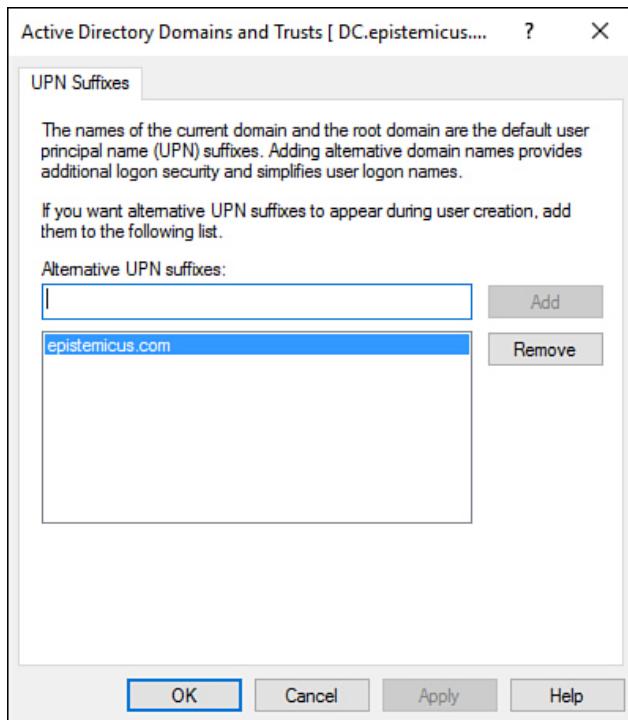


Figure 1-22 Configuring the UPN suffix for a routable domain

4. Once the UPN suffix has been added in the **Active Directory Domains And Trusts dialog box**, you can assign the UPN suffix to user accounts. You can do this manually, as shown in Figure 1-23, by using the **Account** tab of the user's **Properties** dialog box.

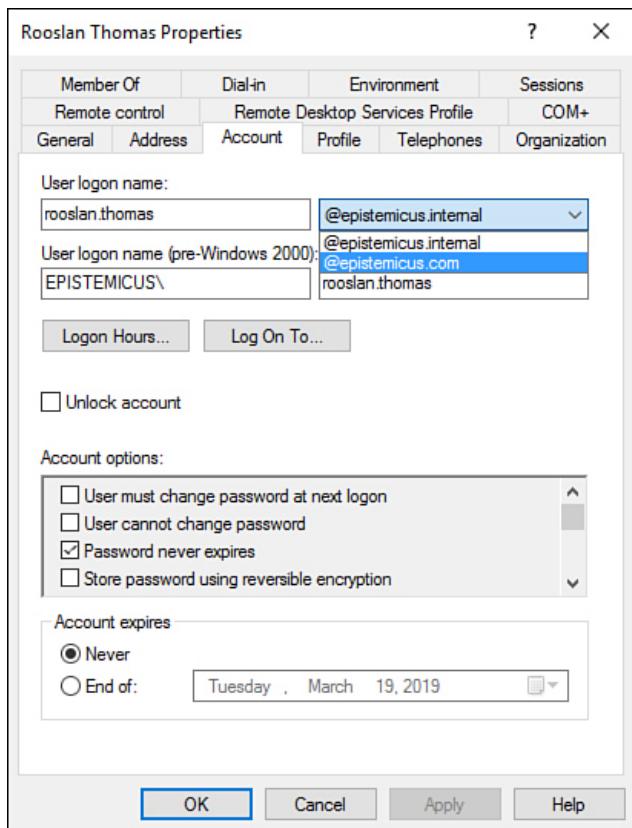


Figure 1-23 Configure UPN

5. You can also use Microsoft PowerShell scripts to reset the UPNs of multiple user accounts. For example, the following script resets UPN suffixes of all user accounts in the `epistemicus.internal` domain to `epistemicus.onmicrosoft.com`.

[Click here to view code image](#)

```
Get-ADUser -Filter {UserPrincipalName -  
like "*@epistemicus.internal"} -SearchBase  
"DC=epistemicus,DC=internal" |  
ForEach-Object {  
$UPN =  
$_ .UserPrincipalName.Replace("epistemicus.internal","epistemicus.onmicrosoft.com")  
  
Set-ADUser $_ -UserPrincipalName $UPN  
}
```

Sign-in options

Azure AD Connect supports a variety of sign-in options. You configure which one you want to use when setting up Azure AD Connect. The default method, Password Synchronization, is appropriate for most organizations.

that will use Azure AD Connect to synchronize identities to the cloud.

Password synchronization

Hashes of on-premises Active Directory user passwords synchronize to Azure AD, and changed passwords immediately synchronize to Azure AD. Actual passwords are never sent to Azure AD and are not stored in Azure AD. This allows for a seamless single sign-on for users of computers that are joined to an Active Directory domain that synchronizes to Azure AD. Also, password synchronization allows you to enable password write-back for self-service password reset functionality through Azure AD.

Pass-through authentication

When authenticating to Azure AD, the user's password is validated against an on-premises Active Directory domain controller. Passwords and password hashes are not present in Azure AD. Pass-through authentication allows for on-premises password policies to apply. Pass-through authentication requires that Azure AD Connect have an agent on a computer joined to the domain that hosts the Active Directory instance that contains the relevant user accounts. Pass-through authentication also allows seamless single sign-on for users of domain joined machines.

With pass-through authentication, the user's password is validated against the on-premises Active Directory controller. The password doesn't need to be present in Azure AD in any form. This allows for on-premises policies, such as sign-in hour restrictions, to be evaluated during authentication to cloud services.

Pass-through authentication uses a simple agent on a Windows Server 2012 R2, Windows Server 2016, or Windows Server 2019 domain-joined machine in the on-premises environment. This agent listens for password

validation requests. It doesn't require any inbound ports to be open to the Internet.

In addition, you can also enable single sign-on for users on domain-joined machines that are on the corporate network. With single sign-on, enabled users only need to enter a username to help them securely access cloud resources.

Active Directory Federation

This allows users to authenticate to Azure AD resources using on-premises credentials. When you choose the Federation with AD FS option, Active Directory Federation Services is installed and configured; also, a Web Application Proxy server to facilitate communication between the on-premises AD FS deployment and Microsoft Azure Active Directory is installed. This is the most complicated identity synchronization configuration, and it is only likely to be implemented in environments with complicated identity configurations.

More Info Azure AD Connect Sign-In Options

You can learn more about sign-in options by consulting the following article: <https://docs.microsoft.com/en-us/azure/active-directory/connect/active-directory-aadconnect-user-signin>.

Implement and manage Azure AD self-service password reset

Something that is challenging to deploy in an on-premises environment but which is relatively straightforward to deploy in an environment that uses Azure AD as a source of identity authority is self-service password reset. A self-service password reset allows users to reset their own passwords when they forget them, rather than having to contact the service desk and have a member of the IT staff perform the task for them. To enable self-service password reset, perform the following steps:

1. Open the Azure Active Directory portal at <https://aad.portal.azure.com> with an account that has tenant administrator permissions.
2. In the Azure Active Directory admin center, click the **Users** node, which will open the **Users** blade, as shown in Figure 1-24.

Figure 1-24 Azure Active Directory Admin Center

3. On the **Users** blade of the Azure Active Directory admin center, click **Password Reset**.
4. On the **Password Reset – Properties** page, click **All**, as shown in Figure 1-25, to enable the self-service password reset for all Microsoft 365 users.

Figure 1-25 Enable Self-Service Password Reset

Once enabled, users will be prompted for additional information the next time that they sign in. This information will be used to verify their identities if they use the self-service password reset tool. Users can reset their passwords by navigating to the website <https://passwordreset.microsoftonline.com> shown in Figure 1-26 and completing the form.

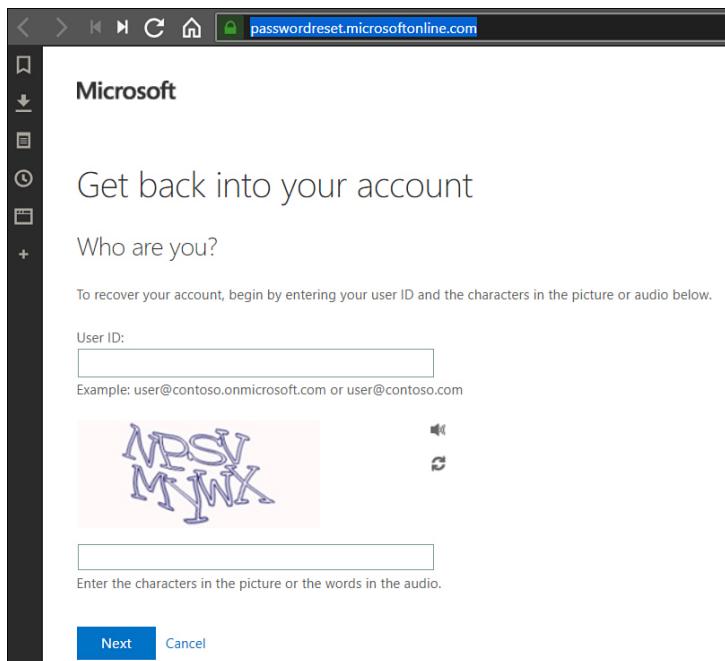


Figure 1-26 Reset Password

More Info Self-Service Password Reset

You can learn more about configuring self-service password at
<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-sspr-howitworks>.

Configure authentication methods including password hash and Pass Through Authentication (PTA), OATH, and passwordless authentication

Another important aspect around designing authentication is deciding which authentication methods will be supported for accounts in your organization's Azure AD instance. For example, you must decide whether you want to support self-service password reset or Azure multifactor authentication, as shown in [Figure 1-27](#).

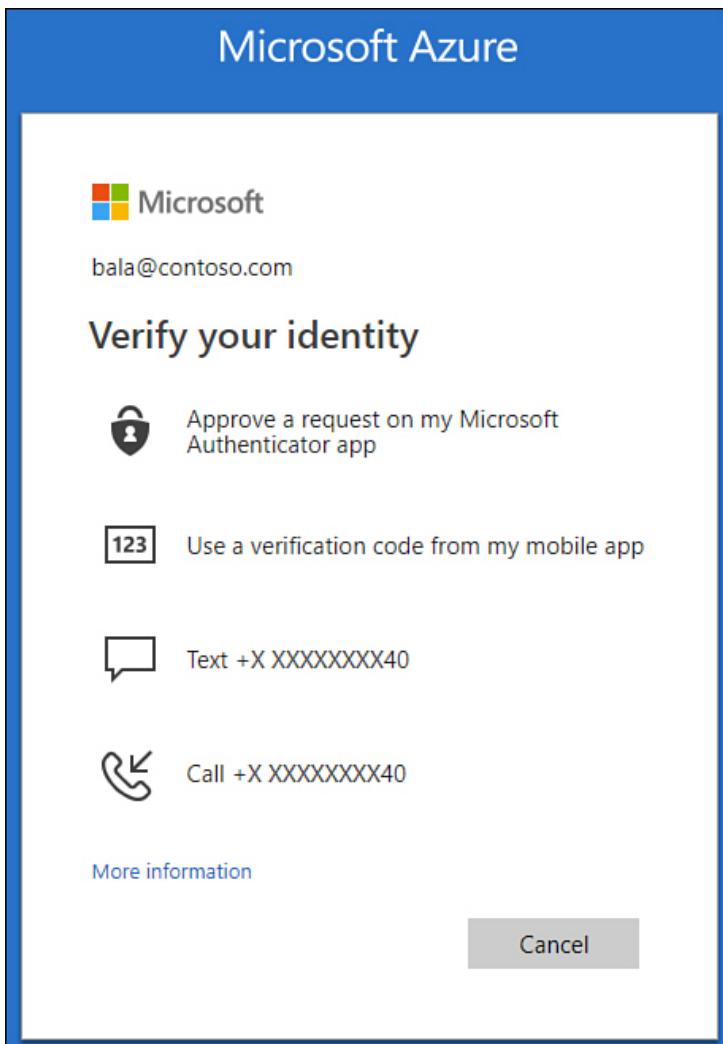


Figure 1-27 Multiple methods of verifying identity during authentication

You can use the authentication methods listed in Table 1-1 with accounts hosted in Azure Active Directory.

Table 1-1 Authentication methods and usage

Authentication method Where it can be used	
Password	Multifactor authentication and self-service password reset
Security questions	Self-service password reset only
Email address	Self-service password reset only

Microsoft Authenticator app	Multifactor authentication and self-service password reset
OATH hardware tokens	Multifactor authentication and self-service password reset
SMS	Multifactor authentication and self-service password reset
Voice call	Multifactor authentication and self-service password reset
App passwords	Multifactor authentication in some cases

These authentication methods have the following properties:

- **Password** The password assigned to an Azure AD account is an authentication method. While you can perform password-less authentication, you cannot disable the password as an authentication method.
- **Security questions** These are only available to Azure AD Self-Service Password Reset and can only be used with accounts that have not been assigned administrative roles. Questions are stored on the user object within Azure AD and cannot be read or modified by an administrator. They should be used in conjunction with another method. Azure AD includes the following predefined questions, and it is possible to create custom questions:
 - In what city did you meet your first spouse/partner?
 - In what city did your parents meet?
 - In what city does your nearest sibling live?
 - In what city was your father born?
 - In what city was your first job?
 - In what city was your mother born?
 - What city were you in on New Year's 2000?
 - What is the last name of your favorite teacher in high school?
 - What is the name of a college you applied to but didn't attend?
 - What is the name of the place in which you held your first wedding reception?
 - What is your father's middle name?
 - What is your favorite food?

- What is your maternal grandmother's first and last name?
 - What is your mother's middle name?
 - What is your oldest sibling's birthday month and year? (for example, November 1985)
 - What is your oldest sibling's middle name?
 - What is your paternal grandfather's first and last name?
 - What is your youngest sibling's middle name?
 - What school did you attend for sixth grade?
 - What was the first and last name of your childhood best friend?
 - What was the first and last name of your first significant other?
 - What was the last name of your favorite grade school teacher?
 - What was the make and model of your first car or motorcycle?
 - What was the name of the first school you attended?
 - What was the name of the hospital in which you were born?
 - What was the name of the street of your first childhood home?
 - What was the name of your childhood hero?
 - What was the name of your favorite stuffed animal?
 - What was the name of your first pet?
 - What was your childhood nickname?
 - What was your favorite sport in high school?
 - What was your first job?
 - What were the last four digits of your childhood telephone number?
 - When you were young, what did you want to be when you grew up?
 - Who is the most famous person you have ever met?
- **Email address** This is only used for Azure AD self-service password resets and should be separate from the user's Microsoft 365 Exchange Online email address.
 - **Microsoft Authenticator app** Is available for Android and iOS. Either involves the user being notified through the mobile app and being asked to select the same number on the mobile app as is displayed on the log in prompt, or it involves the user entering a set of periodically changing numbers displayed on the mobile app.
 - **OATH hardware tokens** Azure AD supports the use of OATH-TOTP SHA-1 tokens of both the 30- and 60-second variety. Secret

keys can have a maximum of 128 characters. Once a token is acquired, it must be uploaded in comma-separated format including UPN, serial number, secret key, time interval, manufacturer, and model. Note that OATH is different from OAuth. OATH is a reference architecture for authentication; OAuth is a standard related to authorization.

- **Mobile phone** Can be used either to send a code through text message that must be entered into a dialog box to complete authentication or where a phone call is made to the user who then needs to provide a personal authentication PIN. Phone numbers must include the country code.
- **App passwords** A number of nonbrowser apps do not support multifactor authentication. An app password allows these users to continue to authenticate using these apps when multifactor authentication is not supported. An app password can be generated for each app, allowing each app password to be individually revoked.

More Info Authentication Methods

You can learn more about authentication methods at
<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-authentication-methods>.

Certificate-based authentication

Certificate-based authentication allows you to eliminate the need for a username and password combination.

Certificate-based authentication is supported on Windows, Android, and iOS devices, and has the following requirements:

- Is only supported for Federated environments for browser applications or where native clients use modern authentication through the Active Directory Authentication Library (ADAL). Exchange Active Sync (EAS) for Exchange Online (EXO) is exempt from the federation requirement and can be used with both federated and managed accounts.
- The organization's root certificate authority (CA) and any intermediate CAs must be integrated with Azure AD.
- Each organizational CA must publish a Certificate Revocation List (CRL) in a location that is accessible to the Internet.
- The Windows, Android, or iOS device must have access to an organizational CA that is configured to issue client certificates.
- The Windows, Android, or iOS device must have a valid certificate installed.
- Exchange ActiveSync clients require that the client certificate have the user's routable email address included in the Subject Alternative Name field.

To add an organizational CA that is trusted by Azure Active Directory, you need to ensure that the CA is configured with a CRL publication location that is accessible on the Internet and to then export the CA certificate. Once you have the CA certificate exported, which will include the Internet-accessible location where the CRL is published, use the New-

AzureADTrustedCertificateAuthority PowerShell cmdlet to add the organizational CA's certificate to Azure Active Directory. You can view a list of trusted CAs for your organization's Azure AD instance using the Get-

AzureADTrustedCertificateAuthority cmdlet.

More Info Certificate Based Azure AD Authentication

You can learn more about certificate-based Azure AD authentication at
<https://docs.microsoft.com/en-us/azure/active-directory/authentication/active-directory-certificate-based-authentication-get-started>.

Passwordless authentication

Passwordless authentication allows you to replace authentication using a password with authentication requiring "something you have" and "something you know." An example of this might be a biometric such as your face or fingerprint combined with a code generated by an authenticator device.

Microsoft currently offers three passwordless authentication options. These are

- **Windows Hello for Business** This method uses biometric authentication technologies included with Windows computers, such as Windows Hello-compatible cameras for facial recognition or Windows Hello-compatible fingerprint readers. Most appropriate for users that are the only people that interact with a specific Windows computer on a regular basis.
- **Security key sign in** Allows access via FIDO2 Security keys. This method is appropriate for users who sign in to shared machines such as those in a call center. Because it requires the physical FIDO2 security key, this is also an excellent method of protecting privileged identities because this key can in turn be secured in a safe that another person has the access code for.

- **Phone sign in through Microsoft Authenticator app** The Microsoft Authenticator app runs on iOS and Android phones and supports identity verification via biometrics or PIN-based authentication. When using this method, a user will be prompted on the screen to select a specific number displayed among a list of options on the Microsoft Authenticator app as well as to perform identity verification via biometrics or a PIN.

Deploying passwordless authentication requires the following administrative roles:

- **Global administrator** Role that allows the implementation of the combined registration experience in the directory.
- **Authentication administrator** Role that can implement and manage authentication methods for individual user accounts.
- **User** Although not an administrative role, this account is necessary to be able to configure an authenticator app on a device or enroll security device for their specific accounts once passwordless authentication is enabled for their accounts.

To enable passwordless phone sign-in authentication, perform the following steps:

1. In the Azure Active Directory admin portal, click **Security**.
2. On the **Security** page shown in Figure 1-28, click **Authentication Methods**.

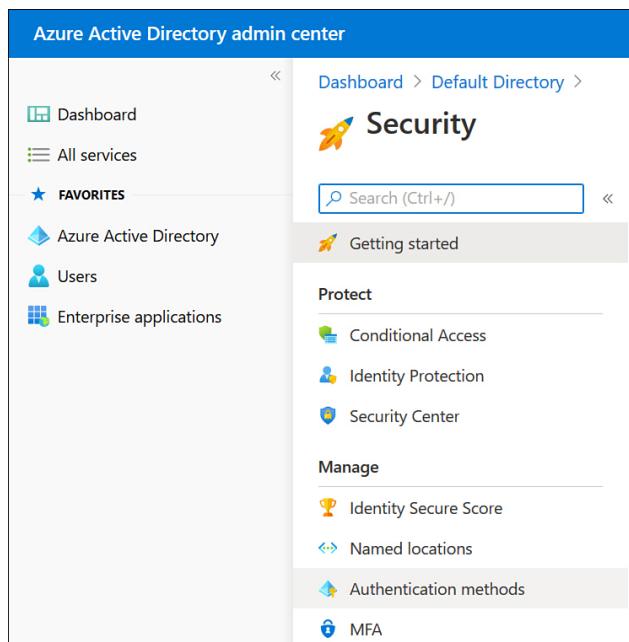


Figure 1-28 Authentication methods section of the Security page

3. On the **Authentication Methods** page shown in Figure 1-29, select the authentication method that you want to enable, toggle

the slider to **On**, and then choose whether you want to enable the authentication method for some or all Azure AD users by choosing **All Users** or **Select Users**.

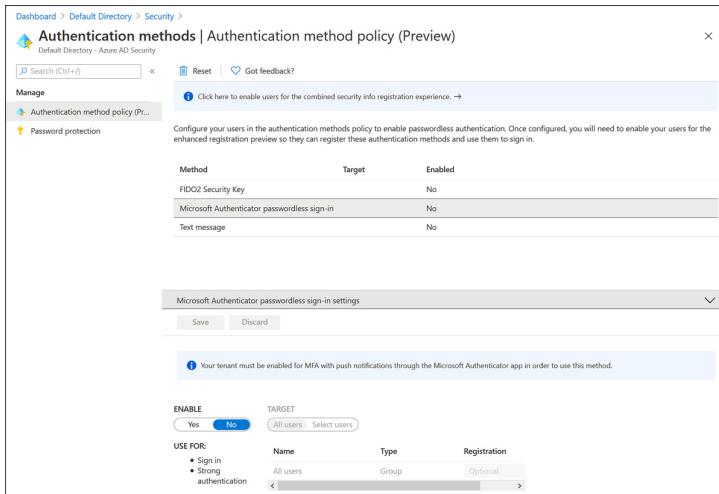


Figure 1-29 Enable passwordless authentication method

More Info Passwordless Azure AD Authentication

You can learn more about passwordless authentication at
<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-authentication-passwordless>.

Transfer Azure subscriptions between Azure AD tenants

Each Azure subscription is associated with and can trust only a specific Azure AD tenancy. Multiple subscriptions can be associated with a specific tenancy. This allows different parts of an organization to use a single set of user accounts across multiple subscriptions. For example, your organization might have a production subscription for production resources, a development subscription for development environment resources, and individual subscriptions for individual developers. A single subscription cannot be associated with multiple Azure AD tenancies. However, it is possible to transfer the Azure AD tenancy with which a subscription is associated.

Reasons that you might need to transfer Azure subscriptions include

- Your organization acquires another organization. You want to move their existing Azure subscriptions under a central tenancy. For example, Contoso acquires Fabrikam, which has 3 existing Azure subscriptions, while Contoso has 10. Contoso wants to move the 3 existing Fabrikam Azure subscriptions in the Contoso Azure AD tenancy.
- Your organization is going to spin off a subsidiary, and you want to move one or more subscriptions to a new directory.
- A subscription has expired, and you have lost access to the resources associated with that subscription. You can associate another subscription with that original subscription's Azure AD tenancy and transfer the existing resources to the new subscription.

When you change a subscription's Azure AD association, any users that have been assigned roles through role-based access control, which you will learn more about later in this chapter, will also lose access. In addition, transferring a subscription to a different Azure AD tenancy removes Policy assignments.

Prior to transferring a subscription to a new Azure AD tenancy, you must have an account that has the Owner role assignment for the subscription. This account must exist in the current directory as well as the new directory. You can add an account from one Azure AD tenancy to another one with B2B collaboration users.

More Info ADD B2B Collaboration Users

You can learn more about B2B collaboration users at
<https://docs.microsoft.com/en-us/azure/active-directory/b2b/add-users-administrator>.

To transfer an existing subscription from one Azure AD tenant to another, perform the following steps:

1. On the **Subscriptions** page of the Azure portal, select **Change Directory**. This will only be possible if the account used to perform this operation has the requisite permissions to perform this action.
2. On the **Change The Directory** dialog box shown in Figure 1-30, select the Azure AD tenancy you want to associate the subscription with and select **Change**, which will change the tenancy.

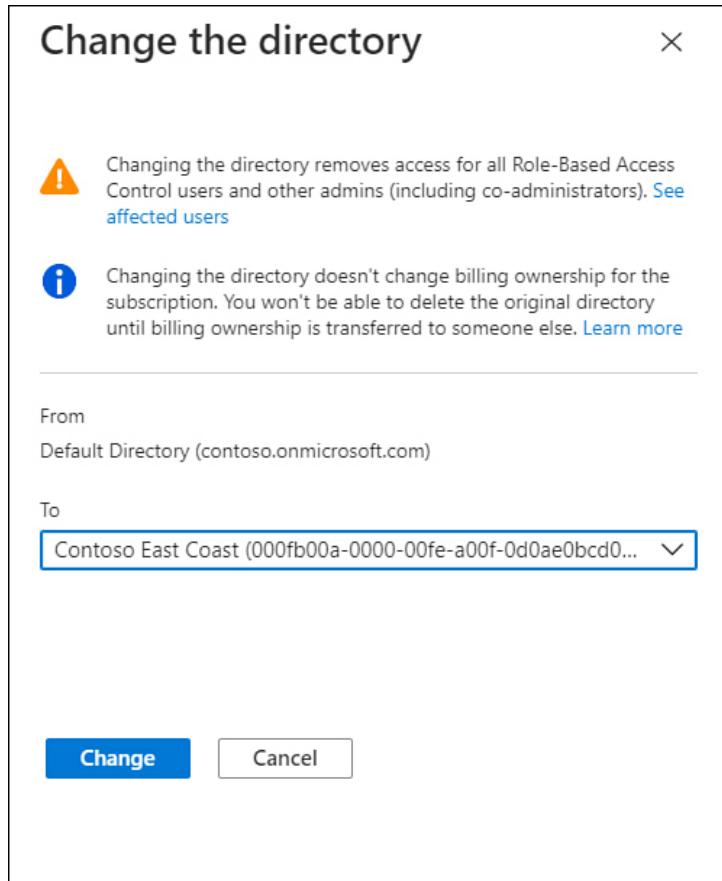


Figure 1-30 Change The Directory dialog box

More Info Associate a Subscription with a Different Tenant
You can learn more about associating a subscription with a different tenant at <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-how-subscriptions-associated-directory>.



Exam Tip

Remember that you can assign rights to an application by associating the application's service principal with specific Azure AD roles.

SKILL 1.2: CONFIGURE SECURE ACCESS BY USING AZURE AD

Azure AD Privileged Identity Management (PIM) allows you to assign Azure AD and Azure Role Based Access Control (RBAC) roles to users on a temporary rather than a permanent basis. Roles can be granted upon request or subject to conditions such as the requestor performing multifactor authentication (MFA) or having their request reviewed and approved by another user. Role activation requests are logged, which provides organizations with an ability to audit the use of administrative privileges in Azure subscription. This objective deals with monitoring privileged access, configuring access reviews, and the process of activating PIM.

Monitor privileged access for Azure AD Privileged Identity Management (PIM)

Privileged Identity Management (PIM) allows you to implement time-based and approval-based activation of administrative roles. For example, you could configure PIM so that a help desk support staff member only has the right to change a user's password for a maximum of 60 minutes once the request for that right has been approved by a specific authorized user. PIM differs from earlier administrative models where the help desk might always be able to change Azure AD user passwords. PIM allows you to do the following:

- Configure just-in-time privileged access to Azure AD and Azure resources. Just-in-time access is access limited to an amount of time, rather than providing permanent access to those resources.
- Assign time-bound access to resources using start and end dates.
- Require approval from another user when activating privileged roles.
- Require multifactor authentication to occur before role activation.
- Require users to provide recorded written justification of why they need to perform activation. This allows auditors at a later stage to correlate the administrative activity that occurs with the stated reason for providing privileged access.
- Provide notifications, such as email alerts sent to a distribution list, when privileged roles are activated.
- Perform access reviews to determine how often privileges are used and whether specific users still require roles.

- Export an audit history that can be examined by internal or external auditors.

To view all activity associated with Azure AD roles, you need to view the resource audit history. To view resource audit history, perform the following steps:

1. In the Azure AD admin center blade of the Azure portal, select **Identity Governance** in the **Manage** area, as shown in Figure 1-31.

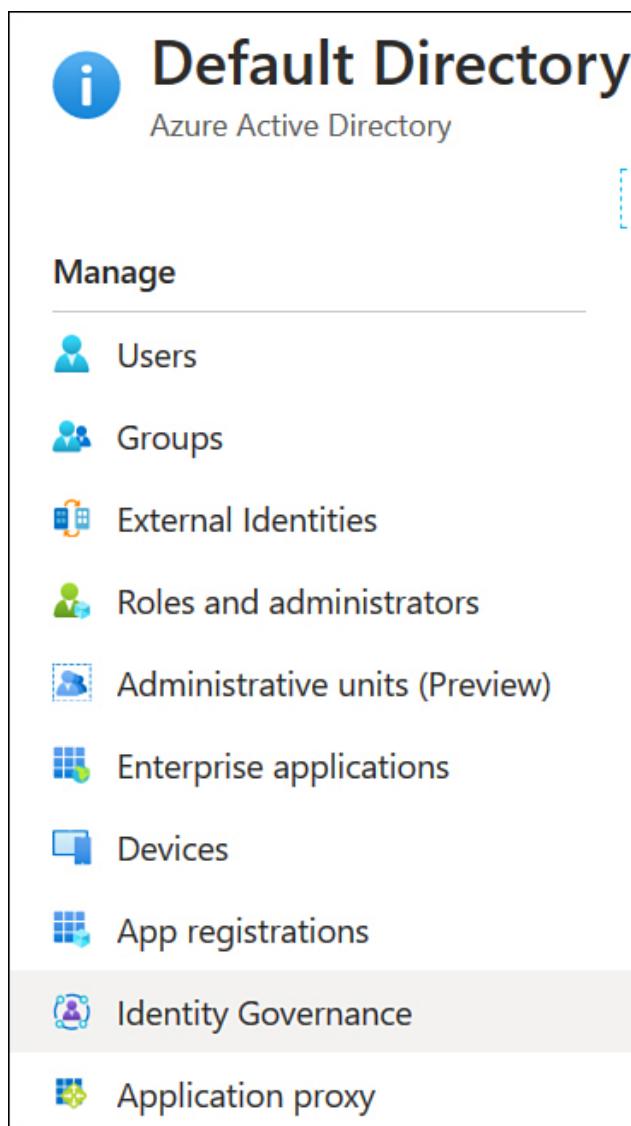


Figure 1-31 Identity Governance

2. On the **Identity Governance** blade, select **Azure AD Roles** under **Privileged Identity Management**.
3. Click **Resource Audit** and then use the filters to view the appropriate information, as shown in Figure 1-32.

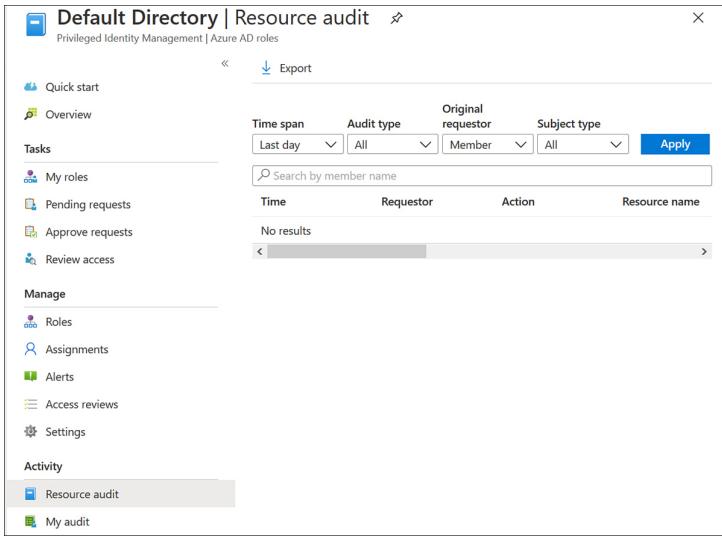


Figure 1-32 Resource Audit

Configure access reviews

Many security incidents have occurred because an attacker has gained access through a forgotten account with administrative privileges. Access reviews allow you to determine whether existing PIM role assignments are still relevant and which role assignments can be removed because they are no longer being actively used.

There are two types of access review: access reviews of Azure resource PIM roles and access reviews of Azure AD PIM roles. To perform an access review of an Azure resource PIM role, perform the following steps:

1. In the Azure AD admin center blade of the Azure portal, select **Identity Governance** in the **Manage** area and then select **Privileged Identity Management**.
2. On the **Privileged Identity Management** blade, click **Azure Resources**, as shown in Figure 1-33.

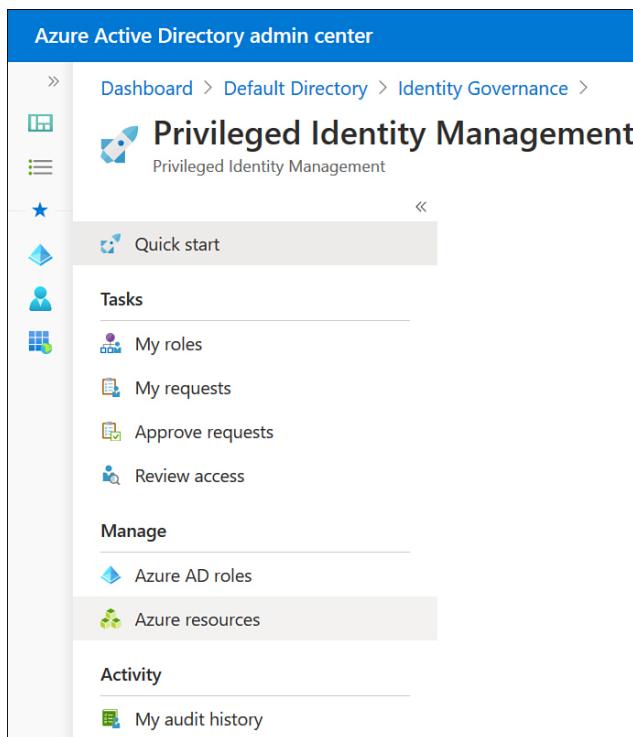


Figure 1-33 Azure resources

3. Existing access reviews will be displayed on the report shown in Figure 1-34.

This screenshot shows the 'Azure Resource access review report' interface. It features a search bar with 'Search by resource name' and a 'Resource filter : Subscription'. Below the search bar are several buttons: 'Refresh', 'Discover resources', 'Activate role', 'Resource type', 'Member', and 'Role'. There are also up and down arrows for sorting.

Figure 1-34 Azure Resource access review report

4. Click **New** to create a new access review. Provide the following information:

1. **Access Review Name** A name for the access review.
2. **Start Date** Date when the review is scheduled to start.
3. **Frequency** How often the review should occur. You can choose a frequency of one time, weekly, monthly, quarterly, annually, or semiannually.
4. **Duration** Specify the number of days the access review will occur over. A longer duration will give you a better idea of how often privileged roles are used.
5. **End** Specify how to end recurring access reviews. You can specify an end date or configure the review to end after a specific number of cycles.
6. **Users** Specify the roles that you are reviewing the membership of.
7. **Reviewers** Specify which people will review all the users.

- 8. Upon Completion** As shown in Figure 1-35, configure how you want the results of the access review implemented. If you want to automatically remove access for users, set **Auto Apply Results To Resources** to **Enable**. If you want to manually apply results once the review is complete, set this to **Disable**.



Figure 1-35 Upon Completion Settings

- **Should Reviewer Not Respond** In this drop-down menu, you have the following options:
 - **No Change** This will ensure that no changes are made to current PIM settings.
 - **Remove Access** This will remove access of users where access is no longer found to be necessary.
 - **Approve Access** Approve user access.
 - **Take Recommendations** Use the system's recommendation when it comes to removing or approving continued access.

The steps for configuring an access review of an Azure AD PIM role are similar to those that you perform when configuring a review to Azure resources, except that you select **Azure AD Roles** instead of **Azure Resources** on the **Manage** menu of the **Privileged Identity Management** blade of the Azure AD admin center.

More Info Review Access to Azure AD Roles

You can learn more about this topic at <https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-how-to-perform-security-review>.

Activate and configure PIM

Azure AD Privileged Identity Management (PIM) allows you to make role assignment temporary and contingent on approval, rather than making the role assignment permanent, as is the case when you manually add a member to the role. PIM requires Azure AD P2, which must be enabled before you can configure it. To

configure an Azure AD administrative role for use with PIM, perform the following steps:

1. In the Azure AD admin center, select **Roles And Administrators**.
2. Select the role to which you want to add a user. This will open the role's properties page.
3. On the **Role Properties page**, click **Manage In PIM**. The role will open, and any members assigned permanently to the role will be listed with the status of **Permanent**, as shown in Figure 1-36.

The screenshot shows the 'Members' section of the 'Security administrator - Members' blade. It includes a search bar and a table with columns for MEMBER, EMAIL, ASSIGNMENT TYPE, and EXPIRATION. Two users are listed: MOD Administrator (email: admin@M365x381963.onmicrosoft.com) and Adele Vance (email: AdeleV@epicentrus.com), both with Permanent assignment type and no expiration date.

MEMBER	EMAIL	ASSIGNMENT TYPE	EXPIRATION
MOD Administrator	admin@M365x381963.onmicrosoft.com	Permanent	-
Adele Vance	AdeleV@epicentrus.com	Permanent	-

Figure 1-36 Members of the Password Administrators role

4. Select the user who you want to convert from **Permanent** to **Eligible**. An eligible user can request access to the role, but that user will not have its associated rights and privileges until that access is granted. On the user's properties page, click **Make Eligible**.

You can edit the conditions under which an eligible user can be granted by performing the following steps:

1. On the **Privileged Identity Management** blade, click **Azure AD Roles**.
2. Under **Manage**, as shown in Figure 1-37, click **Settings**.

The screenshot shows the 'Azure AD roles - Overview' blade. It features a sidebar with tasks like 'My roles', 'My requests', and 'Review access'. The main area has a 'Refresh' button and a chart titled 'My Activation history for the past 7 days'. The chart shows 10 activations with a peak at 1 AM. A legend indicates four categories: PRIVILEGED..., SECURITY AD..., PASSWORD..., and GLOBAL AD... The URL in the address bar is 'Dashboard > Privileged Identity Management > Azure AD roles - Overview'.

Figure 1-37 Manage PIM

3. Click **Roles** and then select the role that you want to configure.

Figure 1-38 shows the PIM settings for the Security Administrator role, where role activation can occur for an hour at most but where MFA and an approval are not required.

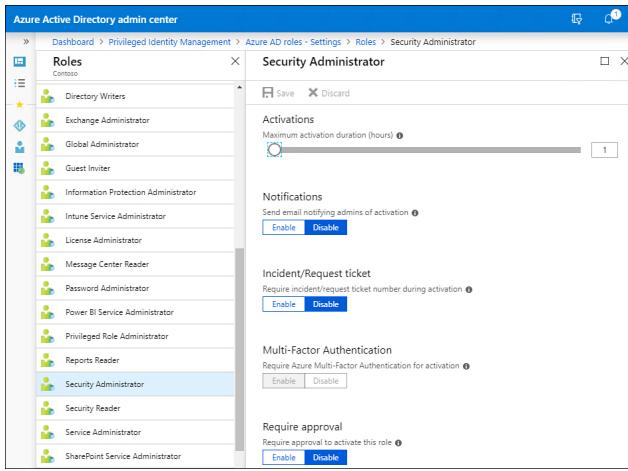


Figure 1-38 Manage PIM settings for a role

Users can activate roles that they are eligible for from the **Privileged Identity Management** area of the Azure AD Administrative console. Administrators with the appropriate permissions can also use the **Privileged Identity Management** area of the Azure AD Administrative console to approve requests that require approval and review role activations.

More Info Privileged Identity Management

You can learn more about topic at <https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-configure>.

PIM requires that you configure Azure AD users with appropriate licenses. PIM requires one of the following license categories to be assigned to users who will perform PIM-related tasks:

- Azure AD Premium P2
- Enterprise Mobility + Security (EMS) E5
- Microsoft 365 M5

The PIM-related tasks that require a license are as follows:

- Any user who is eligible for an Azure AD role that is managed using PIM
- Any user who can approve or reject PIM activation requests
- Users assigned to Azure resource roles with just-in-time or time-based assignments
- Any user who is able to perform an access review
- Any user who is assigned to an access review

More Info PIM License Requirements

You can learn more about PIM license requirements at
<https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/subscription-requirements>.

You cannot use PIM to manage the following classic subscription administrator roles:

- Account Administrator
- Service Administrator
- Co-Administrator

The first person to activate PIM will be assigned the Security Administrator and Privileged Administrator roles for the tenancy.

More Info Activating Privileged Identity Management

You can learn more about topic at <https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-security-wizard>.

Implement conditional access policies including multifactor authentication

Conditional Access policies allow you to require additional steps to be taken when a certain set of circumstances occur. For example, you could configure a conditional access policy to require MFA to occur if a user attempts to access a specific resource in Azure or if a user is accessing Azure from an unusual location.

Conditional access policies can also be used to completely block access to Azure resources when certain conditions are met, such as when someone attempts to access an application from a region from which IP address ranges have been blocked.

Conditional access policies

Conditional access policies will only be enforced after the first-factor authentication has been completed.

Conditional access policies require an Azure AD P2 or equivalent subscription. Commonly used conditional access policies include

- Require MFA for all users with administrative roles
- Require MFA prior to performing Azure management tasks
- Block sign-ins for legacy authentication protocols
- Require trusted location when registering for Azure MFA
- Block access from specific locations
- Require organization-managed devices for certain applications

Conditional access policies can be applied based on user circumstances that include but not limited to the following:

- **IP address location** An administrator can designate certain IP address ranges as trusted, such as the public IP addresses associated with the organization's Internet gateway devices. Administrators can also specify regional IP address ranges as being blocked from access, such as those belonging to people trying to access resources from Tasmania.
- **Device** Whether the user is attempting to access Azure AD resources from a trusted device or from a new untrusted device.
- **Application** Whether the user is attempting to access a specific Azure AD application.
- **Group membership** Whether the user is a member of a specific group.

In addition to the simple option to block access, conditional access policies can be configured to

- Require multifactor authentication
- Require a device to be marked as compliant
- Require the device to be Hybrid Azure AD-joined
- Require an approved client app
- Require an app protection policy

To create a conditional access policy, perform the following steps:

1. In the Azure Active Directory area of the Azure portal, select **Security** and then select **Conditional Access**, as shown in

Figure 1-39.

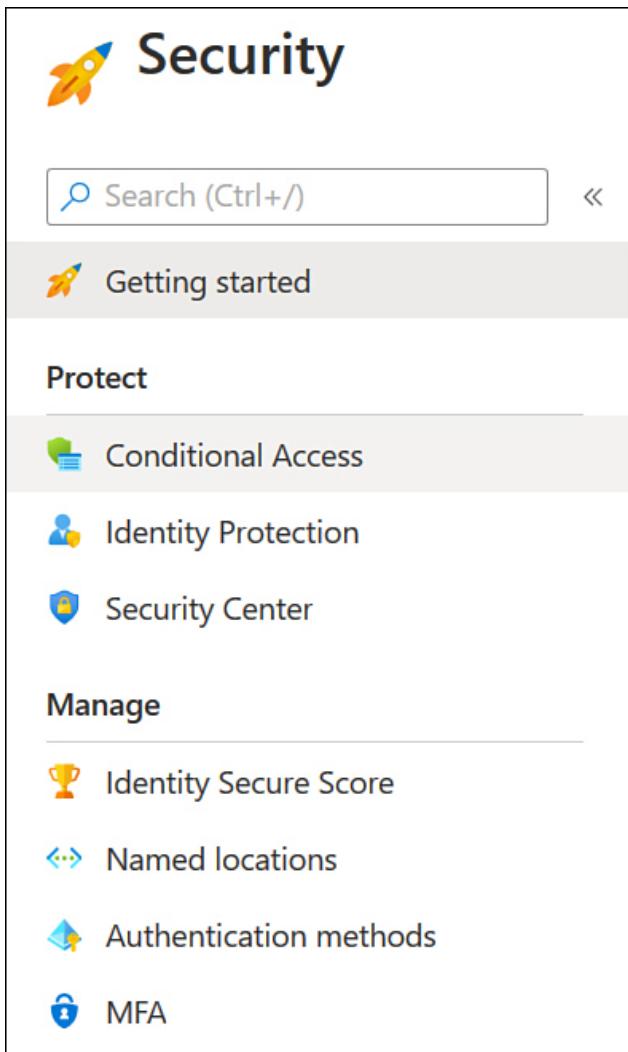


Figure 1-39 Security page with Conditional Access highlighted

2. On the **Conditional Access | Policies** page shown in Figure 1-40, select **New Policy**.

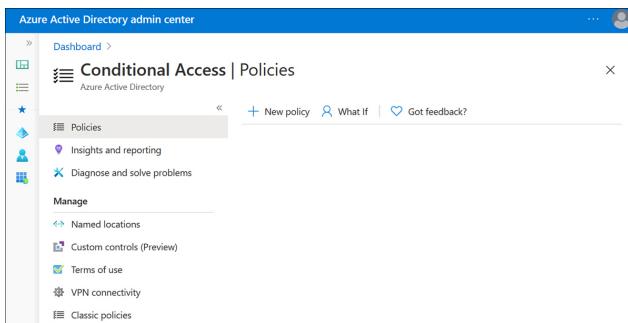


Figure 1-40 Conditional Access policies

3. On the **New Conditional Access Policy** page shown in [Figure 1-41](#), provide the following information:

1. **Name** A name for the conditional access policy.
2. **Users And Groups** Users and groups that the policy applies to.
3. **Cloud Apps Or Actions** Which cloud apps or user actions the policy applies to. Policies can apply to some or all cloud apps. You can also specify specific user actions that will trigger the conditional access policy, such as attempting to access a specific Azure resource, such as a virtual machine.
4. **Conditions** The conditions associated with the policy. These include User risk, sign-in risk, device platforms, locations, client apps and device state.
5. **Access Controls** Select which additional controls are required to grant access. This gives you the option of requiring MFA, a compliant device, an Hybrid Azure AD-joined device, an approved client app, an app protection policy, or that the user perform a password change.
6. **Session** Allows you to specify the behavior of specific cloud applications. Options include **Conditional Access App Control**, **Sign-In Frequency**, and **Persistent Browser Session**.
7. **Enable Policy** Can be set to **Report Only**, which you should use to determine how the policy will function prior to enforcing it, enabling the policy, or disabling the policy.

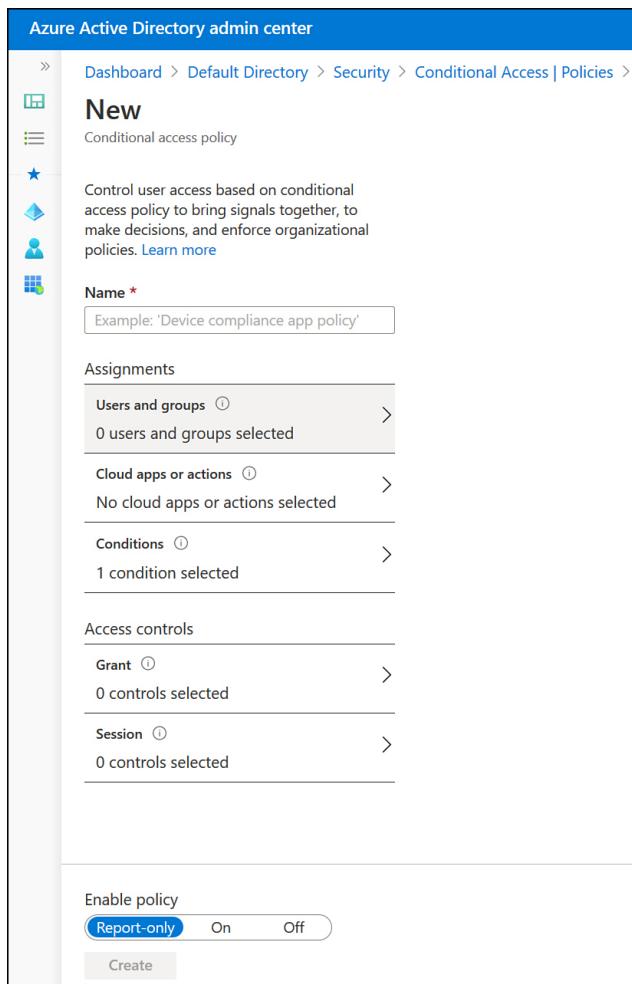


Figure 1-41 New Conditional Access policy

4. Click **Create** to create the policy.

More Info Conditional Access Policies

You can learn more about Conditional access policies at
<https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/overview>.

Implementing MFA

When implementing MFA, you need to make decisions about which MFA capabilities will be available to the users associated with your organization's Azure AD tenancy. MFA requires that more than one authentication method be used when signing in to a resource. Usually, this involves the user providing their

username and password credentials, and then providing one of the following:

- **A code generated by an authenticator app** This can be the Microsoft Authenticator app or a third-party authenticator app, such as the Google authenticator app.
- **A response provided to the Microsoft authenticator app** When this method is used, Azure AD provides an on-screen code to the user authenticating that must also be selected on an application that is registered with Azure AD.
- **A phone call to a number registered with Azure AD** The user needs to provide a preconfigured pin that they will be instructed to enter by the automated service that performs the phone call. Microsoft provides a default greeting during authentication phone calls, so you don't have to record one for your own organization.
- **An SMS message sent to a mobile phone number registered with Azure AD** The user provides the code sent in the message as a second factor during authentication.

When designing your solution, you'll need to have a way of ensuring that users have access to the appropriate MFA technology. This might require you to come up with a method of ensuring that all users in your organization already have the Microsoft Authenticator app installed on their mobile devices before you enable MFA on their accounts.

More Info Plan for Multifactor Authentication

You can learn more about designing a multifactor authentication solution for Office 365 deployments at <https://docs.microsoft.com/en-us/office365/admin/security-and-compliance/multi-factor-authentication-plan>.

MFA is not enabled by default on Azure AD tenancies. Before you can configure accounts to use MFA, you'll need to enable MFA on the tenancy. To enable MFA on an Azure AD tenancy and configure MFA for specific users, perform the following steps:

1. In Azure Active Directory admin center, navigate to **Users** and then click **All Users**.
2. Click **More**, and then click **Multi-Factor Authentication**, as shown in Figure 1-42.

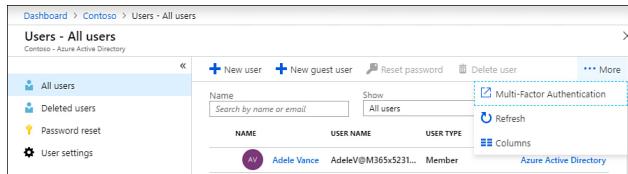


Figure 1-42 Set up Azure MFA

- After selecting this option, MFA will be enabled for the tenancy, and you'll be provided with a list of users that is similar to that shown in Figure 1-43.

DISPLAY NAME	USER NAME	MULTI-FACTOR AUTH STATUS
Adele Vance	AdeleV@M365x523191.OnMicrosoft.com	Disabled
Alex Wilber	AlexW@M365x523191.OnMicrosoft.com	Disabled
Allan Deyoung	AllanD@M365x523191.OnMicrosoft.com	Disabled

Figure 1-43 Set up users for Azure MFA

- Select the users who you want to set up for MFA, as shown in Figure 1-44, and then click **Enable**.

DISPLAY NAME	USER NAME	MULTI-FACTOR AUTH STATUS
<input checked="" type="checkbox"/> Adele Vance	AdeleV@M365x523191.OnMicrosoft.com	Disabled
<input checked="" type="checkbox"/> Alex Wilber	AlexW@M365x523191.OnMicrosoft.com	Disabled
<input checked="" type="checkbox"/> Allan Deyoung	AllanD@M365x523191.OnMicrosoft.com	Disabled
<input type="checkbox"/> Bianca Pisani	BiancaP@M365x523191.onmicrosoft.com	Enabled
<input type="checkbox"/> Brian Johnson (TAILSPIN)	BrianJ@M365x523191.onmicrosoft.com	Disabled

Figure 1-44 Enable Azure MFA

- On the **About Enabling Multi-Factor Auth** dialog box shown in Figure 1-45, click **Enable Multi-Factor Auth**.



Figure 1-45 Enabling multifactor authorization

- The next time that users sign on, they will be prompted to enroll in multifactor authentication and will be presented with a dialog box

similar to that shown in Figure 1-46, asking them to provide additional information.

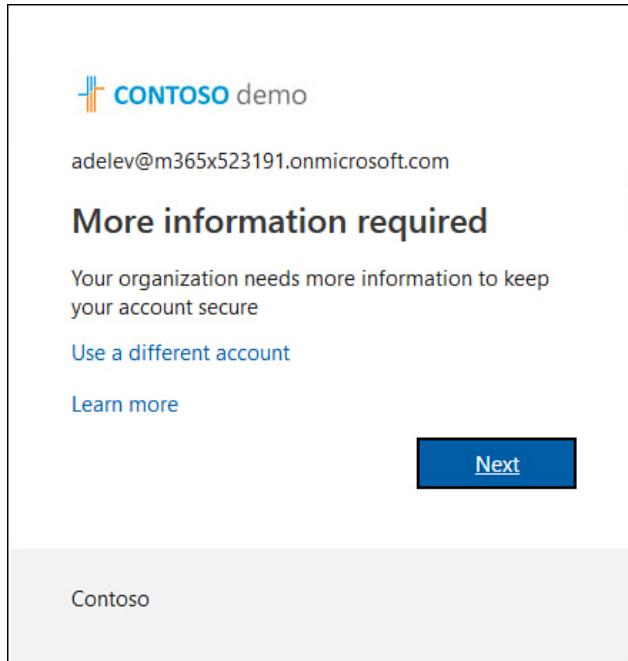


Figure 1-46 More information required

7. And then choose between providing a mobile phone number or an office phone number or configuring a mobile app, as shown in Figure 1-47.

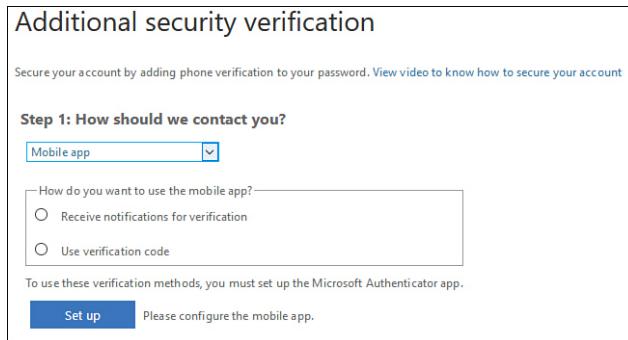


Figure 1-47 Contact preferences

8. When you specify one of these options, you are presented with a QR code. Within the app, you can add a new account by scanning the QR code. Once you have configured the application, you will be required to confirm that configuration has completed successfully by approving a sign in through the app, as shown in Figure 1-48.

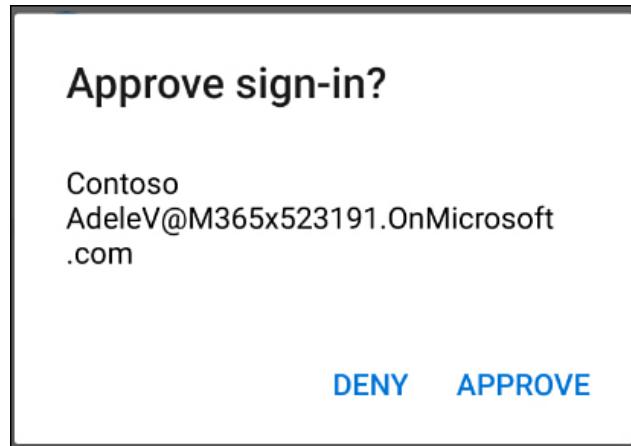


Figure 1-48 Verify the app

9. Once this is done, you'll be prompted to provide additional security information in the form of a phone number, as shown in Figure 1-49.

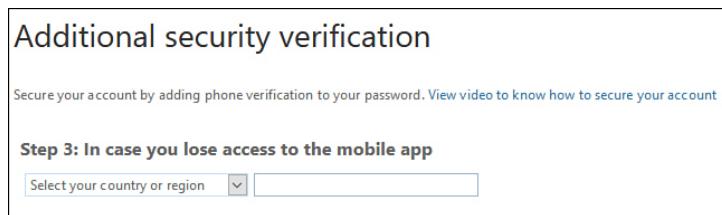


Figure 1-49 Provide additional security information

You can configure the following multifactor authentication service settings, as shown in Figure 1-50.

- **App Passwords** Allow or disallow users from using app passwords for nonbrowser apps that do not support multifactor authentication.
- **Trusted IP Addresses** Configure a list of trusted IP addresses where MFA will be skipped when federation is configured between the on-premises environment and the Microsoft 365 Azure AD tenancy.
- **Verification Options** Specify which verification options are available to users, including phone call, text message, app-based verification, or hardware token.
- **Remember Multi-Factor Authentication** Decide whether to allow users to have MFA authentication remembered for a specific period of time on a device, so that MFA does not need to be performed each time the user signs on. The default is 14 days.

multi-factor authentication

users service settings

app passwords ([learn more](#))

Allow users to create app passwords to sign in to non-browser apps
 Do not allow users to create app passwords to sign in to non-browser apps

trusted ips ([learn more](#))

Skip multi-factor authentication for requests from federated users on my intranet
Skip multi-factor authentication for requests from following range of IP address subnets

192.168.1.0/27
192.168.1.0/27
192.168.1.0/27

verification options ([learn more](#))

Methods available to users:

Call to phone
 Text message to phone
 Notification through mobile app
 Verification code from mobile app or hardware token

remember multi-factor authentication ([learn more](#))

Allow users to remember multi-factor authentication on devices they trust
Days before a device must re-authenticate (1-60):

Figure 1-50 MFA service settings

More Info Set Up Multifactor Authentication

You can learn more about multifactor authentication at
<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-mfa-howitworks>.

Administer MFA users

Once MFA is configured for users, there might be certain times when you want to force users to provide updated contact methods, you might want to revoke all app passwords, or you might want to restore MFA on all

remembered devices. You can do this by performing the following steps:

1. With an account that has been assigned the Global Admin role, open the Azure AD admin center and select the **All Users** node, as shown in Figure 1-51. Select the user to manage MFA.

The screenshot shows the 'Users - All users' page in the Azure Active Directory admin center. The left sidebar includes options like 'All users', 'Deleted users', 'Password reset', 'User settings', 'Activity', 'Sign-ins', 'Audit logs', 'Troubleshooting + Support', 'Troubleshoot', and 'New support request'. The main area displays a table of users with columns for NAME, USER NAME, USER TYPE, and SOURCE. The users listed are Adele Vance, Alex Wilber, Allan Deyoun, Bianca Pisani, Brian Johnson, Cameron Whi, and Christie Cline, all categorized as 'Member' under 'Azure Active Directory'.

Figure 1-51 Select the user to manage MFA

2. On the user's properties page, select **Authentication Methods**.
3. On the **Authentication Methods** page shown in Figure 1-52, select which action to perform.

The screenshot shows the 'Adele Vance - Authentication methods' page. The left sidebar lists 'Manage' options: Profile, Directory role, Groups, Applications, Licenses, Devices, Azure resources, and Authentication methods (which is selected and highlighted in blue). Below that is an 'Activity' section with Sign-ins and Audit logs. The main content area contains two sections: 'Require MFA re-registration' and 'Revoke MFA sessions'. The 'Require MFA re-registration' section explains the process and includes a 'Re-' button. The 'Revoke MFA sessions' section explains the process and includes a 'Revoke' button.

Figure 1-52 Authentication methods

If you want to perform a bulk reset for multiple users, use the following steps:

1. From the **All User's** page shown in Figure 1-53, click **Multi-factor authentication**.

Users - All users					
Contoso - Azure Active Directory					
		+ New user	+ New guest user	Reset password	Delete user
		Search by name or email	Show All users		
Name	User Name	User Type	Source		
AV Adele Vance	AdeleV@M365x523191.OnMicrosoft.com	Member	Azure Active Directory		
AW Alex Wilber	AlexW@M365x523191.OnMicrosoft.com	Member	Azure Active Directory		
AD Allan Deyoung	AllanD@M365x523191.OnMicrosoft.com	Member	Azure Active Directory		
BP Bianca Pisani	BiancaP@M365x523191.onmicrosoft.com	Member	Azure Active Directory		

Figure 1-53 List of users

2. On the **Multi-factor Authentication** users page shown in Figure 1-54, select the users for whom you want to reset MFA settings and click **Manage User Settings**.

multi-factor authentication		
users service settings		
Before you begin, take a look at the multi-factor auth deployment guide.		
View: Sign-in allowed users Multi-Factor Auth status: Any		
DISPLAY NAME	USER NAME	MULTI-FACTOR AUTH STATUS
<input checked="" type="checkbox"/> Adele Vance	AdeleV@M365x523191.OnMicrosoft.com	Enforced
<input checked="" type="checkbox"/> Alex Wilber	AlexW@M365x523191.OnMicrosoft.com	Enabled
<input checked="" type="checkbox"/> Allan Deyoung	AllanD@M365x523191.OnMicrosoft.com	Enabled
<input type="checkbox"/> Bianca Pisani	BiancaP@M365x523191.onmicrosoft.com	Enabled
<input type="checkbox"/> Brian Johnson (TAILSPIN)	BrianJ@M365x523191.onmicrosoft.com	Disabled
<input type="checkbox"/> Cameron White	CameronW@M365x523191.onmicrosoft.com	Disabled
		3 selected
		quick steps
		To update Multi-factor Auth status, select users who have the same status.
		Manage user settings

Figure 1-54 Select users for MFA reset

3. On the **Manage User Settings** page shown in Figure 1-55, select which tasks you want to perform, such as requiring users to provide contact methods again, deleting all existing app passwords, and restoring MFA on remembered devices. After making the selection, click **Save**.

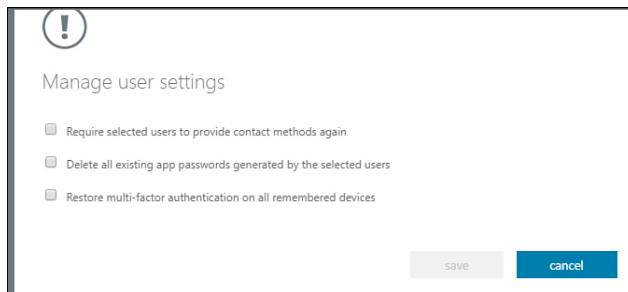


Figure 1-55 Managing user settings

More Info Setting Up Multifactor Authentication

You can learn more about setting up multifactor authentication at <https://docs.microsoft.com/en-us/office365/admin/security-and-compliance/setup-multi-factor-authentication>.

Account lockout

Account Lockout settings for MFA, shown in [Figure 1-56](#), allow you to configure the conditions under which MFA lockout will occur. On this page, you can configure the number of MFA denials that will trigger the account lockout process, how long before the account lockout counter is reset, and the number of minutes until the account will be unblocked. For example, if the account lockout counter is reset after 10 minutes, and the number of MFA denials to trigger account lockout is set to 5, then 5 denials in 10 minutes will trigger a lockout, but 5 denials over a course of 30 minutes would not because the account lockout counter would reset during that period.

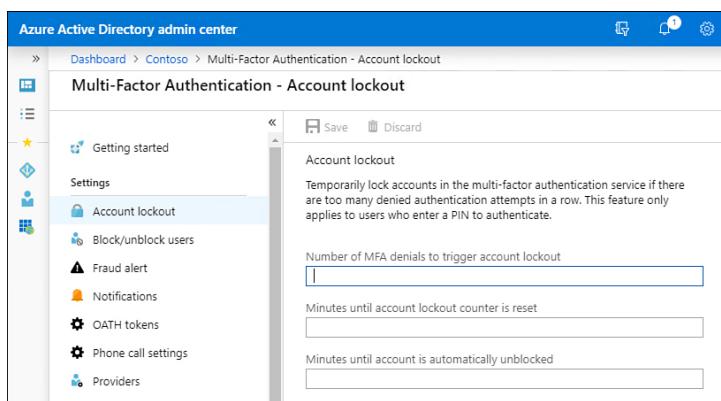


Figure 1-56 Account lockout settings

Block/unblock users

The blocked user setting shown in [Figure 1-57](#) allows you to block specific users of an on-premises MFA server from being able to receive an MFA request. Any requests sent to a user on the blocked users list will automatically be denied. Users on this list remain blocked for 90 days, after which they are removed from the blocked users list. To unblock a blocked user, click **Unblock**.

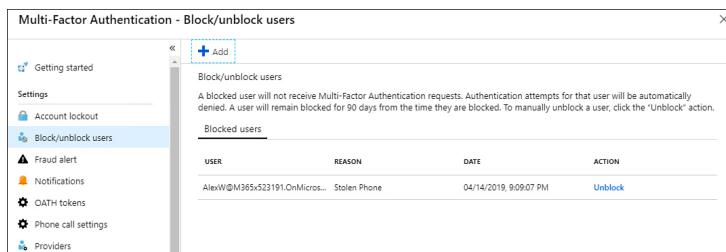


Figure 1-57 Block/Unblock Users page

Fraud alert settings

Fraud alert settings, shown in [Figure 1-58](#), allow you to configure whether users can report fraudulent verification requests. A fraudulent verification request might occur when an attacker has access to a user's password but does not have access to an alternative MFA method. A user becomes aware of this by receiving an MFA prompt, either through his or her app, an SMS, or a phone call when they haven't attempted to authenticate against a Microsoft 365 workload. When a user reports fraud, you can choose an option to have his or her account automatically blocked for 90 days, which indicates that the password is likely to be compromised.

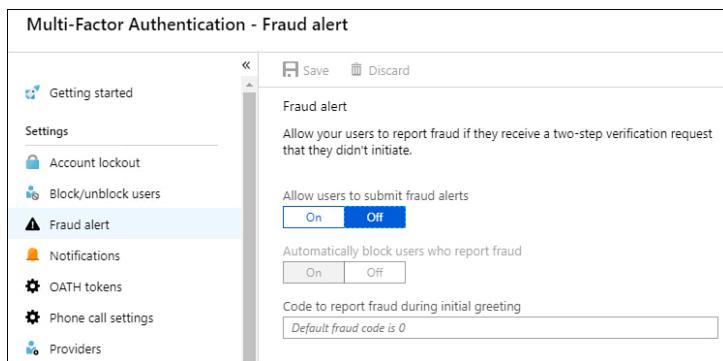


Figure 1-58 Fraud alert page

OATH tokens

The OATH tokens page shown in [Figure 1-59](#) allows you to upload a specially formatted CSV file that contains the details and keys of the OATH tokens that you want to use for multifactor authentication. The specially formatted CSV file should include a header row that is formatted as

shown here with the UPN (user principal name), serial number, secret key, time interval, manufacturer, and model. Each file is associated with a specific user. If a user has multiple OATH tokens, these should be included in the file associated with his or her account.

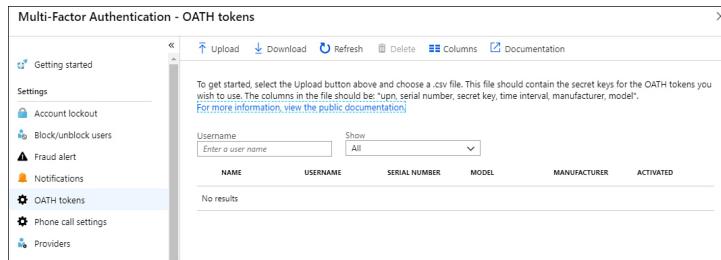


Figure 1-59 OATH tokens page

Phone call settings

Phone call settings allow you to configure the caller ID number that is displayed when the user is contacted for MFA authentication. This number must be a United States number. You can also use the phone call settings page shown in [Figure 1-60](#) to configure custom voice messages. The voice messages must be in .wav or .mp3 format, must be no larger than 5 MB, and should be shorter than 20 seconds.

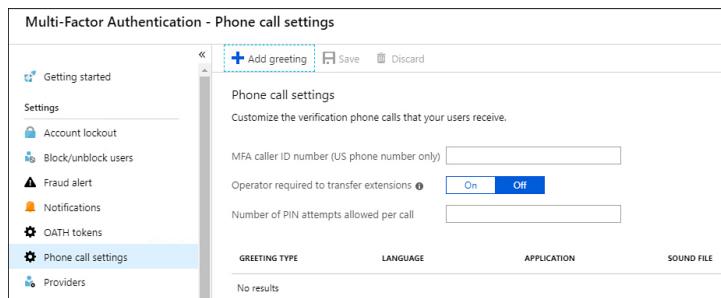


Figure 1-60 Phone Call Settings page

More Info Managing MFA Settings

You can learn more about managing MFA settings at
<https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-mfa-mfasettings>.

Report MFA utilization

Azure MFA provides a number of reports that you can use to understand how MFA is being used in your organization, including

- **Blocked User History** Provides a history of requests to block or unblock users.
- **Usage And Fraud Alerts** Provides information on a history of fraud alerts submitted by users. Also provides information on overall MFA usage
- **Usage For On-Premises Components** Provides information of utilization of MFA through the Network Policy Server extension, Active Directory Federation Services, and on-premises MFA server
- **Bypassed User History** Provides information on requests to bypass MFA by a specific user
- **Server Status** Provides status data of MFA servers associated with your organization's Azure AD tenancy

More Info Azure Multifactor Authentication Reports

You can learn more about Azure multifactor authentication reports at <https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-mfa-reporting>.



Exam Tip

Remember the steps that you can take to automatically lock out users who incorrectly answer MFA prompts.

Configure Azure AD Identity Protection

Azure AD Identity Protection allows you to automate the detection and remediation of identity-based risks, including the following:

- **Atypical travel** When a user's account sign-in indicates he or she has performed unusual shifts in location. This could include a user signing in from Sydney and then Los Angeles in a two-hour period, when the flight between the two cities takes about seven times that amount of time.
- **Anonymous IP address** When a user signs in from an anonymous IP address. While a user might be using an anonymizing VPN to access organizational resources, attackers

also use tools such as TOR nodes when launching compromise attempts.

- **Unfamiliar sign-in properties** When a user's sign-in properties differ substantially from those that have been observed in the past.
- **Malware-linked IP address** When the IP address the user is signing in from is known to be part of a malware botnet or has exhibited other malicious network activity in the past.
- **Leaked credentials** That the user's credentials have been discovered in a data breach, such as those recorded on haveibeenpwned.com.
- **Azure AD threat intelligence** That the sign-in behavior correlates with a known attack pattern identified by Microsoft's internal or external threat intelligence sources.

Enabling Azure AD Identity protection requires an Azure AD P2 license.

Azure AD Identity Protection allows you to configure two types of risk policy: a sign-in risk policy and a user-risk policy:

- **Sign-in risk** These policies analyze signals from each sign-in and determine how likely it is that the sign-in was not performed by the person associated with the user account. If a sign-in is determined to be risky, administrators can specify whether to block access or allow access but require multifactor authentication.
- **User-risk** These policies are based on identifying deviations from the user's normal behavior. For example, the user signs in from an unusual location at a time that substantially differs from when they usually sign in. User risk policies allow administrators to block access, allow access, or allow access but require a password change when the policy is triggered.

To enable user risk and sign-in risk policies, perform the following steps:

1. In the Azure Active Directory admin center, select **Security** in the **Manage** area and then select **Identity Protection**.
2. In the **Protect** section of the **Identity Protection** blade, which is shown in Figure 1-61, select **User Risk Policy**.

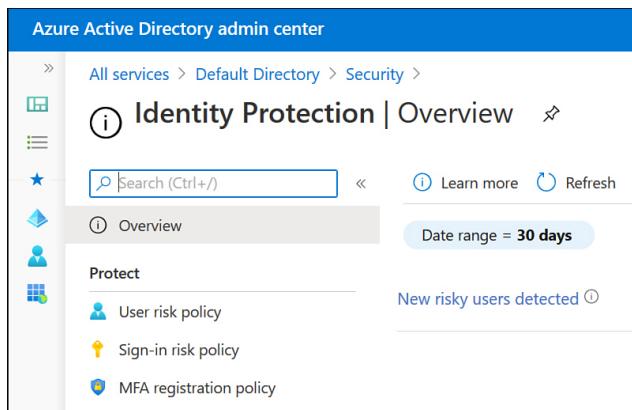


Figure 1-61 Identity Protection blade

3. Click **User Risk Policy**. On the **User Risk Policy** blade, which is shown in Figure 1-62, configure the following settings.

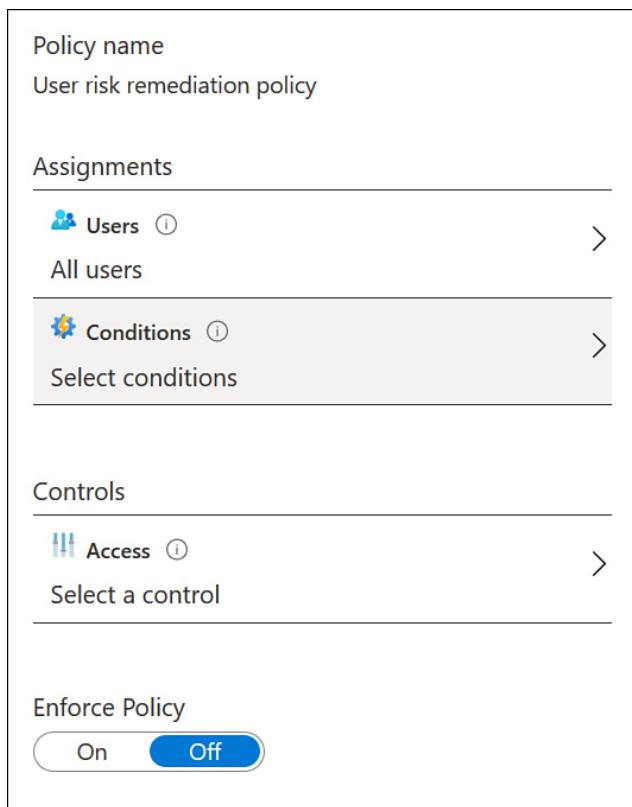


Figure 1-62 User Risk Remediation Policy

4. Click **Sign-In Risk Policy**. On the **Sign-In Risk Remediation Policy** blade, which is shown in Figure 1-63, configure the following settings and click **Save**:

1. **Assignments: Users** Determine which users the user risk remediation policy applies to.
2. **Assignments: Conditions** Allows you to determine at which risk level the policy applies. You can choose

between **Low And Above**, **Medium And Above**, or **High**.

3. **Controls: Access** For a user risk policy, you can choose between **Block**, **Allow**, and **Allow And Require Multi-Factor Authentication**.
4. **Enforce policy** The policy can be switched **On** or **Off**.

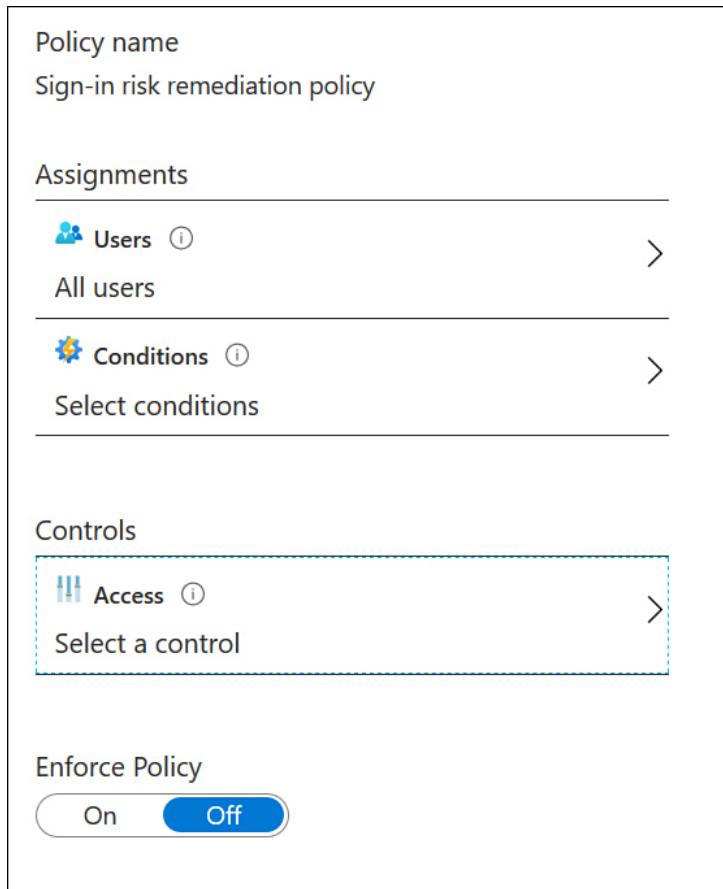


Figure 1-63 Sign-In Risk Remediation Policy

More Info Azure AD Identity Protection

You can learn more about Azure AD identity protection at
<https://docs.microsoft.com/en-us/azure/active-directory/identity-protection/overview-identity-protection>.



Exam Tip

Remember the requirements for enabling MFA on an Azure AD tenancy.

SKILL 1.3: MANAGE APPLICATION ACCESS

This objective deals with the steps that can be taken to configure and manage application access. This includes understanding application registration with an Azure AD tenancy, managing access to applications themselves, configuring application permissions scopes, application permission consent, and API access to Azure subscriptions and resources.

This section covers the following topics:

- Create app registrations
- Configure app registration permission scopes
- Manage app registration permission consent
- Manage API access to Azure subscriptions and resources

Create app registrations

As you learned earlier in the chapter, registering an application with Azure Active Directory allows you to use Azure Active Directory's functionality, such as user identity and permissions, with the application. To register an application with Azure Active Directory using the Azure portal, perform the following steps:

1. In the Azure portal, open the **Azure Active Directory** blade.
2. In the **Manage** section shown in Figure 1-64, click **App Registrations**.

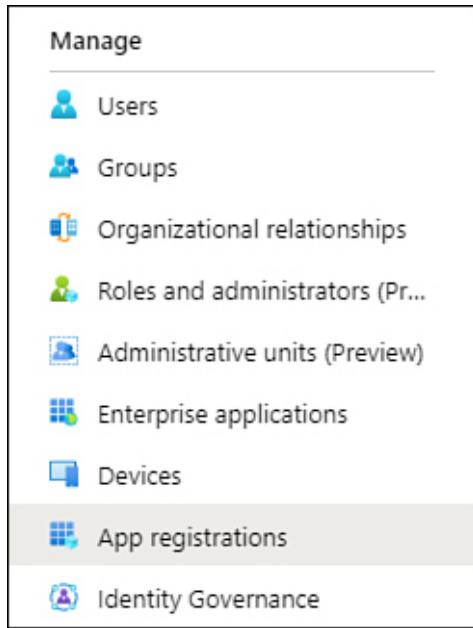


Figure 1-64 App Registrations section of the Azure Active Directory blade

3. On the **App Registrations** blade of the **Azure Active Directory** section of the Azure portal, click **New Registration**. Figure 1-65 shows the New Registration item.

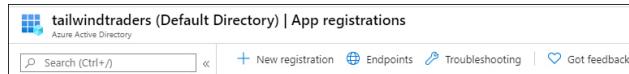


Figure 1-65 App Registrations blade with the New Registration option

4. On the **Register An Application** page, shown in Figure 1-66, choose which users can use this application or access this API. You can choose among the following options:

1. **Accounts In This Organizational Directory Only**

Appropriate for single-tenant scenarios where the only people who will use the application have accounts that reside within the Azure AD instance. You can switch to the multitenant option and back to the single-tenant option after registration is complete using the **Authentication** page in the Azure portal.

2. **Accounts In Any Organizational Directory**

Choose this option when you want to make the application available to users in your own and other Azure AD tenancies. This is also known as the multitenant option. You can switch between this option and the single-tenant option using the **Authentication** page in the Azure portal.

3. **Accounts In Any Organizational Directory And Personal Microsoft Accounts**

This option allows not only users who have accounts in Azure AD tenancies but

also personal Microsoft accounts, such as Hotmail.com and outlook.com accounts. You can't presently switch from this mode to multitenant or single-tenant in the Azure portal, but you can make this change if you use the application manifest editor.

The screenshot shows the 'Register an application' form. The 'Name' field contains 'AZ500Example'. Under 'Supported account types', the radio button for 'Accounts in this organizational directory only (tailwindtraders (Default Directory) only - Single tenant)' is selected, while the other two options ('Accounts in any organizational directory (Any Azure AD directory - Multitenant)' and 'Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)') are unselected.

Figure 1-66 Supported account types for app registration

5. The **Redirect URI (Optional)** section, shown in Figure 1-67, allows you to specify the type of app that is being registered, with the options being **Web** or **Public Client (Mobile & Desktop)**. If you are registering a web app, you need to specify the base URL of the app (for example, <https://newapp.tailwindtraders.net:31544>). If you choose the Public Client option, you instead need to provide the Uniform Resource Identifier (URI) that Azure AD will use to return token responses that is specific to the application that you are registering.

The screenshot shows the 'Redirect URI (optional)' section. A dropdown menu is set to 'Web', and a text input field contains 'e.g. https://myapp.com/auth'.

Figure 1-67 Redirect URI

6. After providing this information, click **Register**.

Once the app registration process is complete, the app will be assigned a unique application or client ID and it will be listed on the **App Registrations** page in the Azure portal, as shown in Figure 1-68.

The screenshot shows the 'App registrations' page. It lists three applications:

Display name	Application (client) ID	Created on	Certificates & secrets
WindowsAdminCenter-https://wt-virtnv-01	c0572472-6494-4d92-a443-e1a1388b9a97	4/25/2019	-
WindowsAdminCenter-https://mel-fs-1.contoso.internal	0efff188-bc39-4c20-9a9b-7d35980e0f0f	8/3/2019	-
AZ500Example	e5ba2bc8-7e53-4eb3-9ebc-63ac0bc24fb0	4/27/2020	-

Figure 1-68 App Registrations

More Info Registering an Application

You can learn more about registering an application at
<https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-register-app>.

Managing access to apps

How you assign access to applications depends on the edition of Azure AD that your organization has licensed. If your organization only has a free edition of Azure AD, you'll only be able to assign access to applications on a per-user basis. If your organization licenses a paid edition of Azure AD, then you'll be able to perform a group-based assignment. When you perform a group-based assignment, whether a user is able to access an application will depend on whether the user is a member of the group at the time he or she attempts to access the application.

Any form of Azure AD group can be used to assign access to applications, including attribute-based dynamic groups, on-premises Active Directory groups, or self-service managed groups. Nested group membership are not presently supported when it comes to assigning access to applications through Azure AD.

More Info Managing Access to Apps

You can learn more about managing access to apps at
<https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/what-is-access-management>.

Assigning users access to an application

To assign access to an application to a user or group, perform the following steps:

1. In the Azure AD admin center, select **Azure Active Directory** and in the **Manage** section, click **Enterprise Applications**, as shown in Figure 1-69.

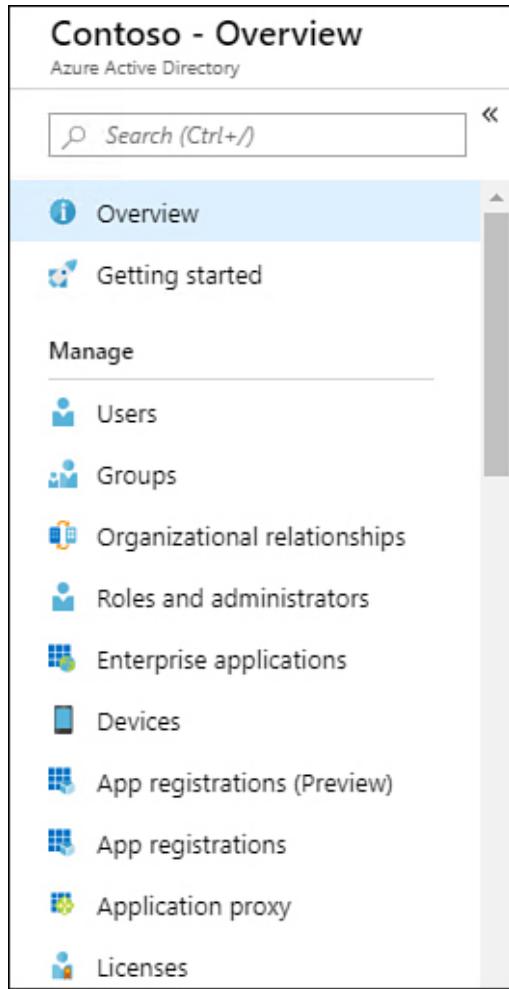


Figure 1-69 Azure AD Manage section

2. On the **Enterprise Applications** blade, ensure that **All Applications** is selected, as shown in Figure 1-70, and then select the application to which you want to enable user access.

The screenshot shows the 'Enterprise applications - All applications' blade. The left sidebar has options: 'Overview' (selected), 'Manage', 'All applications' (selected), 'Application proxy', 'User settings', 'Security', and 'Conditional Access'. The main area has a search bar and a table with columns: NAME, HOMEPAGE URL, OBJECT ID, and APPLICATION ID. Two applications are listed:

NAME	HOMEPAGE URL	OBJECT ID	APPLICATION ID
Box	https://www.box.com/	50d91d99-c0c0-4682-abdb-62...	f6656aaf-bb14-48fe-a...
BrowserStack	https://www.browserstack.com	7bdcc572-05e4-4721-8254-e7...	f52d241-8fa1-49ff-8b...

Figure 1-70 All Applications

3. Once the application opens, click **Users And Groups** from the application's navigation pane, which is shown in Figure 1-71.

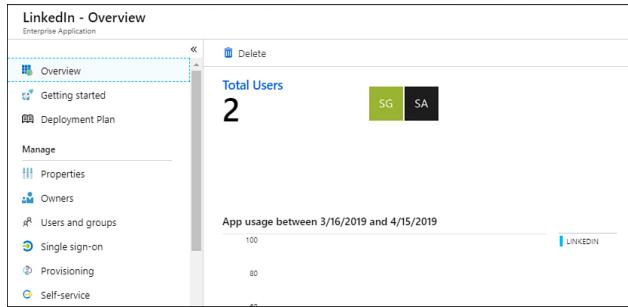


Figure 1-71 Application overview

- On the application's **Users And Groups** page, shown in Figure 1-72, click **Add User**. Note that you use the **Add User** button to also add a group assignment if Azure AD is licensed at the appropriate level.

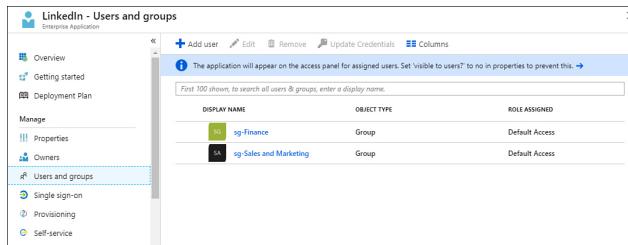


Figure 1-72 Users And Groups

- On the **Add Assignment** page shown in Figure 1-73, search for the user or group to which you want to grant application access.

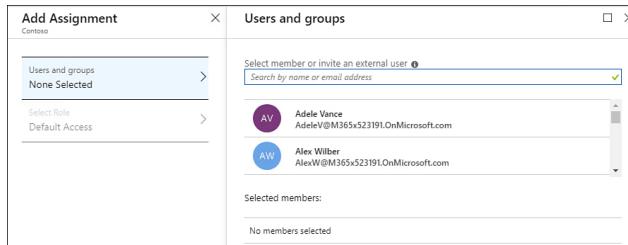


Figure 1-73 Add Assignment for Users And Groups

- Select a user or group and then click **Select**, as shown in Figure 1-74.

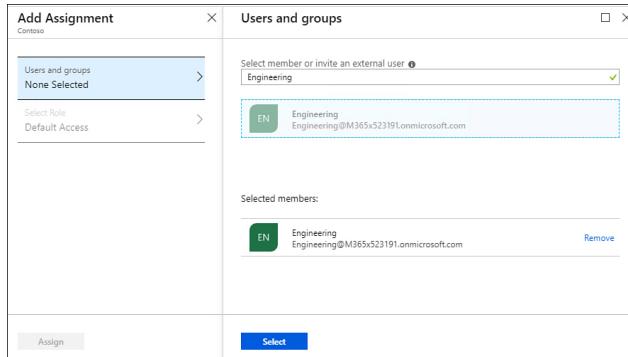


Figure 1-74 Selecting the group assignment

- Once the user or group is selected, click **Assign**. Verify that the assignment has occurred by reviewing the newly updated list of users and groups, as shown in Figure 1-75.

LinkedIn - Users and groups		
	+ Add user	Edit Remove Update Credentials Columns
Overview		
Getting started		
Deployment Plan		
Manage		
Properties		
Owners		
Users and groups		
First 100 shown; to search all users & groups, enter a display name.		
DISPLAY NAME	OBJECT TYPE	ROLE ASSIGNED
sg-Finance	Group	Default Access
sg-Sales and Marketing	Group	Default Access
Engineering	Group	Default Access

Figure 1-75 Users And Groups

More Info Assign Users and Groups Access

You can learn more about this topic at <https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/methods-for-assigning-users-and-groups>.

Configure app registration permission scopes

Configuring application registration permission scopes controls what information an application has access to. The Microsoft identity platform's way of implementing OpenID Connect uses several scopes that correspond to the Microsoft Graph. When configuring app registration, you can use the following permission scopes to determine what information the application can access:

- OpenID** Use this scope if an application performs a sign-in using OpenID Connect. This permission grants an app a unique identifier for the user in the form of a subclaim and also gives the app access to the `Userinfo` endpoint. This scope is used when interacting with the Microsoft identity platform to acquire ID

tokens, which can then be used by the application for authentication.

- **Email** The email scope gives the app access to a user's email address in the form of an email address associated with a user account.
- **Profile** The profile scope can be used to provide the application with information about the user. This may include a user's given name, surname, preferred username, and object ID.
- **Offline_access** The offline_access scope will provide an app access to resources on behalf of the user for an extended period. If a user consents to the offline_access scope, the app can receive a long-lived refresh token, which can be updated as older tokens expire.

More Info Permissions and Consent

You can learn more about permissions and consent in a Microsoft identity platform endpoint at <https://docs.microsoft.com/en-us/azure/active-directory/develop/v2-permissions-and-consent>.

Manage app registration permission consent

App registration permission consent allows users and administrators to control how and what data can be accessed by applications. The Microsoft identity platform supports the following types of permissions:

- **Delegated permissions** These permissions are used by apps that are leveraged by a signed-in user. The user or an administrator consents to the permissions required by the app. The app then uses a delegated permission to function as the signed-in user when attempting to access the target resource.
- **Application permissions** These permissions are used by apps that execute without a signed-in user. These might be long-running background applications. Application permissions can only be consented to by an administrator.

Effective permissions are the least-privileged set of permissions calculated when comparing the permissions that the application has been granted directly and the permissions of the signed-in user. To configure a list of statically requested permissions for an application, perform the following steps:

1. On the **App Registrations** blade of the Azure Active Directory console, select the registered application or which you want to configure static permissions.

2. Under **Manage**, click **API Permissions**, as shown in Figure 1-76.

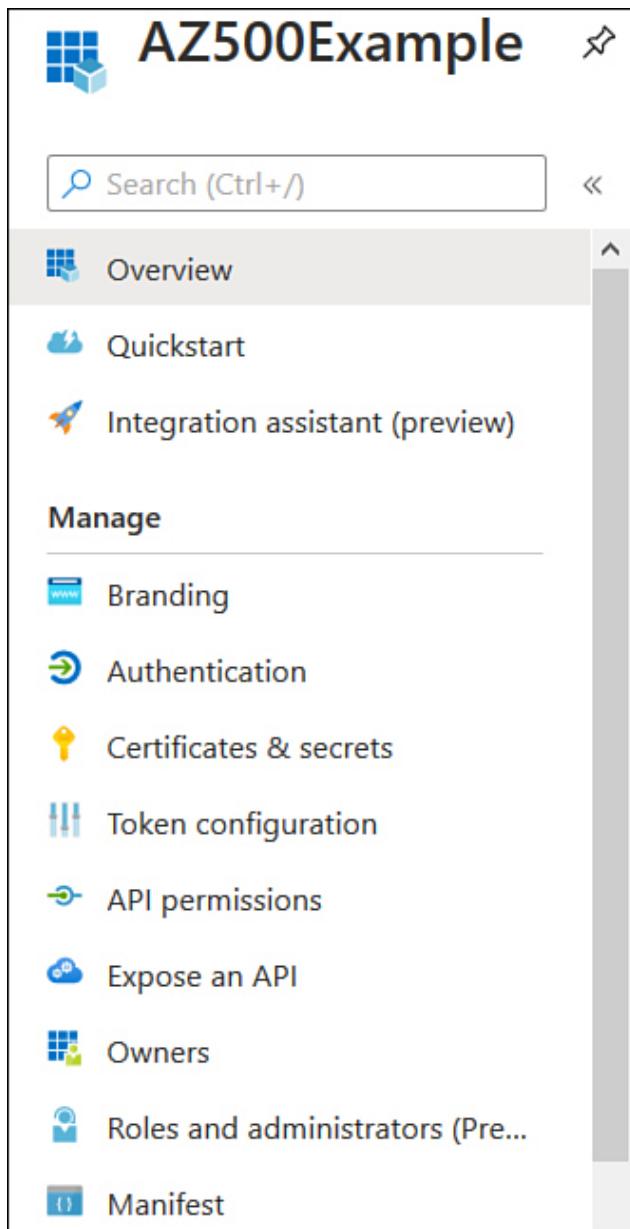


Figure 1-76 API Permissions on the Manage menu of a registered app

3. On the API Permissions blade shown in Figure 1-77, configure which permissions you would like the application to have. You can use this page to add permissions or to grant admin consent. Admin consent allows you to grant the application permissions to a specific Azure AD tenancy.

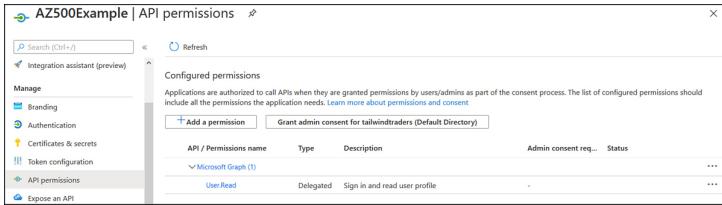


Figure 1-77 Manage API Permissions

More Info App Registration Permission and Consent

You can learn more about app registration permission and consent at:
<https://docs.microsoft.com/en-us/azure/active-directory/develop/v2-permissions-and-consent>.

Manage API access to Azure subscriptions and resources

API management policies allow you to control the behavior of an API. An API management policy is a collection of statements that apply sequentially to requests to or responses from the API. For example, these policies include format conversion from XML to JSON or call rate limiting. Call rate limiting can be a useful way of ensuring that an API hosted in Azure doesn't get flooded by requests, which can lead to unusually high subscription charges. API management policies are XML documents that are divided into inbound, outbound, back end, and on-error sections.

API management policies are evaluated depending on the scope at which they apply. Policy scopes are evaluated in the following order:

1. Global scope
2. Product scope
3. API scope
4. Operation scope

You can view all policies that apply in the current scope by clicking **Recalculate Effective Policy For Selected Scope** in the API management policy editor.

To set or edit an Azure API management policy, perform the following steps:

1. In the Azure portal, select the APIM instance. On the **APIs** tab, select the imported API.
2. On the **Design** tab select the operation against which you want to apply the policy. You also have the option of applying the policy to all operations.
3. Click the </> (code editor) icon in the **Inbound Processing** or **Outbound Processing** sections.
4. Enter the desired policy code into the appropriate section of code.

More Info API Management Access Restriction Policies

You can learn more about API management access restriction policies at <https://docs.microsoft.com/en-us/azure/api-management/api-management-policies>.

SKILL 1.4: MANAGE ACCESS CONTROL

Access control is another term for assigning permissions to resources. In this section, you'll learn how to secure resources within an Azure subscription through the assignment of permissions, which is most easily done through assigning users to roles. To master this objective, you'll need to understand subscription and resource permissions, resource group permissions, custom RBAC roles, the principle of least privilege, how to interpret permissions, and how to check access.

This section covers the following topics:

- Configure subscription and resource permissions
- Configure resource group permissions
- Configure custom RBAC roles
- Apply the principle of least privilege
- Interpret permissions
- Check access

Configure subscription and resource permissions

Azure Role Based Access Control (RBAC) allows you to configure fine-grained access management to Azure resources. Using RBAC, you can control what a security principal can do and where the security principal can do it. You do this with a combination of security principals, roles, and scopes.

As you recall from earlier in the chapter, security principals are Azure objects that represent individuals, collections of individuals, applications, or services.

Security principals include

- **Individual people** These are represented as Azure AD users or user objects that are references within Azure AD from other tenancies.
- **Collections of individuals** These are represented as Azure AD groups.
- **Applications and services** These are represented as service principals or managed identities.

An RBAC role is a collection of permissions. Permissions can be thought of as a set of operations—such as read, write, and delete—that can be performed against the Azure object to which the role is assigned.

The scope is the boundary to which the permissions defined in the role apply. You can configure the scope for a role assignment to occur at the management group, subscription, resource group, or individual Azure resource level. Scope assignments function in a parent-child relationship, which means the assignment of permissions that occurs at the parent scope level is inherited at the child scope level. For example, if you configure the scope for a role assignment to be at the resource group level, all the resources within that group will have that role assignment.

Assigning permissions to Azure subscriptions and resources requires combining security principals that represent who you want to assign the permission to, the role definition that defines the permissions, and scope that defines where the permissions are assigned.

More Info Understanding Rbac

You can learn more about understanding RBAC at
<https://docs.microsoft.com/en-us/azure/role-based-access-control/overview>.

Manage admin roles

Azure Active Directory includes many roles that provide a variety of permissions to different aspects of Azure AD and Microsoft 365 workloads. These roles and the permissions they grant are listed in [Table 1-2](#):

Table 1-2 Azure AD Roles

Role	Description
Application Administrator	Can administer enterprise applications, application registrations, and application proxy settings.
Application Developer	Can create application registrations.
Authentication Administrator	Can view current authentication method settings. Can set or reset nonpassword credentials. Can force MFA on the next sign in.
Billing	Can purchase and manage subscriptions. Can manage support tickets and monitor service health.

Ad mi nist rat or	
Cloud Application Administrator	Can manage all aspects of enterprise applications and registrations but cannot manage the application proxy.
Cloud Device Administrator	Can enable, disable, and remove devices in Azure AD. Can view Windows 10 BitLocker Drive Encryption Keys through the Azure portal.
Compliance Administrator	Manage features in the Microsoft 365 compliance center, Microsoft 365 admin center, Azure, and Microsoft 365 Security and Compliance Center.
Conditional Access Administrator	Administrative rights over Azure AD conditional access configuration.
Cust	Manages Customer Lockbox requests. Can also enable and

to me r Loc kbo x acc ess app rov er	disable the Customer Lockbox feature.
De vic e Ad mi nist rat ors	Users assigned this role will become local administrators on all computers running Windows 10 that are joined to Azure AD.
Dir ect ory Rea der s	Role for applications that do not support consent framework. Should not be assigned to users.
Dir ect ory Syn chr oni zati on Acc ou nts	Assigned to the Azure AD Connect service and not used for user accounts.
Dir ect ory Wri ters	A legacy role assigned to applications that do not support the consent framework. Should only be assigned to applications, not user accounts.
Dy na mic s 365 Ad	Administrative access to Dynamics 365 Online.

mi nist rat or/ CR M Ad mi nist rat or	
Exc han ge Ad mi nist rat or	Administrative access to Exchange Online.
Glo bal Ad mi nist rat or / Co mp any Ad mi nist rat or	Administrative access to all Azure AD features. This includes administrative access to services that use Azure AD Identities including Microsoft 365 security center, Microsoft 365 compliance center, Exchange Online, SharePoint Online, and Skype for Business Online. The account used to sign up for the tenancy becomes the global administrator. Global administrators can reset the passwords of any user, including other global administrators.
Gu est Inv iter	Can manage Azure AD B2B guest user invitations.
Inf or ma tio n Pro tect ion Ad mi nist	Can manage all aspects of Azure Information Protection, including configuring labels, managing protection templates, and activating protection.

rat or	
Int une Ad mi nist rat or	Has full administrative rights to Microsoft Intune.
Lic ens e Ad mi nist rat or	Can manage license assignments on users and groups. Cannot purchase or manage subscriptions.
Me ssa ge Ce ntr e Rea der	Can monitor notification and Microsoft advisories in the Microsoft 365 Message Center.
Pas sw ord Ad mi nist rat or / Hel pde sk Ad mi nist rat or	Can perform the following tasks for all users except those who have administrative roles: <ul style="list-style-type: none"> • Change passwords • Invalidate refresh tokens • Manage service requests • Monitor service health
Po wer BI Ad mi nist	Has administrator permissions over Power BI.

rat or	
Pri vile ged Rol e Ad mi nist rat or	Can manage all aspects of Azure AD Privileged Identity Management. Can manage role assignments in Azure AD.
Re por ts Rea der	Can view reporting data in the Microsoft 365 reports dashboard.
Sec uri ty Ad mi nist rat or	Has administrator-level access to manage security features in the Microsoft 365 security center, Azure AD Identity Protection, Azure Information Protection, and Microsoft 365 Security and Compliance Center.
Sec uri ty Rea der	Has read-only access to security Microsoft 365-related security features.
Ser vic e Su ppo rt Ad mi nist rat or	Can open and view support requests with Microsoft for Microsoft 365-related services.
Sha reP oin t	Has global administrator permissions for SharePoint Online workloads.

Administrator	
Skype for Business / Lynch Administrator	Has global administrator permissions for Skype for Business workloads.
Teams Administrator	Can administer all elements of Microsoft Teams.
Teams Communications Administrator	Can manage Teams workloads related to voice and telephony, including telephone number assignment and voice and meeting policies.
Teams Communication Support	Can troubleshoot communication issues within Teams and Skype for Business. Can view details of call records for all participants in a conversation.

En gin eer	
Tea ms Co m mu nic atio ns Su ppo rt Spe cial ist	Can troubleshoot communication issues within Teams and Skype for Business. Can only view user details in the call for a specific user.
Use r Acc ou nt Ad mi nistrat or	Can create and manage user accounts. Can create and manage groups. Can manage user views and support tickets and can monitor service health.

More Info Azure AD Administrator Roles

You can learn more about Azure AD Administrator roles at
<https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles>.

Configure RBAC within Azure AD

Azure RBAC (Role Based Access Control) allows you to configure fine-grained access control to Azure resources, such as virtual machines and storage accounts. When you configure RBAC, you assign a role and a scope, with the scope being the resource you want to have managed. Azure RBAC includes more than 70 roles. Providing the details of all 70 is beyond the scope of this text, but there are 4 fundamental roles that people who are responsible for managing Microsoft 365 should be aware of. These

roles can be assigned to specific Azure subscriptions, resource groups, or resources:

- **Owner** Users who hold this role have full access to all resources within the scope of the assignment and can delegate access to others.
- **Contributor** Users who hold this role can create and manage resources within the scope of the assignment but cannot grant access to others.
- **Reader** Users who hold this role can view resources within the scope of the assignment but can't perform other tasks and cannot grant access to others.
- **User Access Administrator** Users who hold this role can manage user access to Azure resources within the scope of the assignment.

More Info Azure RBAC

You can learn more about Azure RBAC at docs.microsoft.com/en-us/azure/role-based-access-control/rbac-and-directory-admin-roles.

Delegate admin rights

To view which users are assigned a specific role, perform the following steps:

1. In the Azure AD admin center, select **Roles And Administrators**, as shown in Figure 1-78.

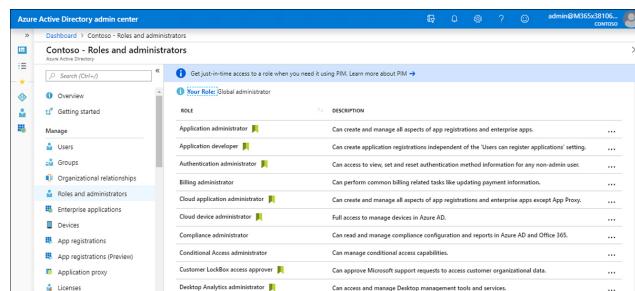


Figure 1-78 Roles And Administrators

2. To see the membership information of a role, click the role you want. Figure 1-79 shows members of the Password Administrators role.

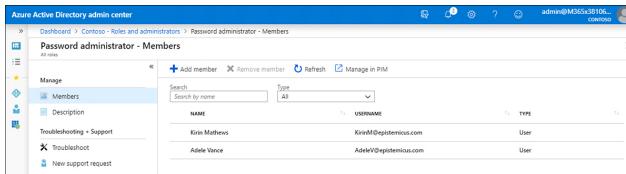


Figure 1-79 Members of the Password Administrators role

You can use the following Azure PowerShell cmdlets to view roles and role membership:

- **Get-AzureADDirectoryRole** View a list of Azure AD Directory roles
- **Get-AzureADDirectoryRoleMember** View the users assigned membership in an Azure AD Directory role

More Info Delegating Admin Rights

You can learn more about delegating admin rights at <https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/roles-concept-delegation>.

Manage role allocations by using Azure AD

To assign a user to a specific role within Azure AD, perform the following steps:

1. In the Azure AD admin center, select **Roles And Administrators**.
2. Select the role to which you want to add a user. This will open the role's properties page.
3. On the **Role Properties** page, click **Add Member**. Figure 1-80 shows adding the user Adele Vance to the Security Administrator role.

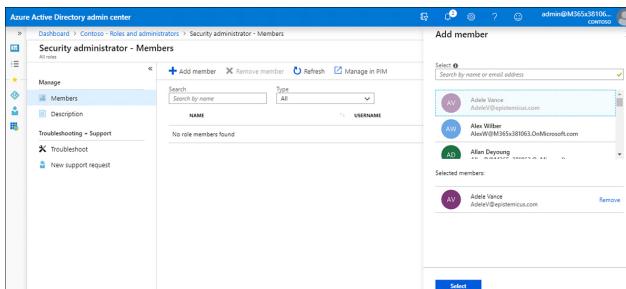


Figure 1-80 Members of the Security Administrators role

You can use the following Azure PowerShell cmdlets to manage role memberships:

- **Add-AzureADDirectoryRoleMember** Adds a user to an Azure AD Directory role
- **Remove-AzureADDirectoryRoleMember** Removes a user from an Azure AD Directory role

More Info View and Assign Azure AD Administrator Roles
You can learn more about viewing and assigning administrator roles at <https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-manage-roles-portal>.

Configure resource group permissions

Any permission assigned at the resource group level will apply to all resources stored within that resource group. For example, if you assign the virtual machine administrator role at the resource group level to a group of users, those users will have that role for all virtual machines stored within the resource group. To assign permissions at the resource group level, assign a specific role to a user, group, service principal, or managed identity. To assign a role at the resource group level, perform the following steps:

1. On the **Resource Groups** blade in the Azure portal, select the resource group for which you want to configure the permission, as shown in [Figure 1-81](#).

Name	Subscription	Location	...
cloud-shell-storage-southeastasia	Azure Pass - Sponsorship	Southeast Asia	...
ExampleRG	Azure Pass - Sponsorship	East US	...

Figure 1-81 Assigning roles at the resource group level

2. On the **Resource Groups** blade, click **Access Control (IAM)**.
3. On the **Access control (IAM)** page, choose **Add > Role Assignment**.
4. On the **Add Role Assignment** page, which is shown in [Figure 1-82](#), select the role that you want to assign, specify which user, group, service principal, or system managed identity you want the role to apply to, and then specify the identity of that security principal.

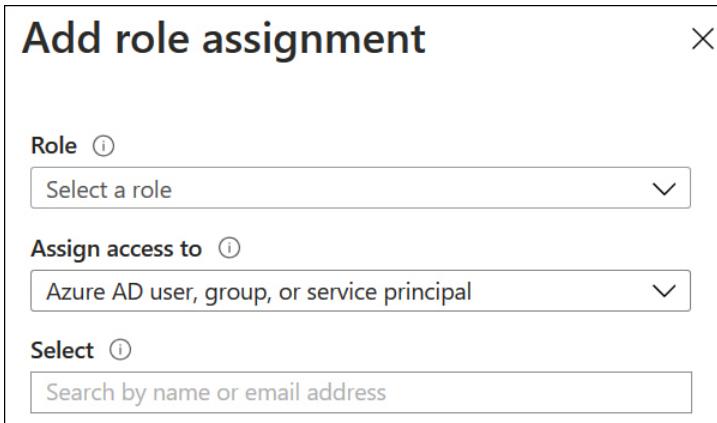


Figure 1-82 Add Role Assignment

More Info Resource Group Permissions

You can learn more about resource group permissions at
<https://docs.microsoft.com/en-us/rest/api/authorization/permissions/listforresourcegroup>.

Identify the appropriate role

There are a large number of preexisting roles available within Azure, and it is likely that an existing role will meet your needs, so you likely will not need to configure a custom role. First, you should specify exactly what actions a security principle should and should not be able to perform. Once you have generated this list, you should review the existing roles and determine if one of the existing roles meets your needs or if you need to create a custom role.

More Info Roles by Category

You can learn more about Azure RBAC roles by category at
<https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles>.

Apply the principle of least privilege

When configuring Azure RBAC, make sure that you follow the principle of least privilege. This means that you should only grant the access required to perform specific tasks. Doing so reduces the chance of unauthorized or accidental actions being performed. For

example, if a group only requires the ability to view the configuration of an Azure resource, you only need to assign a role that has the Read permission to that resource. If a group only requires Azure portal access to one virtual machine in a resource group (even though the resource group hosts multiple virtual machines), set the scope of the role assignment to the virtual machine rather than the resource group when assigning the role to that group.

More Info Azure Access Control Best Practices

You can learn more about Azure RBAC best practices, including least privilege, at <https://docs.microsoft.com/en-us/azure/security/fundamentals/identity-management-best-practices>.

Configure custom RBAC roles

If one of the many existing RBAC roles doesn't meet your organization's requirements, you can create a custom RBAC role. For example, there are three RBAC roles related to virtual machines: Virtual Machine Administrator Login, Virtual Machine Contributor, and Virtual Machine Users Login. If you want to allow a user to restart a VM but not log in to the VM or delete the VM, you could create a custom RBAC role that allows that specific permission. As with existing Azure RBAC roles, you can assign custom roles to users, groups, service principals, and managed identities at the management group, subscription, resource group, and individual resource levels.

You can create a custom role through the Azure portal, Azure PowerShell, Azure CLI, or Azure REST API, or you can create an ARM Template. In general, creating a custom role involves following these basic steps:

1. Determine which method you will use to create the custom role.
Determine what permissions the role requires. You can learn what operations are available to define your permission by viewing the Azure Resource Manager resource provider operations. For management operations, these will be `Actions` or `NotActions`. For data operations, these will be `DataActions` or `NotDataActions`.

2. Create the role. You can do this by cloning an existing role and then making modifications or by creating a new role from scratch. The most straightforward method of doing this is through the Azure portal.
3. Test the custom role. Make sure that you test the role thoroughly to determine that it only allows what you want it to allow and doesn't have some unexpected permissions, such as allowing Wally the VM operator to type something in Cloud Shell that locks out every other user in the Azure AD tenancy.

When creating a custom RBAC role, remember to only add the fewest necessary privileges to the role. When you create a custom role, it will appear in the Azure portal with an orange—rather than blue—resource icon. Custom RBAC roles are available between subscriptions that are associated with the same Azure AD tenancy. Each Azure AD tenancy supports up to 5,000 custom roles.

To clone and then modify a role in the Azure portal, perform the following steps:

1. In the Azure portal, open the **Access Control (IAM)** blade at the subscription level or resource group level where you want the custom role to be assignable.
2. Select the **Roles** tab to see the list of all available built-in and custom roles.
3. Select the role that you want to clone and then modify. Figure 1-83 shows the **Virtual Machine Contributor** role being selected for cloning.

<input checked="" type="checkbox"/>	Virtual Machine Contributor	BuiltinRole	0	0	...
<input type="checkbox"/>	Virtual Machine User Login	BuiltinRole	0	0	...
<input type="checkbox"/>	Web Plan Contributor	BuiltinRole	0	0	...
<input type="checkbox"/>	Website Contributor	BuiltinRole	0	0	...
<input type="checkbox"/>	Workbook Contributor	BuiltinRole	0	0	...
<input type="checkbox"/>	Workbook Reader	BuiltinRole	0	0	...

Figure 1-83 Select a role to clone

4. On the **Basics** tab of the **Create A Custom Role** page shown in Figure 1-84, provide a **Custom Role Name**.

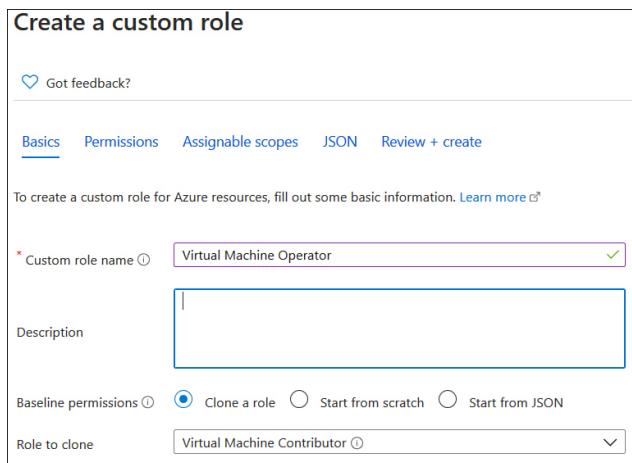


Figure 1-84 Create A Custom Role wizard with the Basics tab selected

5. On the **Permissions** tab shown in Figure 1-85, you can delete existing permissions or add new permissions.

Permission	Description	Permission type
Microsoft.Authorization/*/read	--	Action
Microsoft.Compute/availabilitySets/*	--	Action
Microsoft.Compute/locations/*	--	Action
Microsoft.Compute/virtualMachines/*	--	Action
Microsoft.Compute/virtualMachineScaleSet...	--	Action
Microsoft.Compute/disks/write	Creates a new Disk or updates an existing ...	Action
Microsoft.Compute/disks/read	Get the properties of a Disk	Action

Figure 1-85 Create A Custom Role wizard with the Permissions tab selected

6. On the **Assignable Scopes** tab, you can specify where the role can be assigned. You can select subscriptions associated with the Azure AD tenancy, as well as resource groups that are contained within those subscriptions.
7. On the **JSON** tab, you can view the custom role formatted in JSON. This tab gives you the opportunity to edit the role in JSON. If you want to add a wildcard permission, you do so on this tab because this is not possible at other points during the creation of a custom role.
8. Once you have reviewed the JSON code, click **Review And Create** to create the custom role.

More Info Azure Custom Roles

You can learn more about Azure custom roles at
<https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles>.

Interpret permissions

The key to understanding what can be done with permissions is that there are permissions related to management operations and permissions related to data operations. For management plane operations, the permissions determine actions that can be taken against objects in the Azure management plane, which includes the Azure portal, Azure CLI, Azure PowerShell, and Azure REST API. These are defined as `Actions` and `NotActions`. At the data operations level, there are actions that can be taken against data, such as data stored within a storage account. These are defined as `DataActions` and `NotDataActions`. To list the permissions within a role, use the `Get-AzRoleDefinition` PowerShell cmdlet. For example, to view the permissions associated with the Contributor role, run the following command:

[Click here to view code image](#)

```
Get-AzRoleDefinition "Contributor" | FL Actions,  
NotActions
```

Permissions are cumulative. If a user is granted `Actions` or `DataActions` across multiple roles and scopes, all permissions will apply. When multiple roles apply to a security principal, any `NotActions` or `NotDataActions` that apply will override any `Actions` or `DataActions` that apply.

More Info Management and Data Operations

You can learn more about management and data operations at <https://docs.microsoft.com/en-us/azure/role-based-access-control/role-definitions#management-and-data-operations>.

Check access

To view the access that a user has to a specific resource, perform the following steps:

1. In the Azure portal, select the specific resource for which you want to check access.
2. Select **Access control (IAM)** to open the Access Control (IAM) blade.
3. Click the **Check Access** tab.
4. In the **Check Access** section, use the **Find** drop-down menu to select the Azure AD user, group, or service principal option and type the name of the user whose access you want to check, as shown in Figure 1-86. Select the user.

The screenshot shows the 'Check access' tab selected in the top navigation bar. Below it, there's a search bar with 'Azure AD user, group, or service princip...' and a dropdown showing 'gamma'. To the right, there's a card titled 'Add a role assignment' with a green checkmark icon and a button labeled 'Add'. At the bottom, there's a link 'View role assignments'.

Figure 1-86 The Check Access tab

5. On the **Assignments** tab shown in Figure 1-87, review the user's role assignments and deny assignments to the resource.

The screenshot shows the 'Gamma User assignments - TT-VMs' blade. It displays a table of role assignments for the user 'gamma'. There is one entry: 'Virtual Machine Contributor' with a description 'Lets you manage virtual machin...' and a scope 'This resource'. Below the table, there are sections for 'Deny assignments (0)' and 'Classic administrators (0)'.

Figure 1-87 The Role assignments tab

More Info View User Access to Resources
You can learn more about View user access to resources at <https://docs.microsoft.com/en-us/azure/role-based-access-control/check-access>.



Exam Tip

Remember to always apply the principal of least privilege when attempting to determine which

role to assign to a user who needs access to a resource.



Thought Experiment

In this thought experiment, demonstrate your skills and knowledge of the topics covered in this chapter. You can find answers to this thought experiment in the next section.

Identity and access at Tailwind Traders

You are one of the Azure administrators for Tailwind Traders, an online general store that specializes in a variety of products used around the home. As a part of your duties for Tailwind Traders, you have registered a new application with your Azure AD instance. Even though the application is registered, you want to limit what actions the application can perform against resources in the Tailwind Traders Azure Subscription by applying a custom RBAC role. Tailwind Traders has been using PIM for some time as a method of improving security to resources within subscriptions owned by the organization. Because the access was configured some time ago, you are aware that several users who were configured as eligible for PIM roles have changed job roles. To improve security, you want to remove PIM eligibility if it is no longer required. Another goal of Tailwind Traders is to allow some users of the new application to access the application from outside the workplace. However, from a security perspective, anyone accessing the application from outside the Tailwind Traders internal network should take extra steps to verify their identity. With this

information in mind, answer the following questions:

- 1.** How can you assign the custom RBAC role to the new application?
- 2.** How can you determine which staff to remove from eligibility for PIM roles?
- 3.** How can you ensure that all users perform MFA if they are accessing the new application from a location outside the Tailwind Traders office?

THOUGHT EXPERIMENT ANSWERS

This section contains the solution to the thought experiment. Each answer explains why the answer choice is correct.

- 1.** You can assign roles to the new application by assigning roles to the service principal created when the application was registered. By assigning the custom RBAC role to the service principal, you assign that role to the application.
- 2.** You should configure an access review to determine which users that have been configured as eligible for PIM roles aren't actually using those roles.
- 3.** You can configure a conditional access policy to force users to perform MFA when they are in an untrusted location, such as any network location outside the trusted networks identified as belonging to Tailwind Traders.

CHAPTER SUMMARY

- Security principals are created automatically when you register an application with Azure AD.
- You can assign RBAC roles to security principals as a way of assigning permissions to applications.
- Azure AD groups allow you to collect Azure security principals including users, service principals, and other groups.
- Azure AD users represent individuals within Azure AD. They can be cloud only accounts or can be replicated from an on-premises Active Directory Domain Services environment.
- Password writeback allows passwords changed within Azure AD to be written back to an Active Directory Domain Services environment.

- Privileged Identity Management allows just-in-time administration and just-in-time access to Azure resources.
- Conditional Access Policies allow you to implement more stringent authentication requirements if certain conditions are met.
- Application registration permission scopes allow you to control what resources and data an application can access.
- Custom RBAC roles can be configured if an existing RBAC role does not have permissions that are appropriate to your organization's needs.

Chapter 2

Implement platform protection

One of the main aspects of cloud computing is the shared responsibility model, where the cloud solution provider (CSP) and the customer share different levels of responsibilities, depending on the cloud service category.

When it comes to platform security, Infrastructure as a Service (IaaS), customers will have a long list of responsibilities. However, in a Platform as a Service (PaaS) scenario there are still some platform security responsibilities, they are not as extensive as when using IaaS workloads.

Azure has native platform security capabilities and services that should be leveraged to provide the necessary level of security for your IaaS and PaaS workloads while maintaining a secure management layer.

Skills in this chapter:

- Skill 2.1: Implement advanced network security
- Skill 2.2: Configure advanced security for compute

SKILL 2.1: IMPLEMENT ADVANCED NETWORK SECURITY

To implement an Azure network infrastructure, you need to understand the different connectivity options available in Azure. These options will enable you to implement a variety of scenarios with different requirements. This section of the chapter covers the skills necessary to implement advanced network security.

Overview of Azure network components

Azure networking provides built-in capabilities to enable connectivity between Azure resources, connectivity from on-premises networks to Azure resources, and branch office to branch office connectivity in Azure.

While those skills are not directly called out in the AZ-500 exam outline, it is important for you to understand these concepts. If you’re already comfortable with your skill level, you can skip to “Secure the connectivity of virtual networks,” later in this chapter.

To better understand the different components of an Azure network, let’s review Contoso’s architecture diagram shown in Figure 2-1.

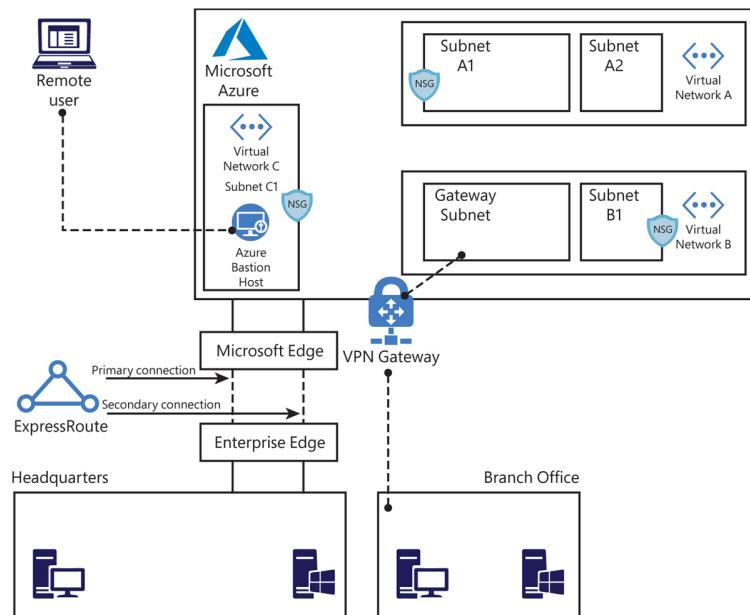


Figure 2-1 Contoso network diagram

In Figure 2-1, you can see Azure infrastructure (on top), with three virtual networks. Contoso needs to segment its Azure network in different virtual networks (VNets) to provide better isolation and security. Having VNets in its Azure infrastructure allows Contoso to connect Azure Virtual Machines (VMs) to securely communicate with

each other, the Internet, and Contoso's on-premises networks.

If you think about the traditional physical network on-premises where you operate in your own data center, that's basically what VNet is, but with it additional benefits of Azure's infrastructure, which includes scalability, availability, and isolation. When you are creating a VNet, you must specify a custom private IP address that will be used by the resources that belong to this VNet. For example, if you deploy a VM in a VNet with an address space of 10.0.0.0/24, the VM will be assigned a private IP, such as 10.0.0.10/24.

Important Multiple Vnets and Virtual Network Peering

An Azure VNet is scoped to a single region/location. If you need to connect multiple virtual networks from different regions, you can use Virtual Network Peering.

Notice in [Figure 2-1](#) that there are subnets in each VNet in Contoso's network. Contoso needs to segment the virtual network into one or more subnetworks and allocate a portion of the virtual network's address space to each subnet. With this setup, Contoso can deploy Azure resources in a specific subnet, just like it used to do in its on-premises network. From an organizational and structure perspective, subnets have allowed Contoso to segment its VNet address space into smaller segments that are appropriate for its internal network. By using subnets, Contoso also was able to improve address allocation efficiency.

Another important trio of component is shown in [Figure 2-1](#): subnets A1, B1, and C1. Each one of these subnets has a network security group (NSG) bound to it, which provides an extra layer of security based on rules that allow or deny inbound or outbound network traffic.

NSG security rules are evaluated by their priority, and each is identified with a number between 100 and 4096,

where the lowest numbers are processed first. The security rules use 5-tuple information (source address, source port, destination address, destination port, and protocol) to allow or deny the traffic. When the traffic is evaluated, a flow record is created for existing connections and the communication is allowed or denied based on the connection state of the flow record. You can compare this type of configuration to the old VLAN segmentation that was often implemented with on-premises networks.

Important Traffic Interruptions Might not be Interrupted

Existing connections might not be interrupted when you remove a security rule that enabled the flow. An interruption of traffic occurs when connections are stopped, and no traffic is flowing in either direction for at least a few minutes.

Contoso is headquartered in Dallas, and it has a branch office in Sydney. Contoso needs to provide secure and seamless RDP/SSH connectivity to its virtual machines directly from the Azure portal over TLS. Contoso doesn't want to use jumpbox VMs and instead wants to allow remote access to back-end subnets through the browser. For this reason, Contoso implemented Azure Bastion, as you can see in the VNet C, subnet C1 in [Figure 2-1](#).

Azure Bastion is a platform-managed PaaS service that can be provisioned in a VNet.

For Contoso's connectivity with Sydney's branch office, it is using a VPN gateway in Azure. A virtual network gateway in Azure is composed of two or more VMs that are deployed to a specific subnet called gateway subnet. The VMs that are part of the virtual network gateway contain routing tables and run specific gateway services. These VMs are automatically created when you create the virtual network gateway, and you don't have direct access to those VMs to make custom configurations to the operating system.

When planning your VNets, consider that each VNet may only have one virtual network gateway of each type, and the gateway type may only be VPN or ExpressRoute. Use VPN when you need to send encrypted traffic across the public Internet to your on-premises resources.



Exam Tip IP Address Configuration

When taking the exam, pay extra attention to scenarios that include IP addresses for different subnets and potential connectivity issues because of incorrect IP configuration.

For example, let's say that Contoso needs a faster, more reliable, secure, and consistent latency to connect its Azure network to its headquarters in Dallas. Contoso decides to use ExpressRoute, as shown in [Figure 2-1](#). ExpressRoute allows Contoso to extend its on-premises networks into the Microsoft cloud (Azure or Office 365) over a private connection because ExpressRoute does not go over the public Internet.

In [Figure 2-1](#), notice that the ExpressRoute circuit consists of two connections, both of which are Microsoft Enterprise Edge Routers (MSEEs) at an ExpressRoute Location from the connectivity provider or your network edge. While you might choose not to deploy redundant devices or Ethernet circuits at your end, the connectivity providers use redundant devices to ensure that your connections are handed off to Microsoft in a redundant manner. This Layer 3 connectivity redundancy is a requirement for Microsoft SLA to be valid.

Network segmentation is important in many scenarios, and you need to understand the design requirements to suggest the implementation options. Let's say you want to ensure that Internet hosts cannot communicate with

hosts on a back-end subnet but can communicate with hosts on the front-end subnet. In this case, you should create two VNets: one for your front-end resources and another for your back-end resources.

When configuring your virtual network, also take into consideration that the resources you deploy within the virtual network will inherit the capability to communicate with each other. You can also enable virtual networks to connect to each other, or you can enable resources in either virtual network to communicate with each other by using virtual network peering. When connecting virtual networks, you can choose to access other VNets that are in the same or in different Azure regions. Follow the steps below to configure your virtual network using the Azure portal:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar type **virtual networks** and under **Services**, click **Virtual Networks**. The **Virtual Networks** page appears, as shown in Figure 2-2.

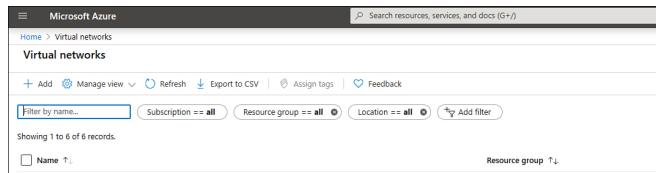


Figure 2-2 Azure Virtual Networks page

3. Click the **Add** button and the **Create Virtual Network** page appears, as shown in Figure 2-3.

A screenshot of the 'Create virtual network' page. The top navigation bar shows 'Home > Virtual networks > Create virtual network'. The main title is 'Create virtual network'. Below it, there are tabs for 'Basics' (which is selected), 'IP Addresses', 'Security', 'Tags', and 'Review + create'. A descriptive paragraph explains what a VNet is. Under 'Project details', there are fields for 'Subscription' (set to 'Visual Studio Ultimate with MSDN') and 'Resource group' (with a dropdown menu and a 'Create new' button). Under 'Instance details', there are fields for 'Name' and 'Region' (set to '(US) West US').

Figure 2-3 The Create Virtual Network page allows you to customize your VNet deployment

4. On the **Basics** tab, select the **Subscription** for the VNet and the **Resource Group**.
5. In the **Name** field, type a comprehensive name for the VNet, and in the **Region** field, select the Azure region in which the VNet is going to reside. Finally, click the **IP Addresses** tab.
6. On the **IP Addresses** page, in the **IPv4** field, type the address space in classless inter-domain routing (CIDR) format; for example, you could enter **10.3.0.0/16**.
7. Click the **Add Subnet** button. The **Add Subnet** blade appears, as shown in Figure 2-4.

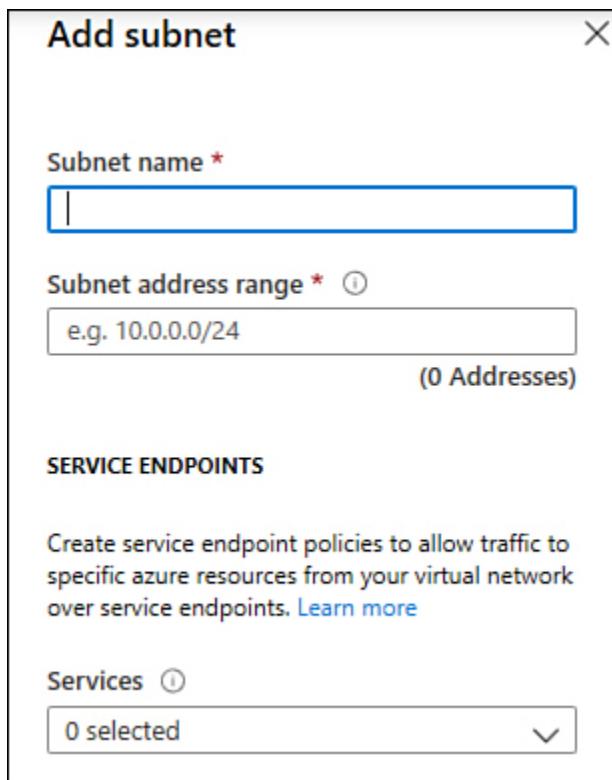


Figure 2-4 Add Subnet blade

8. In the **Subnet Name** field, type a name for this subnet.
9. In the **Subnet Address Range**, type the IP range for this subnet in CIDR format, such as **10.3.0.0/16**. Keep in mind that the smallest supported IPv4 subnet is /29, and the largest is /8.
10. Click the **Add** button; the subnet that you just created appears under the **Subnet Name** section.
11. Leave the default selections for now and click the **Review + Create** button. The validation result appears, which is similar to the one shown in Figure 2-5.

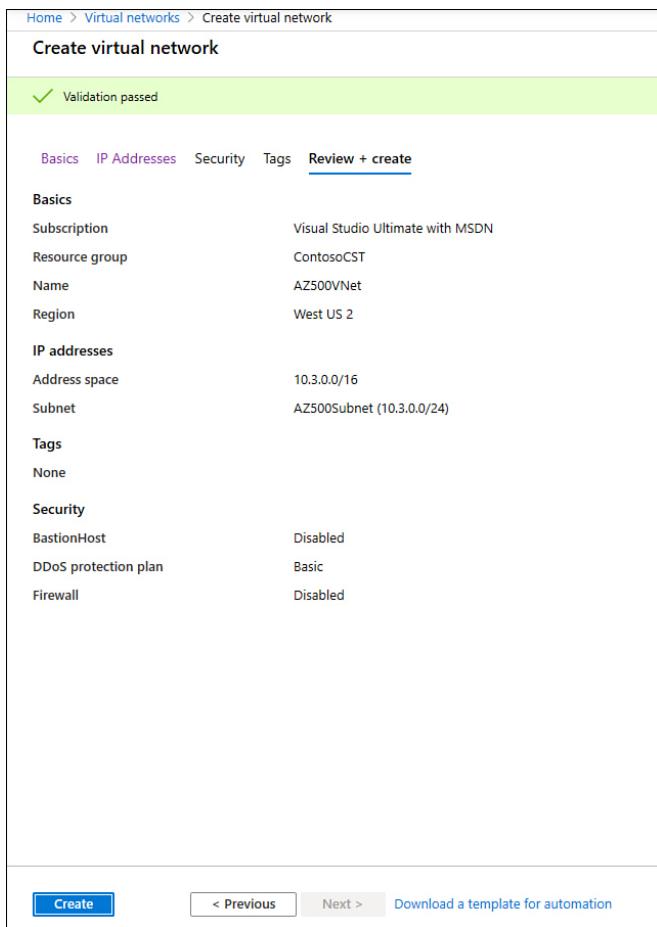


Figure 2-5 Summary of the selections with the validation results

12. Click the **Create** button.
13. The **Overview** page appears with the deployment final status. On this page, click the **Go To Resource** button and review these options on the left navigation pane: **Overview**, **Address Space**, and **Subnets**.

Notice that the parameters you configured during the creation of your VNet will be distributed among the different options on the VNet page. As you saw in the previous steps, creating a VNet using the Azure portal is a straightforward process, though in some circumstances, you might need to automate the creation process, and you can use PowerShell to do just that.

When you are creating your virtual network, you can use any IP range that is part of RFC 1918, which includes

- 224.0.0.0/4 (multicast)
- 255.255.255.255/32 (broadcast)
- 127.0.0.0/8 (loopback)
- 169.254.0.0/16 (link-local)
- 168.63.129.16/32 (internal DNS)

Also consider the following points:

- Azure reserves `x.x.x.0` as a network address and `x.x.x.1` as a default gateway.
- `x.x.x.2` and `x.x.x.3` are mapped to the Azure DNS IPs to the VNet space.
- `x.x.x.255` is reserved for a network broadcast address.

To automate that, you can either use PowerShell on your client workstation (using `Connect-AzAccount` to connect to your Azure subscription) or by using Cloud Shell directly from <https://shell.azure.com>. To create a virtual network using PowerShell, you need to use the `New-AzVirtualNetwork` cmdlet, as shown here:

[Click here to view code image](#)

```
$AZ500Subnet = New-AzVirtualNetworkSubnetConfig -  
Name AZ500Subnet -AddressPrefix  
"10.3.0.0/24"  
New-AzVirtualNetwork -Name AZ500VirtualNetwork -  
ResourceGroupName ContosoCST -Location  
centralus -AddressPrefix "10.3.0.0/16" -Subnet  
$AZ500Subnet
```

In this example, you have the `$AZ500Subnet` variable, which configures a new subnet for this VNet using the `New-AzVirtualNetworkSubnetConfig` cmdlet. Next, the `New-AzVirtualNetwork` cmdlet is used to create the new VNet, and it calls the `$AZ500Subnet` variable at the end of the command line to create the subnet.

After creating your VNet, you can start connecting resources to it. In an IaaS scenario, it is very common to connect your virtual machines (VMs) to the VNet.

Assuming you have Virtual Machine Contributor privileges in the subscription, you can quickly deploy a new VM the New-AzVM PowerShell cmdlet, as shown here:

[Click here to view code image](#)

```
New-AzVm ` 
    -ResourceGroupName "ContosoCST" ` 
    -Location "East US" ` 
    -VirtualNetworkName "AZ500VirtualNetwork" ` 
    -SubnetName "AZ500Subnet" ` 
    -Name "AZ500VM" `
```

Routing

In a physical network environment, you usually need to start configuring routes as soon as you expand your network to have multiple subnets. In Azure, the routing table is automatically created for each subnet within an Azure VNet. The default routes created by Azure and assigned to each subnet in a virtual network can't be removed. The default route that is created contains an address prefix and the next hop (where the package should go). When traffic leaves the subnet, it goes to an IP address within the address prefix of a route; the route that contains the prefix is the route used by Azure.

When you create a VNet, Azure creates a route with an address prefix that corresponds to each address range that you defined within the address space of your VNet. If the VNet has multiple address ranges defined, Azure creates an individual route for each address range. You don't need to worry about creating routes between subnets within the same VNet because Azure automatically routes traffic between subnets using the routes created for each address range. Also, differently from your physical network topology and routing mechanism, you don't need to define gateways for Azure to route traffic between subnets. In an Azure routing table, this route appears as:

- **Source** Default
- **Address prefix** Unique to the virtual network
- **Next hop type** Virtual network

If the destination of the traffic is the Internet, Azure leverages the system-default route `0.0.0.0/0` address prefix, which routes traffic for any address not specified by an address range within a virtual network to the Internet. The only exception to this rule is if the destination address is for one of Azure's services. In this case, instead of routing the traffic to the Internet, Azure routes the traffic directly to the service over Azure's backbone network. The other scenarios in which Azure will add routes are as follows:

- **When you create a VNet peering** In this case, a route is added for each address range within the address space of each virtual network peering that you created.
- **When you add a Virtual Network Gateway** In this case, one or more routes with a virtual network gateway listed as the next hop type are added.
- **When a VirtualNetworkServiceEndpoint is added** When you enable a service endpoint to publish an Azure service to the Internet, the public IP addresses of the services are added to the route table by Azure.

You might also see `None` in the **Next Hop Type** column, in the routing table. Traffic routed to this hop is automatically dropped. Azure automatically creates default routes for `10.0.0.0/8`, `192.168.0.0/16` (RFC 1918), and `100.64.0.0/10` (RFC 6598).



Exam Tip

The exam might include scenarios that involve routing-related problems. Make sure to pay close attention to the details about the routing configuration and whether any routing configurations are missing.

At this point, you might ask: “If all these routes are created automatically, in which scenario should I create a custom route?” You should do this only when you need to alter the default routing behavior. For example, if you add an Azure Firewall or any other virtual appliance, you can change the default route (`0.0.0.0/0`) to point to this virtual appliance. This will enable the appliance to inspect the traffic and determine whether to forward or drop the traffic. Another example is when you want to ensure that traffic from hosts doesn’t go to the Internet; you can control the routing rules to accomplish that.

To create a custom route that is effective for your needs, you need to create a custom routing table, create a custom route, and associate the routing table to a subnet, as shown in the PowerShell sequence that follows.

1. Create the routing table using `New-AzRouteTable` cmdlet, as shown here:

[Click here to view code image](#)

```
$routeTableAZ500 = New-AzRouteTable `  
    -Name 'AZ500RouteTable' `  
    -ResourceGroupName ContosoCST `  
    -location EastUS
```

2. Create the custom route using multiple cmdlets. First, you retrieve the route table information using `Get-AzRouteTable`, and then you create the route using `Add-AzRouteConfig`. Lastly, you use the `Set-AzRouteTable` to write the routing configuration to the route table:

[Click here to view code image](#)

```
Get-AzRouteTable `  
    -ResourceGroupName "ContosoCST" `  
    -Name "AZ500RouteTable" `  
    | Add-AzRouteConfig `  
    -Name "ToAZ500Subnet" `  
    -AddressPrefix 10.0.1.0/24 `  
    -NextHopType "MyVirtualAppliance" `  
    -NextHopIpAddress 10.0.2.4 `  
    | Set-AzRouteTable
```

3. Now that you have the routing table and the custom route, you can associate the route table with the subnet. Notice here that you first

write the subnet configuration to the VNet using the `Set-AzVirtualNetwork` cmd. After that you use `Set-AzVirtualNetworkSubnetConfig` to associate the route table to the subnet:

[Click here to view code image](#)

```
$virtualNetwork | Set-AzVirtualNetwork  
Set-AzVirtualNetworkSubnetConfig `  
    -VirtualNetwork $virtualNetwork `  
    -Name 'CustomAZ500Subnet' `  
    -AddressPrefix 10.0.0.0/24 `  
    -RouteTable $routeTableAZ500 | `  
Set-AzVirtualNetwork
```

Virtual network peering

When you have multiple VNets in your Azure infrastructure, you can connect those VNets using VNet peering. You can use VNet peering to connect VNets within the same Azure region or across Azure regions; doing so is called global VNet peering.

When the VNets are on the same region, the network latency between VMs that are communicating through the VNet peering is the same as the latency within a single virtual network. It's also important to mention that the traffic between VMs in peered virtual networks is not through a gateway or over the public Internet; instead, that traffic is routed directly through the Microsoft backbone infrastructure. To create a VNet peering using the Azure portal, follow these steps:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar type **virtual networks**, and under **Services**, click **Virtual Networks**.
3. Click the VNet that you want to peer, and on the left navigation pane, click **Peerings** (see Figure 2-6).

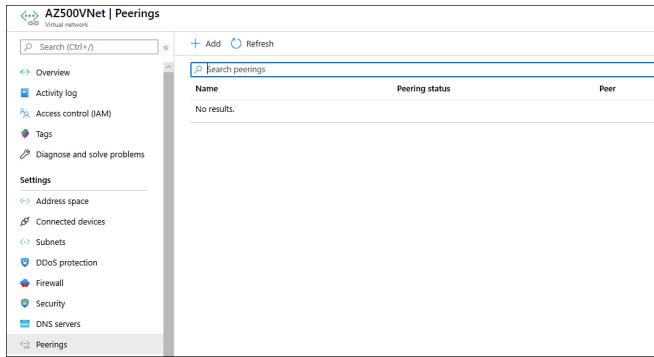


Figure 2-6 Configuring VNet peering

4. Click the **Add** button, and the **Add Peering** page appears, as shown in Figure 2-7.

Figure 2-7 Adding a new peering

5. In the **Name** field, type a name for this peering.
6. In the **Subscription** field, select the subscription that has the VNet to which you want to connect.
7. In the **Virtual Network** field, click the drop-down menu and select the VNet that you want to peer.
8. In the **Name Of The Peering From Remote Virtual Network** field, type the name that you want this peering

connection to appear on the other VNet.

9. The next two options—**Allow Virtual Network Access From [VNet name] To Remote Virtual Network** and **Allow Virtual Network Access From Remote Virtual To [VNet name]**—are used to control the communication between those VNets. If you want full connectivity from both directions, make sure to leave the **Enabled** option selected (default selection) for both. Enabling communication between virtual networks allows resources connected to either virtual network to communicate with each other with the same bandwidth and latency as if they were connected to the same virtual network.
10. The next two options—**Allow Forwarded Traffic From Remote Virtual Network To [VNet name]** and **Allow Forwarded Traffic From [VNet name] To Remote Virtual Network**—are related to allowing forwarded traffic. You should select **Enable** for both settings only when you need allow traffic that didn't originate from the VNet to be forwarded by a network virtual appliance through a peering. For example, consider three virtual networks named VNetTX, VNetWA, and MainHub. A peering exists between each spoke VNet (VNetTX and VNetWA) and the Hub virtual network, but peerings don't exist between the spoke VNets. A network virtual appliance is deployed in the Hub VNet, and user-defined routes can be applied to each spoke VNet to route the traffic between the subnets through the network virtual appliance. If this option is disabled, there will be no traffic flow between the two spokes through the hub.

11. Click **OK** to finish the configuration.

To configure a VNet peering using PowerShell, you just need to use the `Add-AzVirtualNetworkPeering` cmdlet, as shown here:

[Click here to view code image](#)

```
Add-AzVirtualNetworkPeering -Name  
'NameOfTheVNetPeering' -VirtualNetwork SourceVNet  
-RemoteVirtualNetworkId RemoteVNet
```

A peered VNet can have its own gateway, and the VNet can use its gateway to connect to an on-premises network. One common use of VNet peering is when you are building a hub-spoke network. In this type of topology, the hub is a VNet that acts as a central hub for connectivity to your on-premises network. The spokes are VNets that are peering with the hub, allowing them

to be isolated, which increases their security boundaries. An example of this topology is shown in [Figure 2-8](#).

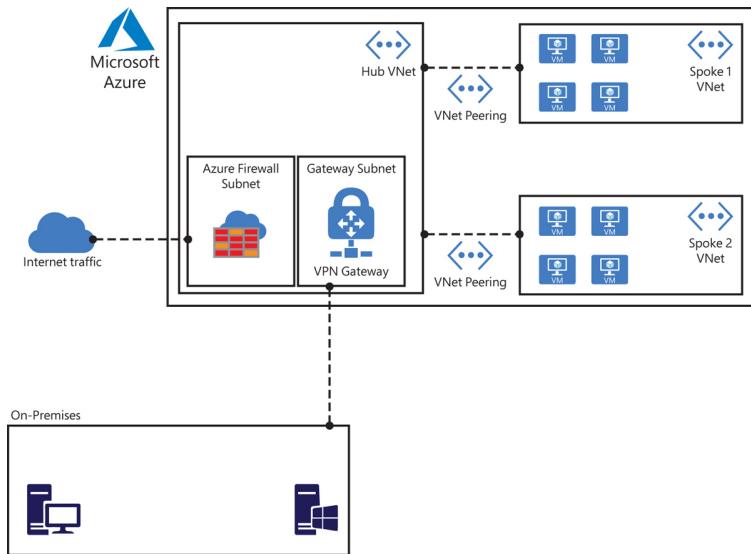


Figure 2-8 Hub-spoke network topology using VNet peering

A hybrid network uses the hub-spoke architecture model to route traffic between Azure VNets and on-premises networks. When there is a site-to-site connection between the Azure VNet and the on-premises data center, you must define a gateway subnet in the Azure VNet. All the traffic from the on-premises data center would then flow via the gateway subnet.

Network address translation

Azure has a Virtual Network NAT (network address translation) capability that enables outbound-only Internet connectivity for virtual networks. This is a common scenario when you want that outbound connectivity to use a specified static public IP address (static NAT) or you want to use a pool of public IP addresses (Dynamic NAT).

Keep in mind that outbound connectivity is possible without the use of an Azure load balancer or a public IP address directly attached to the VM. [Figure 2-9](#) shows an example of the topology with a NAT Gateway.

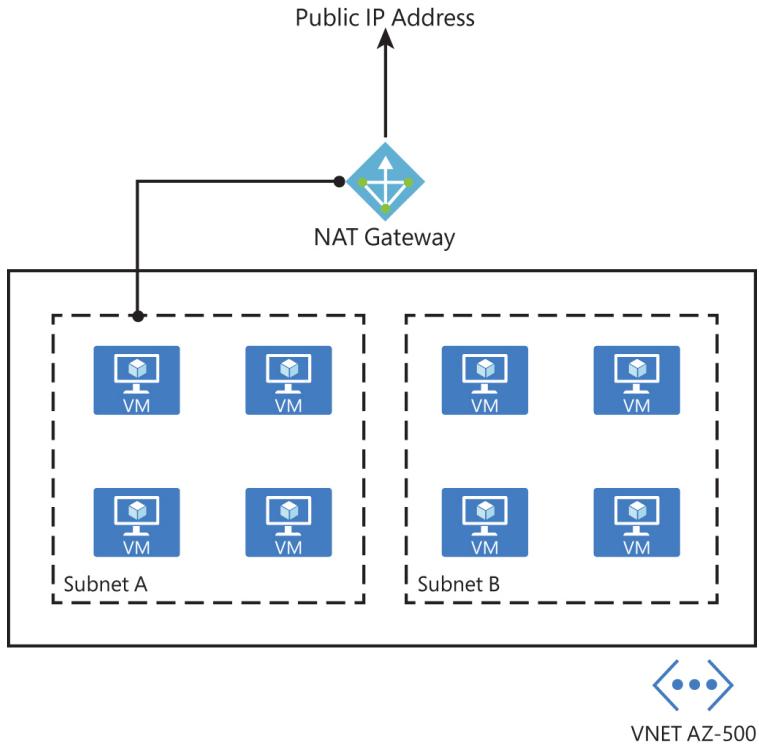


Figure 2-9 NAT Gateway topology

You can implement NAT by using a public IP prefix directly, or you can distribute the public IP addresses of the prefix across multiple NAT gateway resources. NAT also changes the network route because it takes precedence over other outbound scenarios, and it will replace the default Internet destination of a subnet. From an availability standpoint (which is critical for security), NAT always has multiple fault domains, which means it can sustain multiple failures without service outage.

Important Nat Gateway Billing

A NAT gateway is billed with two separate meters: resource hours and data processed. Consult the Azure NAT pricing page for the latest pricing.

To create a NAT Gateway for your subnet, you first need to create a public IP address and a public IP prefix. Follow the steps below to perform these tasks:

1. Navigate to the Azure portal at <https://portal.azure.com>.

2. In the main dashboard, click the **Create A Resource** button.
3. On the **New** page, type **Public IP** and click the **Public IP Address** option that appears in the list.
4. On the **Public IP Address** page, click the **Create** button; the **Create Public IP Address** page appears, as shown in Figure 2-10.

Create public IP address

IP Version *

IPv4 IPv6 Both

SKU *

Basic Standard

IPv4 IP Address Configuration

Name *

IP address assignment *
 Dynamic Static

Idle timeout (minutes) *
 4

DNS name label

.eastus.cloudapp.azure.com

Subscription *

Resource group *

[Create new](#)

Location *

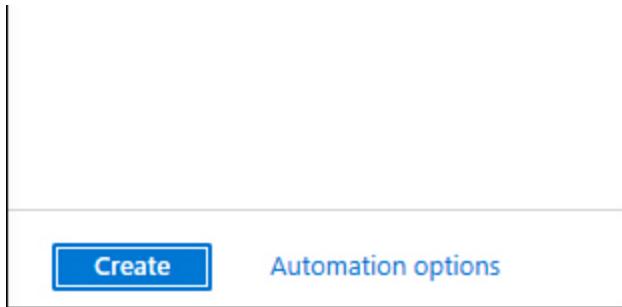


Figure 2-10 Creating a public IP address to be used by NAT Gateway

5. Type the name for this public IP address and select the subscription, resource group, and the Azure location. For this example, you can leave all other options with their default selections. Once you finish, click the **Create** button.
6. Now you should repeat steps 1 and 2. In the third step, type **public IP prefix** and click the **Public IP Prefix** option that appears in the drop-down menu.
7. On the **Create A Public IP Prefix** page, configure the following relevant options:
 1. Select the appropriate **Subscription**.
 2. Select the appropriate **Resource Group**.
 3. Type the **Prefix Name**.
 4. Select the appropriate **Azure Region**.
 5. In the **Prefix Size** drop-down menu, select the appropriate size for your deployment.
8. Once you finish configuring these options, click the **Review + Create** button and click **Create** to finish.
9. Now that you have the two requirements fulfilled, you can create the NAT Gateway.
10. Navigate to the Azure portal at <https://portal.azure.com>.
11. In the main dashboard, click the **Create A Resource** button.
12. On the New page, type **NAT Gateway** and click the **NAT Gateway** option in the list.
13. On the **NAT Gateway** page, click **Create**. The **Create Network Address Translation (NAT) Gateway** page appears, as shown in Figure 2-11.

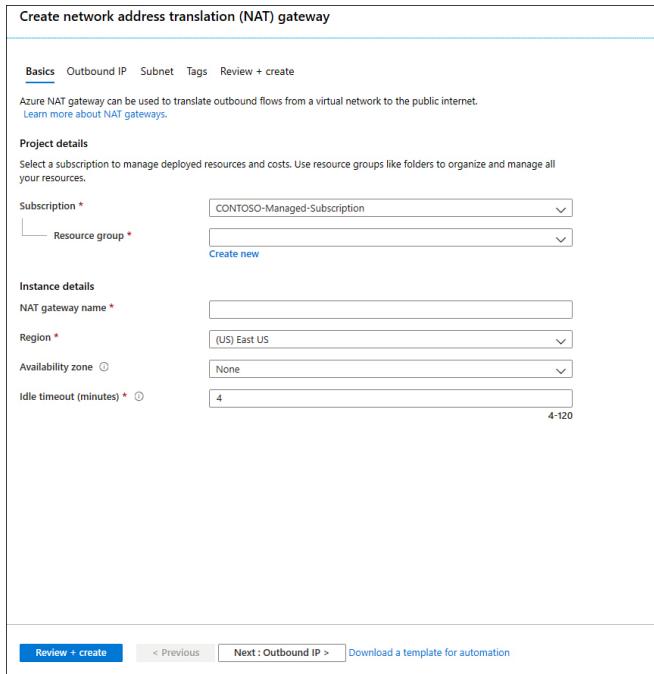


Figure 2-11 Creating a NAT Gateway in Azure

14. On the **Basics** tab, make sure to configure the following options:
 1. Select the appropriate **Subscription** and **Resource Group**.
 2. Type the **NAT Gateway Name**.
 3. Select the appropriate **Azure Region** and **Availability Zone**.
15. Move to the next tab, **Outbound IP**, and select the Public IP Address and Prefix Name that you created previously.
16. Next, on the **Subnet** tab, you will configure which subnets of a VNet should use this NAT gateway.
17. The **Tags** tab is optional, and you should use it only when you need to logically organize your resources in a particular taxonomy to easily identify them later.
18. You can review a summary of the selections in the **Review + Create** tab. Once you finish reviewing it, click the **Create** button.

You can also use the `New-AzNatGateway` cmdlet to create a NAT Gateway using PowerShell, as shown:

[Click here to view code image](#)

```
New-AzNatGateway -ResourceGroupName "AZ500RG" -
  Name "nat_gt" -IdleTimeoutInMinutes 4
  -Sku "Standard" -Location "eastus2" -
  PublicIpAddress PublicIpAddressName
```

Secure the connectivity of virtual networks

With organizations migrating to the cloud, virtual private networks (VPNs) are constantly used to establish a secure communication link between on-premises and cloud network infrastructure. While this is one common scenario, there are many other scenarios where a VPN can be used. You can use Azure VPN to connect two different Azure regions or different subscriptions.

Azure natively offers a service called VPN gateway, which is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and on-premises resources. You can also use a VPN gateway to send encrypted traffic between Azure virtual networks. When planning your VPN Gateway implementation, be aware that each virtual network can have only one VPN gateway, and you can create multiple connections to the same VPN gateway. Depending on the scenario, you can select from different types of VPN connectivity. The available options are

- **Site-to-Site (S2S) VPN** This type of VPN is used in scenarios where you need to connect on-premises resources to Azure. The encrypted connection tunnel uses IPsec/IKE (IKEv1 or IKEv2).
- **Point-to-Site (P2S) VPN** This type of VPN is used in scenarios where you need to connect to your Azure VNet from a remote location. For example, you would use P2S when you are working remotely (hotel, home, conference, and the like) and you need to access resources in your VNet. This VPN uses SSTP (Secure Socket Tunneling Protocol) or IKE v2 and does not require a VPN device.
- **VNet-to-VNet** As the name states, this VPN is used in scenarios where you need to encrypt connectivity between VNets. This type of connection uses IPsec (IKE v1 and IKE v2).
- **Multi-Site VPN** This type of VPN is used in scenarios where you need to expand your site-to-site configuration to allow multiple on-premises sites to access a virtual network.

ExpressRoute is another option that allows connectivity from your on-premises resources to Azure. This option

uses a private connection to Azure from your WAN, instead of a VPN connection over the Internet.

VPN authentication

The Azure VPN connection is authenticated when the tunnel is created. Azure generates a pre-shared key (PSK), which is used for authentication. This pre-shared key is an ASCII string character no longer than 128 characters. This authentication happens for policy-based (static routing) or routing-based VPN (dynamic routing). You can view and update the pre-shared key for a connection with these PowerShell cmdlets:

- **Get-AzVirtualNetworkGatewayConnectionSharedKey**
This command is used to show the pre-shared key.
- **Set-AzVirtualNetworkGatewayConnectionSharedKey**
This command is used to change the pre-shared key to another value.

For point-to-site (P2S) VPN scenarios, you can use native Azure certificate authentication or Azure AD authentication. For native Azure certificate authentication, a client certificate is presented on the device, which is used to authenticate the users who are connecting. The certificate can be one that was issued by an enterprise certificate authority (CA), or it can be a self-signed root certificate. For native Azure AD, you can use the native Azure AD credentials. Keep in mind that native Azure AD is only supported for the OpenVPN protocol and Windows 10. (Windows 10 requires the use of the Azure VPN Client.)

If your scenario requires the enforcement of a second factor of authentication before access to the resource is granted, you can use Azure Multi-Factor Authentication (MFA) with conditional access. Even if you don't want to implement MFA across your entire company, you can scope the MFA to be employed only for VPN users using conditional access capability.

More Info Configuring MFA for VPN Access

You can see the steps for configuring MFA for VPN access at
<http://aka.ms/az500mfa>.

Another option available for P2S is the authentication using RADIUS (which also supports IKEv2 and SSTP VPN). Keep in mind that RADIUS is only supported for VpnGw1, VpnGw2, and VpnGw3 SKUs. For more information about the latest VPN SKUs, visit <http://aka.ms/az500vpnsku>. Figure 2-12 shows an example of the options that appear when you are configuring a P2S VPN and you need to select the authentication type.

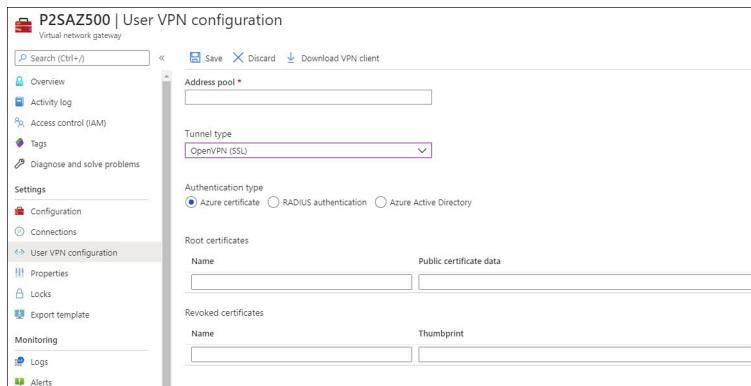


Figure 2-12 Authentication options for VPN

The options that appear right under the **Authentication Type** section will vary according to the Authentication Type you select. In Figure 2-12, **Azure Certificate** is chosen, and the page shows options to enter the **Name** and **Public Certification Data** for the **Root Certificates** and the **Name** and **Thumbprint** for the **Revoked Certificates**. If you select **RADIUS authentication**, you will need to specify the **Server IP Address** and the **Server Secret**. Lastly, if you select the **Azure Active Directory** option, you will need to specify the **Tenant's URL**; the **Audience** (which identifies the recipient resource the token is intended for); and the **Issuer** (which identifies the Security Token

Service (STS) that issued the token). Lastly, choose the Azure AD tenant.

Your particular scenario will dictate which option to use. For example, Contoso's IT department needs to implement a VPN solution that can integrate with a certificate authentication infrastructure that it already has through RADIUS. In this case, you should use RADIUS certificate authentication. When using the RADIUS certificate authentication, the authentication request is forwarded to a RADIUS server, which handles the certificate validation. If the scenario requires that the certificate authentication be performed by the Azure VPN gateway, the right option would be to use the Azure native certificate authentication.

ExpressRoute encryption

If your connectivity scenario requires a higher level of reliability, faster speeds, consistent latencies, and higher security than typical connections over the Internet, you should use ExpressRoute, which provides layer 3 connectivity between your on-premises network and the Microsoft Cloud.

ExpressRoute supports two different encryption technologies to ensure the confidentiality and integrity of the data that is traversing from on-premises to Microsoft's network. The options are

- Point-to-point encryption by MACsec
- End-to-end encryption by IPSec

MACsec encrypts the data at the media access control (MAC) level or at network layer 2. When you enable MACsec, all network control traffic is encrypted, which includes the border gateway protocol (BGP) data traffic, and your (customer) data traffic. This means that you can't encrypt only some of your ExpressRoute circuits.

If you need to encrypt the physical links between your network devices and Microsoft's network devices when you connect to Microsoft via ExpressRoute Direct, MACsec is preferred. MACsec also allows you to bring your own MACsec key for encryption and store it in Azure Key Vault. If this is the design choice, remember that you will need to decide when to rotate the key.

Tip Expressroute Direct

Although MACsec is only available on ExpressRoute Direct, it comes disabled by default on ExpressRoute Direct ports.

Keep in mind that when you update the MACsec key, the on-premises resources will temporarily lose connectivity to Microsoft over ExpressRoute. This happens because MACsec configuration only supports pre-shared key mode, so you must update the key on both sides. In other words, if there is a mismatch, traffic flow won't occur. Plan the correct maintenance window to reduce the impact on production environments.

The other option is to use end-to-end encryption with IPsec, which encrypts data at the Internet protocol (IP)-level or at network layer 3. A very common scenario is to use IPsec to encrypt the end-to-end connection between on-premises resources and your Azure VNet. In a scenario where you need to encrypt layer 2 and layer 3, you can enable MACsec and IPsec.

More Info Create Ipsec Over Expressroute

You can learn how to create IPsec over ExpressRoute for Virtual WAN at
<http://aka.ms/az500vpnexpressroute>.

Point-to-site

To implement a point-to-site (P2S) VPN in Azure, you first need to decide what authentication method you will use based on the options that were presented earlier in this section. The authentication method will dictate how

the P2S VPN will be configured. When configuring the P2S VPN, you will see the options available under **Tunnel Type**, as shown in Figure 2-13.

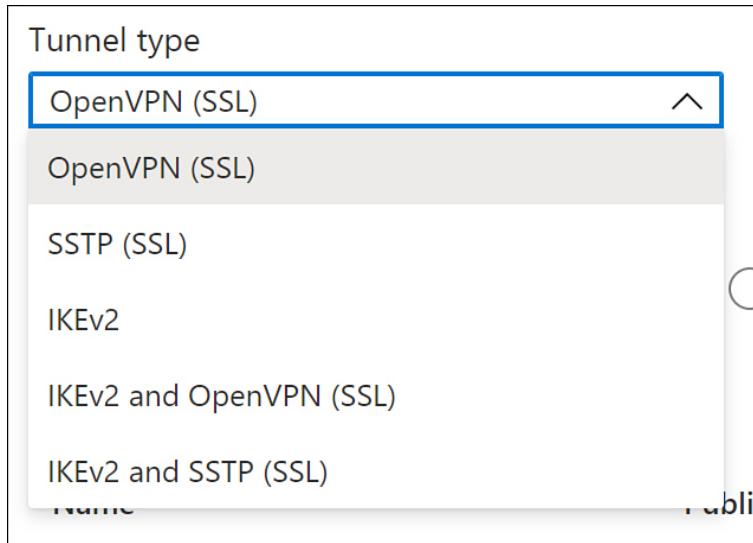


Figure 2-13 Different options for the VPN tunnel

- Another important variable to select is the protocol that will be used. Use Table 2-1 to select the most-appropriate protocol based on the advantages and limitations:

Table 2-1 Advantages and limitations

Protocol	Advantages	Limitations
OpenVPN Protocol	<p>This is a TLS VPN-based solution that can traverse most firewalls on the market.</p> <p>Can be used to connect from a variety of operating systems, including Android, iOS (versions 11.0 and above), Windows, Linux, and Mac devices (OSX versions 10.13 and above).</p>	<p>Basic SKU is not supported.</p> <p>Not available for the classic deployment model.</p>
Secure Socket Tunnel	Can traverse most firewalls because it uses TCP port 443.	<p>Only supported on Windows devices.</p> <p>Supports up to 128 concurrent connections,</p>

ng Prot ocol (SS TP)		regardless of the gateway SKU.
IKE v2	<p>Standard-based IPsec VPN solution.</p> <p>Can be used to connect to Mac devices (OSX versions 10.11 and above).</p>	<p>Basic SKU is not supported.</p> <p>Not available for the classic deployment model.</p> <p>Uses nonstandard UDP ports, so you need to ensure that these ports are not blocked on the user's firewall. The ports in use are UDP 500 and 4500.</p>



Exam Tip

For the AZ-500 exam, make sure to carefully read the scenarios because there will be indications of what the company wants to accomplish, and those indications will be used to decide which protocol to implement or which protocol is not an option for the specified scenario.

Site-to-site

A site-to-site (S2S) VPN is used in most scenarios to allow the communication from one location (on-premises) to another (Azure) over the Internet. To configure a S2S, you need the following prerequisites fulfilled before you start:

- A VPN device on-premises that is compatible with Azure VPN policy-based configuration or route-based configuration. See the full list at <https://aka.ms/az500s2sdevices>.
- Externally-facing public IPv4 address.
- IP address range from your on-premises network that will be utilized to allow Azure to route to your on-premises location.

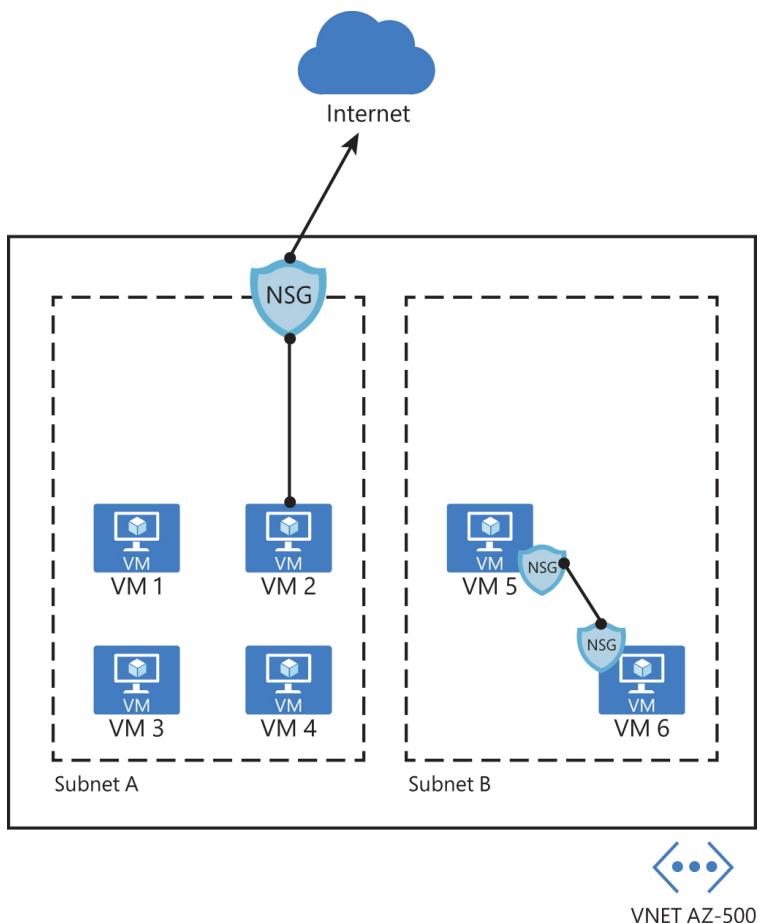
More Info More Info Creating an S2S VPN

Once you have those requirements, you can create your S2S VPN. For more information on the steps, see <https://aka.ms/az500s2svpn>. If your VPN connection is over IPsec (IKE v1 and IKE v2), you need to have a VPN device or an RRAS.

Configure network security groups and Application Security Groups

Network security groups (NSG) in Azure allow you to filter network traffic by creating rules that allow or deny inbound network traffic to or outbound network traffic from different types of resources. For example, you could configure an NSG to block inbound traffic from the Internet to a specific subnet that only allows traffic from a network virtual appliance (NVA).

Network security groups can be enabled on the subnet or to the network interface in the VM, as shown in [Figure 2-14](#).



VNET AZ-500

Figure 2-14 Different NSG implementations

In the diagram shown in [Figure 2-14](#), you have two different uses of NSG. In the first case, the NSG is assigned to the subnet A. This can be a good way to secure the entire subnet with a single set of NSG rules. However, there will be scenarios where you might need to control the NSG on the network interface level, which is the case of the second scenario (subnet B), where VM 5 and VM 6 have a NSG assigned to the network interface.

When inbound traffic is coming through the VNet, Azure processes the NSG rules that are associated with the subnet first—if there are any—and then it processes the NSG rules that are associated with the network interface. When the traffic is leaving the VNet (outbound traffic), Azure processes the NSG rules that are associated with

the network interface first, followed by the NSG rules that are associated to the subnet.

When you create an NSG, you need to configure a set of rules to harden the traffic. These rules use the following parameters:

- **Name** The name of the rule.
- **Priority** The order in which the rule will be processed. Lower numbers have high priority, which means that a rule priority 100 will be evaluated before rule priority 300. Once the traffic matches the rule, it will stop moving forward to evaluate other rules. When configuring the priority, you can assign a number between 100 and 4096.
- **Source** Define the source IP, CIDR Block, Service Tag, or Application Security Group.
- **Destination** Define the destination IP, CIDR Block, Service Tag, or Application Security Group.
- **Protocol** Define the TCP/IP protocol that will be used, which can be set to **TCP**, **UDP**, **ICMP**, or **Any**.
- **Port Range** Define the port range or a single port.
- **Action** This determines the action that will be taken once this rule is processed. This can be set to **Allow** or **Deny**.

Before creating a new NSG and adding new rules, it is important to know that Azure automatically creates default rules on NSG deployments. Following is a list of the inbound rules that are created:

- **AllowVNetInBound**
 - **Priority** 6500
 - **Source** VirtualNetwork
 - **Source Ports** 0-65535
 - **Destination** VirtualNetwork
 - **Destination Ports** 0-65535
 - **Protocol** Any
 - **Access** Allow
- **AllowAzureLoadBalancerInBound**
 - **Priority** 6501
 - **Source** AzureLoadBalancer
 - **Source Ports** 0-65535
 - **Destination** 0.0.0.0/0

- **Destination Ports** 0-65535
 - **Protocol** Any
 - **Access** Allow
- **DenyAllInbound**
 - **Priority** 6501
 - **Source** AzureLoadBalancer
 - **Source Ports** 0-65535
 - **Destination** 0.0.0.0/0
 - **Destination Ports** 0-65535
 - **Protocol** Any
 - **Access** Deny

Below is a list of outbound rules that are created:

- **AllowVnetOutBound**
 - **Priority** 6501
 - **Source** VirtualNetwork
 - **Source Ports** 0-65535
 - **Destination** VirtualNetwork
 - **Destination Ports** 0-65535
 - **Protocol** Any
 - **Access** Allow
- **AllowInternetOutBound**
 - **Priority** 6501
 - **Source** 0.0.0.0/0
 - **Source Ports** 0-65535
 - **Destination** Internet
 - **Destination Ports** 0-65535
 - **Protocol** Any
 - **Access** Allow
- **DenyAllOutBound**
 - **Priority** 6501
 - **Source** 0.0.0.0/0
 - **Source Ports** 0-65535
 - **Destination** 0.0.0.0/0
 - **Destination Ports** 0-65535
 - **Protocol** Any
 - **Access** Deny

Important Default Rules Cannot be Removed

Keep in mind that these default rules cannot be removed, though if necessary, you can override them by creating rules with higher priorities.

Follow the steps below to create and configure an NSG, which in this example, will be associated with a subnet:

1. Navigate to the Azure portal by opening <https://portal.azure.com>.
2. In the search bar, type **network security**, and under **Services**, click **Network Security Groups**; the **Network Security Groups** page appears.
3. Click the **Add** button; the **Create Network Security Group** page appears, as shown in Figure 2-15.

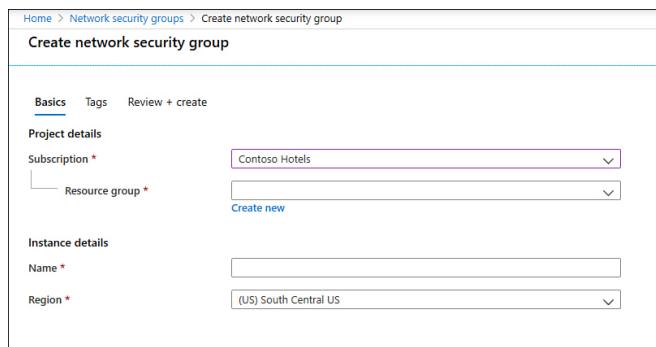


Figure 2-15 Initial parameters of the network security group

4. In the **Subscription** field, select the subscription where this NSG will reside.
5. In the **Resource Group** field, select the resource group in which this NSG will reside.
6. In the **Name** field, type the name for this NSG.
7. In the **Region** field, select the Azure region in which this NSG will reside.
8. Click **Review + Create** button, review the options, and click the **Create** button.
9. Once the deployment is complete, click the **Go To Resource** button. The NSG page appears.

At this point, you have successfully created your NSG, and you can see that the default rules are already part of it. The next step is to create the custom rules, which can be inbound or outbound. (This example uses inbound rules.) The same operation could be done using the New-

AzNetworkSecurityGroup PowerShell cmdlet, as shown in the following example:

[Click here to view code image](#)

```
New-AzNetworkSecurityGroup -Name "AZ500NSG" -  
ResourceGroupName "AZ500RG" -Location  
"westus"
```

Follow these steps to create an inbound rule that allows FTP traffic from any source to a specific server using Azure portal:

1. On the NSG page, under **Settings** in the left navigation pane, click **Inbound Security Rules**.
2. Click the **Add** button; the **Add Inbound Security Rule** blade appears, as shown in Figure 2-16.

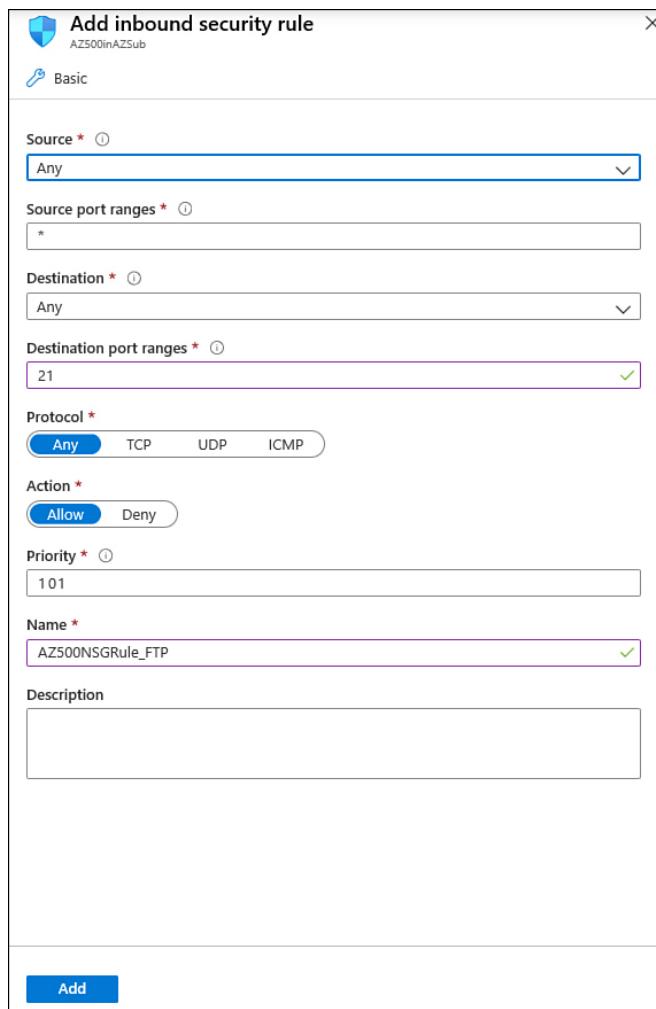


Figure 2-16 Creating an inbound security rule for your NSG

3. On this blade, you start by specifying the source, which can be an IP address, a service tag, or an ASG. If you leave the default option (**Any**), you are allowing any source. For this example, leave this set to **Any**.
4. In the **Source Port Ranges** field, you can harden the source port. You can specify a single port or an interval. For example, you can allow traffic from ports 50 to 100. Also, you can use comma to add another condition to the range, such as 50-100, 135, which specifies ports 50 through 100 and 135. Leave the default selection (*), which allows any source port.
5. In the **Destination** field, the options are nearly the same as the **Source** field. The only difference is that you can select the VNet as the destination. For this example, change this option to **IP Addresses** and enter the internal IP address of the VM that you created at the beginning of this chapter.
6. In the **Destination Port Ranges** field, specify the destination port that will be allowed. The default port is 8080; for this example, change it to 21.
7. In the **Protocol** field, you can select which protocol you are going to allow; in this case, change it to **TCP**.
8. Leave the **Action** field set to **Allow**, which is the default selection.
9. You can also change the **Priority** of this rule. Remember that the lowest priority is evaluated first. For this example, change it to **101**.
10. In the **Name** field, change it to **AZ500NSGRule_FTP** and click the **Add** button.

The NSG will be created, and a new rule will be added to the inbound rules. At this point, your inbound rules should look like the rules shown in Figure 2-17.

Inbound security rules						
Priority	Name	Port	Protocol	Source	Destination	Action
101	AZ500NSGRule_FTP	21	TCP	Any	10.3.0.50	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

Figure 2-17 List of inbound rules

While these are the steps to create the inbound rule, this NSG has no use if it is not associated with a subnet or a virtual network interface. For this example, you will associate this NSG to a subnet. The intent is to block all traffic to this subnet and only allow FTP traffic to this

specific server. Use the following steps to create this association:

1. In the left navigation pane of the **NSG Inbound Security Rules** page under **Settings**, click **Subnets**.
2. Click the **Associate** button, and in the **Virtual Network** drop-down menu, select the VNet where the subnet resides.
3. After this selection, you will see that the **Subnet** drop-down menu appears; select the subnet and click the **OK** button.

You could also use PowerShell to create an NSG and then associate the NSG to a subnet. To create an NSG using PowerShell, use the New-AzNetworkSecurityRuleConfig cmdlet, as shown in the following example:

[Click here to view code image](#)

```
$MyRule1 = New-AzNetworkSecurityRuleConfig -Name
ftp-rule -Description "Allow FTP"
-Access Allow -Protocol Tcp -Direction Inbound -
Priority 100 -SourceAddressPrefix *
-SourcePortRange * -DestinationAddressPrefix * -
DestinationPortRange 21
```

Application security group

If you need to define granular network security policies based on workloads that are centralized on application patterns instead of explicit IP addresses, you need to use the application security group (ASG). An ASG allows you to group VMs and secure applications by filtering traffic from trusted segments of your network, which adds an extra level of micro-segmentation.

You can deploy multiple applications within the same subnet and isolate traffic based on ASGs. Another advantage is that you can reduce the number of NSGs in your subscription. For example, in some scenarios, you can use a single NSG for multiple subnets of your virtual network and perform the micro-segmentation on the application level by using ASG. [Figure 2-18](#) shows an

example of how ASG can be used in conjunction with NSG.

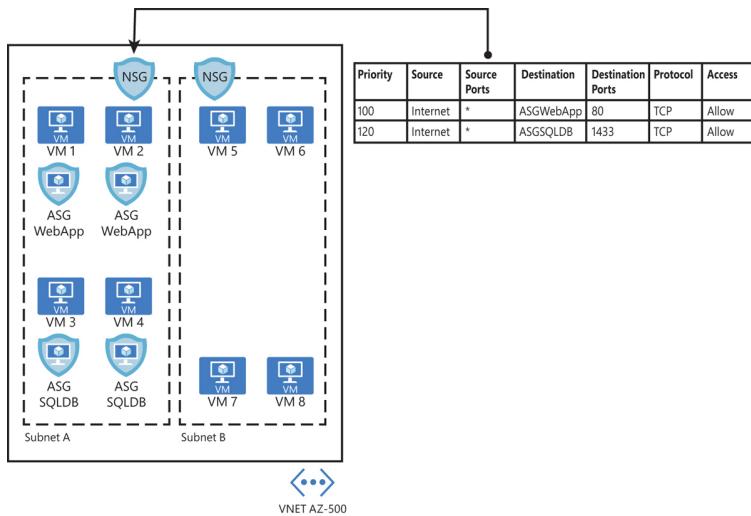


Figure 2-18 ASG used as the destination in the NSG routing table

In the example shown in Figure 2-18, two ASGs have been created to define the application pattern for a web application and another ASG to define the application pattern for a SQL database. Two VMs are part of each group, and the ASG is used in the routing table of the NSG located in subnet A. In the NSG routing table, you can specify one ASG as the source and destination, but you cannot specify multiple ASGs in the source or destination.

When you deploy VMs, you can make them members of the appropriate ASGs. In case your VM has multiple workloads (Web App and SQL, for example), you can assign multiple ASGs to each application. This will allow you to have different types of access to the same VM according to the workload. This approach also helps to implement a zero-trust model by limiting access to the application flows that are explicitly permitted. Follow these steps to create an ASG:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **application security** and under **Services**, click **Application Security Groups**.

3. In the **Application Security Groups** dashboard, click the **Add** button, which makes the **Create An Application Security Group** page appear, as shown in Figure 2-19.

The screenshot shows the 'Create an application security group' wizard. The 'Basics' tab is selected. Under 'Project details', the 'Subscription' dropdown is set to 'Visual Studio Ultimate with MSDN'. The 'Resource group' dropdown has 'Create new' selected. Under 'Instance details', the 'Name' field is empty and the 'Region' dropdown is set to '(South America) Brazil South'. At the bottom, there are buttons for 'Review + create', '< Previous', 'Next : Tags >', and 'Download a template for automation'.

Figure 2-19 Create An Application Security Group

4. In the **Subscription** drop-down menu, select the appropriate subscription for this ASG.
5. In the **Resource Group** drop-down menu, select the resource group in which this ASG will reside.
6. In the **Name** field, type a name for this ASG.
7. In the **Region** drop-down menu, select the appropriate region for this ASG and click the **Review + Create** button.
8. On the **Review + Create** button page, click the **Create** button.

Now that the ASG is created, you need to associate this ASG to the network interface of the VM that has the workload you want to control. Follow these steps to perform this association:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **virtual** and under **Services**, click **Virtual Machines**.
3. Click in the VM that you want to perform this association.
4. On the VM's page, in the **Settings** section, click the **Networking** option.
5. Click the **Application Security Group** tab, and the page shown in Figure 2-20 appears.

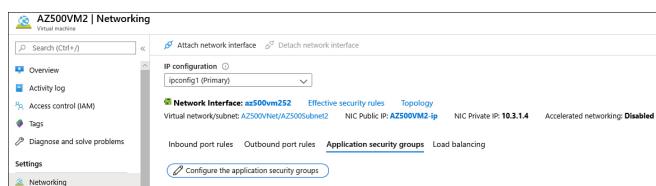


Figure 2-20 Associating the ASG to the virtual network interface card

6. Click the **Configure The Application Security Groups** button and the **Configure The Application Security Groups** blade appears, as shown in Figure 2-21.

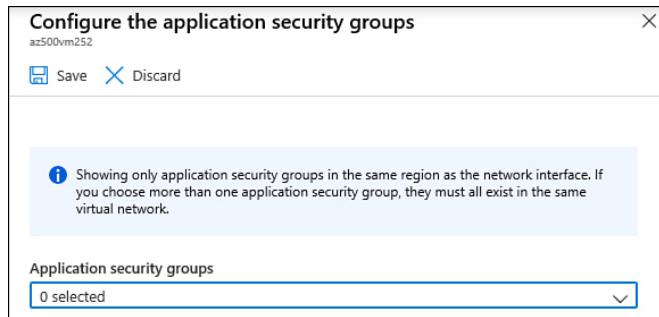


Figure 2-21 Selecting the ASG

7. Select the appropriate ASG and click the **Save** button.

You can also use the `New-AzApplicationSecurityGroup` cmdlet to create a new ASG, as shown in the following example:

[Click here to view code image](#)

```
New-AzApplicationSecurityGroup -ResourceGroupName  
"MyRG" -Name "MyASG" -Location "West  
US"
```

Now when you create your new NSG rule for inbound or outbound traffic, you can select the ASG as source or destination.

Create and configure Azure Firewall

While NSG provides stateful package flow and custom security rules, you will need a more robust solution when you need to protect an entire virtual network. If your company needs a fully stateful, centralized network firewall as a service (FWaaS) that provides network and application-level protection across different subscriptions and virtual networks, you should choose Azure Firewall.

Also, Azure Firewall can be used in scenarios where you need to span multiple availability zones for increased availability. Although there's no additional cost for an Azure Firewall deployed in an availability zone, there are additional costs for inbound and outbound data transfers associated with Availability Zones. [Figure 2-22](#) shows an Azure Firewall in its own VNet and subnet, allowing some traffic and blocking other traffic based on a series of evaluations.

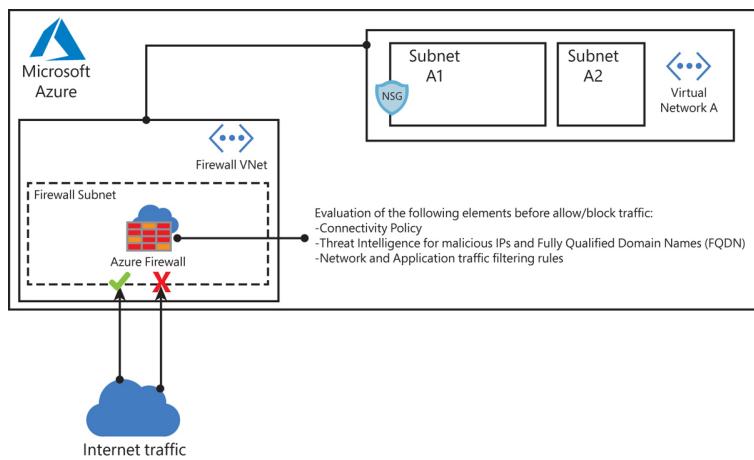


Figure 2-22 Azure Firewall topology

As shown in [Figure 2-22](#), the Azure Firewall will perform a series of evaluations prior to allowing or blocking the traffic. Just as with an NSG, the rules in Azure Firewall are processed according to the rule type in priority order (lower numbers to higher numbers). A rule collection name may contain only letters, numbers, underscores, periods, or hyphens. You can configure NAT rules, network rules, and applications rules on Azure Firewall. Keep in mind that Azure Firewall uses a static public IP address for your virtual network resources, and you need that prior to deploying your firewall. Azure Firewall also supports learning routes via Border Gateway Protocol (BGP).

To evaluate outbound traffic, Azure Firewall will query the network and application rules. Just as with an NSG, when a match is found in a network rule, no other rules

are processed. If there is no match, Azure Firewall will use the infrastructure rule collection. This collection is created automatically by Azure Firewall and includes platform-specific fully qualified domain names (FQDN). If there is still no match, Azure Firewall denies outgoing traffic.

For incoming traffic evaluation, Azure Firewall uses rules based on Destination Network Address Translation (DNAT). These rules are also evaluated in priority and before network rules. If a match is found, an implicit corresponding network rule to allow the translated traffic is added. Although this is the default behavior, you can override this by explicitly adding a network rule collection with deny rules that match the translated traffic (if needed).

Important Web Application Firewall (WAF)

Application rules aren't applied for inbound connections. Microsoft recommends using Web Application Firewall (WAF) if you want to filter inbound HTTP/S traffic.

In [Figure 2-22](#), you also saw that Azure Firewall leverages Microsoft Threat Intelligence during the traffic evaluation. The Microsoft Threat Intelligence is powered by Intelligent Security Graph and is used by many other services in Azure, including Azure Security Center.

Now that you know the key components of the Azure Firewall, use the following steps to deploy and configure it:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the main dashboard, click **Create A Resource**.
3. Type **firewall** and click **Firewall** in the drop-down menu.
4. On the **Firewall** page, click the **Create** button, and the **Create A Firewall** blade appears, as shown in [Figure 2-23](#).

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 * AI Infra Build

Resource group * Create new

Instance details

Name *

Region * (US) South Central US

Availability zone None

Choose a virtual network Create new Use existing

Virtual network name *

Address space * 10.0.0.0/16 (0 addresses)

Subnet AzureFirewallSubnet

Subnet address space * 10.0.0.0/24 (0 addresses)

Firewall public IP address * (New) MyPubIP Add new

Forced tunneling (preview) Disabled

Review + create Previous Next : Tags > Download a template for automation

Figure 2-23 Creating a new Azure Firewall

5. If you have multiple subscriptions, make sure to click the **Subscription** drop-down menu and select the one that you want to use to deploy Azure Firewall.
6. In the **Resource Group** drop-down menu, select the resource group in which you want to deploy your Azure Firewall.
7. In the **Instance Details** section, in the **Name** field, type the name for this Azure Firewall instance. There is a 50-character limit for the name.
8. In the **Region** drop-down menu, select the region where the Azure Firewall will reside.
9. In the **Availability Zone** drop-down menu, select the availability zone in which the firewall will reside.
10. For the **Choose Virtual Network** option, select **Use Existing** and select an existing VNet.
11. In the **Virtual Network** drop-down menu, select the VNet to which you want to deploy Azure Firewall.
12. In the **Firewall Public IP Address** field, select an existing unused public IP address or click **Add New** to create a new one in case all your public IPs are already allocated.
13. You can either enable or disable **Force Tunneling**. The default option is **Disabled**. By enabling this option, you are instructing Azure Firewall to route all Internet-bound traffic to a designated next hop instead of going directly to the Internet. Keep in mind

that if you configure Azure Firewall to support forced tunneling, you can't undo this configuration. Leave the default selection and click the **Review + Create** button.

14. The creation of the Azure Firewall will take several minutes. After the deployment is complete, you can click the **Go To Resource** button.

You can also deploy a new Azure Firewall using the `New-AzFirewall` cmdlet, as shown in the following example:

[Click here to view code image](#)

```
New-AzFirewall -Name "azFw" -ResourceGroupName  
MyRG -Location centralus -VirtualNetwork  
MyVNet -PublicIpAddress MyPubIP
```

Creating an application rule

Now that the Azure Firewall is created you can start creating rules. To start you are going to create an application rule to allow outbound access to www.bing.com. Follow these steps to create a rule:

1. On the page that you have open for the firewall you created, click **Rules**, as shown in Figure 2-24.

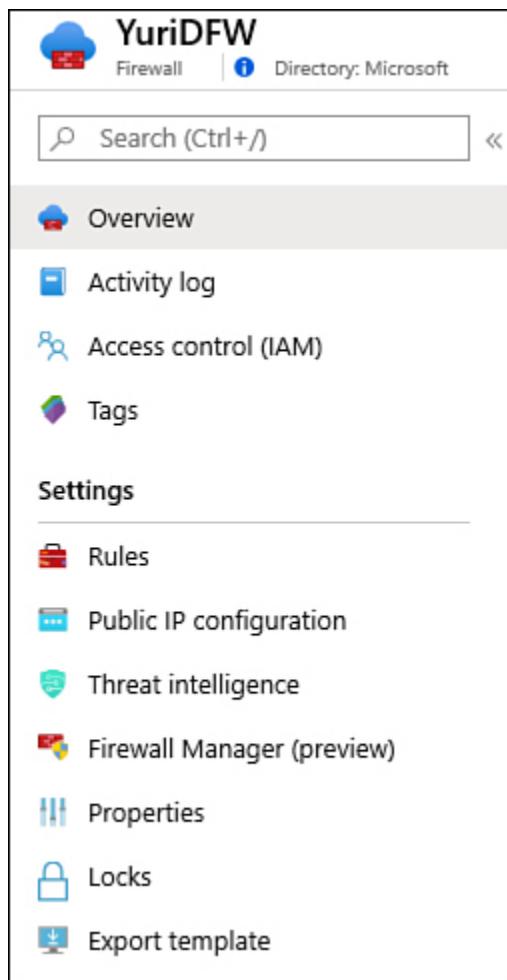


Figure 2-24 Firewall options

2. Click the **Application Rule Collection** tab and then click the **+ Add Application Rule Collection** option. The **Add Application Rule Collection** page appears, as shown in Figure 2-25.

Figure 2-25 Creating a new application rule collection

3. In the **Name** field, type a name for the rule; for this example, type **Bing**.

4. In the **Priority** field, type the priority for this rule; for this example, type **100**.
5. In the **Action** drop-down menu, leave the default option (**Allow**).
6. No changes are necessary in the **FQDN Tags** field.
7. In the **Target FQDNs** field, type **AllowBing** and leave the **Source Type** set to **IP Address**.
8. Type * in the **Source** field.
9. In the **Protocol:Port** field, type **http,https**.
10. In the **Target FQDNs** field, type **www.bing.com**.
11. Click the **Add** button.

In case you want to perform the same configuration using PowerShell, you can use the New-AzFirewallApplicationRule cmdlet, as shown here:

[Click here to view code image](#)

```
$MyAppRule = New-AzFirewallApplicationRule -Name AllowBing -SourceAddress * -Protocol http, https -TargetFqdn www.bing.com $AppCollectionRule = New-AzFirewallApplicationRuleCollection -Name App-Coll01 -Priority 100 -ActionType Allow -Rule $MyAppRule $Azfw.ApplicationRuleCollections = $AppRuleCollection Set-AzFirewall -AzureFirewall $Azfw
```

Tip Azure WEB Application Firewall (WAF)

If your organization needs inbound HTTP/S protection, it is recommended that you use a web application firewall such as Azure Web Application Firewall (WAF) instead of creating an application rule for port 443.

Creating a network rule

Creating a network rule is very similar to creating an application rule. For this example, you are going to create an outbound network rule that allows access to an external DNS Server. Follow these steps to create your network rule:

1. On the **Firewalls** rules page, click the **Network Rule Collection** tab.
2. Click the **Add Network Rule Collection** option; the **Add Network Rule Collection** blade appears, as shown in [Figure 2-26](#).

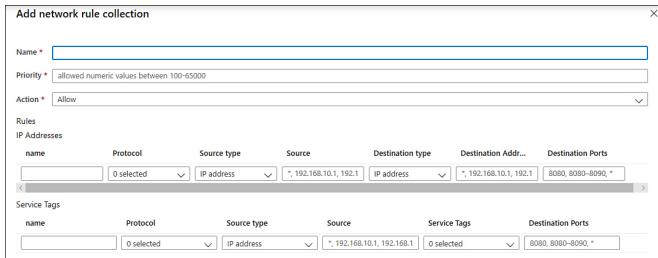


Figure 2-26 Creating a new network rule collection

3. In the **Name** field, type **DNS**.
4. In the **Priority** field, type **200**.
5. In the **Action** field, leave the default selection (**Allow**).
6. Under the **IP Addresses** section, type **DNSOutbound** in the **Name** field.
7. Select **UDP** in the **Protocol** field.
8. Leave **IP Address** selection in the **Source Type** field.
9. In the **Source** field, type the range of your subnet such as **10.30.0.0/24**.
10. Leave the **IP Address** selection in the **Destination Type** field.
11. In the **Destination Address** field, type the IP address of the external DNS.
12. In the **Destination Port**, type **53**.
13. Click the **Add** button.

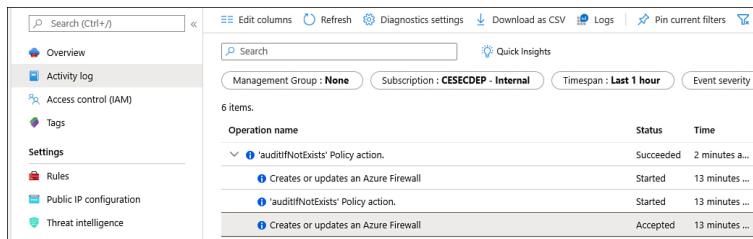
In case you want to perform the same configuration using PowerShell, you can use the `New-AzFirewallNetworkRule` cmdlet, as shown here:

[Click here to view code image](#)

```
New-AzFirewallNetworkRule -Name "DNSOutbound" - 
Protocol UDP -SourceAddress 
"10.30.0.0/24" -DestinationAddress 
IP_of_the_DNSSEver -DestinationPort 53
```

Firewall logs

When system admins need to audit configuration changes in the Azure Firewall, they should use Azure Activity logs. For example, the creation of those two rules (application and network) will appear in the Activity Log, which will look similar to [Figure 2-27](#).



The screenshot shows the Azure Activity Log interface. On the left, there's a navigation pane with 'Overview', 'Activity log' (which is selected), 'Access control (IAM)', 'Tags', 'Settings', 'Rules', 'Public IP configuration', and 'Threat intelligence'. The main area has a search bar, 'Quick Insights' button, and filters for 'Management Group: None', 'Subscription: CESECDEP - Internal', 'Timespan: Last 1 hour', and 'Event severity'. Below these, it says '6 items.' and lists the following:

Operation name	Status	Time
Creates or updates an Azure Firewall	Succeeded	2 minutes ago
Creates or updates an Azure Firewall	Started	13 minutes ago
'auditIfNotExists' Policy action.	Started	13 minutes ago
'auditIfNotExists' Policy action.	Accepted	13 minutes ago
Creates or updates an Azure Firewall	Accepted	13 minutes ago

Figure 2-27 Activity logs showing the changes in the Azure Firewall

While these actions are automatically logged in the Azure Activity Log, the diagnostic logging for application and network rules are not enabled by default. You can also enable Firewall metrics. These metrics are collected every minute and can be useful for alerting because they can be sampled frequently. When you enable metrics collection, the following metrics will be available for Azure Firewall:

- Application rules hit count
- Network rules hit count
- Data processed
- Firewall health state
- SNAT port utilization

These metrics and the diagnostic logging for application and network rule can be enabled in the Azure Firewall dashboard. Use the following steps to enable these logs:

1. On the **Firewalls** page, in the left navigation pane, under the **Monitoring** section, click **Diagnostic Settings**. The **Diagnostic Settings** page appears, as shown in [Figure 2-28](#).

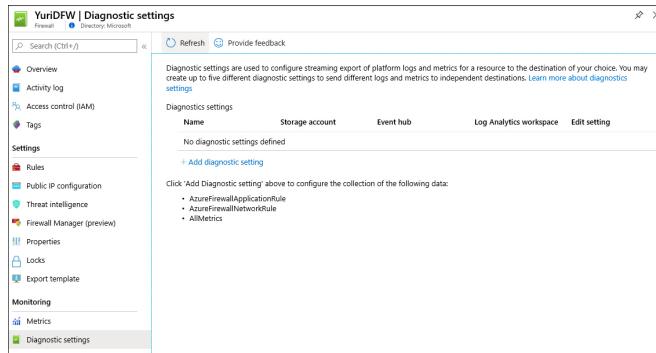


Figure 2-28 Diagnostic settings page

2. Click the **Add Diagnostic Setting** option, which makes the **Diagnostic Settings** blade appear, as shown in Figure 2-29.

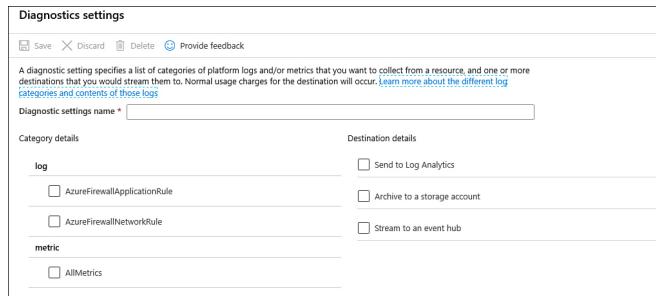


Figure 2-29 Diagnostic Settings page

3. In the **Diagnostic Settings Name** field, type a name for this setting.
4. In the **Log** section, enable **AzureFirewallApplicationRule** and **AzureFirewallNetworkRule**.
5. In the **Metric** section, enable **AllMetrics**.
6. In the **Destination Details** section, you can choose where you want to send the logs: Log Analytics, Storage account, or Event hub. If you need to retain logs for a longer duration for review as needed, **Storage Account** is the best option. If you need to send the logs to a security information and event management (SIEM) tool, the Event Hub is the best option. If you need more real-time monitoring, **Log Analytics** is a better fit. Notice that you can select multiple options, which allows you to address multiple needs.
7. For this example, select **Send To Log Analytics**, and select the workspace in which the logs will reside.
8. Click **Save** and once it is saved, close the blade.
9. Notice that the name of your logging configuration now appears on the **Diagnostic Settings** page.
10. You can use the `Set-AzDiagnosticSetting` cmdlet to enable diagnostic logging, as shown in the following example:

[Click here to view code image](#)

```
Set-AzDiagnosticSetting -ResourceId  
/subscriptions/<subscriptionId>/  
resourceGroups/<resource group  
name>/providers/Microsoft.Network/  
azureFirewalls/<Firewall name>  
-StorageAccountId  
/subscriptions/<subscriptionId>/resourceGroups/<resource  
group  
name>/providers/Microsoft.Storage/storageAccounts/<storage  
account name>  
-Enabled $true
```

11. Now that the diagnostic logging is configured, click **Logs** in the left navigation pane in the **Monitoring** section. The **Log Analytics** workspace appears with the Azure Firewall schema, as shown in Figure 2-30.

The screenshot shows the Azure Log Analytics workspace interface. At the top, there's a header with a 'New Query 1*' button and a '+' icon. Below the header, the workspace is titled 'YuriDFW' and has a 'Select Scope' dropdown. A navigation bar at the top includes 'Tables', 'Queries', 'Filter', and a back arrow. A search bar is located below the navigation bar. The main area displays the 'Group by: Resource Type' and 'Filters: not selected' settings. A 'Favorites' section is present, with a note: 'You can add favorites by clicking on the star icon'. A collapsed section titled 'Firewall' is expanded, showing three items: 'AzureActivity', 'AzureDiagnostics', and 'AzureMetrics', each preceded by a right-pointing arrow and a grid icon.

Figure 2-30 Schema for the Azure Firewall in Log Analytics

12. To query on Log Analytics workspace, you use Kusto Query Language (KQL). You can use the sample query to retrieve the logs that are related to the network rules:

[Click here to view code image](#)

```
AzureDiagnostics  
| where Category ==
```

```
"AzureFirewallNetworkRule"
```

Configure Azure Front Door service as an application gateway

Consider an Azure deployment across different regions that needs to provide a high-performance experience for applications, and it is resilient to failures. For this type of scenario, Azure Front Door is the best solution.

Azure Front Door works at layer 7 (HTTP/HTTPS) and uses the anycast protocol with split TCP, plus Microsoft's global network for improving global connectivity. By using split TCP-based anycast protocol, Front Door ensures that your users promptly connect to the nearest Front Door POP (point of presence).

You can configure Front Door to route your client requests to the fastest and most available application back end, which is any Internet-facing service hosted inside or outside of Azure. Some other capabilities included in Front Door are listed here:

- **Smart health probe** Front Door monitors your back ends for availability and latency. According to its results, it will instant failover when a back end goes down.
- **URL-based routing** Allows you to route traffic to the back end based on the URL's path of the request. For example, traffic to `www.fabrikam.com/hr/*` is routed to a specific pool, whereas `www.fabrikam.com/soc/*` goes to another.
- **Multiple-site hosting** Enables you to configure a more efficient topology for your deployments by adding different websites to a single Front Door and redirecting to different pools.
- **Session affinity** Uses cookie-based session affinity to keep the session in the same back end.
- **TLS termination** Support for TLS termination at the edge.
- **Custom domain and certificate management** You can let Front Door manage your certificate, or you can upload your own TLS/SSL certificate.
- **Application layer security** Allows you to author your own custom web application firewall (WAF) rules for access control, and it comes with Azure DDoS Basic enabled. Front Door is also a layer 7 reverse proxy, which means it only allows web traffic to

pass through to back ends and block other types of traffic by default.

- **URL redirection** Allows you to configure different types of redirection, which includes HTTP to HTTPS redirection, redirection to different hostnames, redirection to different paths, or redirections to a new query string in the URL.
- **URL rewrite** Allows you to configure a custom forwarding path to construct a request to forward traffic to the back end.

Tip Application Gateway

If your scenario requires a layer 7 (HTTP/HTTPS) load balancer just for one region, you can use Azure Application Gateway. If you need a global service that works across multiple regions, you should use Azure Front Door.

The diagram shown in Figure 2-31 reflects some of the features that were mentioned previously and gives you a better topology view of the main use case for Azure Front Door.

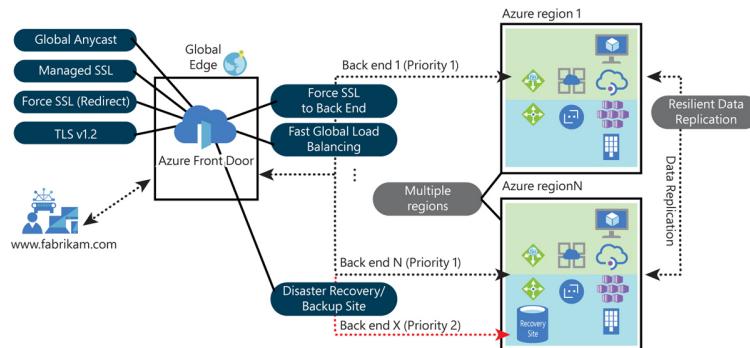


Figure 2-31 A use case for Azure Front Door

Follow the steps below to configure your Azure Front Door:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **front** and under **Services**, click **Front Doors**.
3. On the **Front Doors** page, click the **Add** button; the **Create A Front Door** page appears, as shown in Figure 2-32.

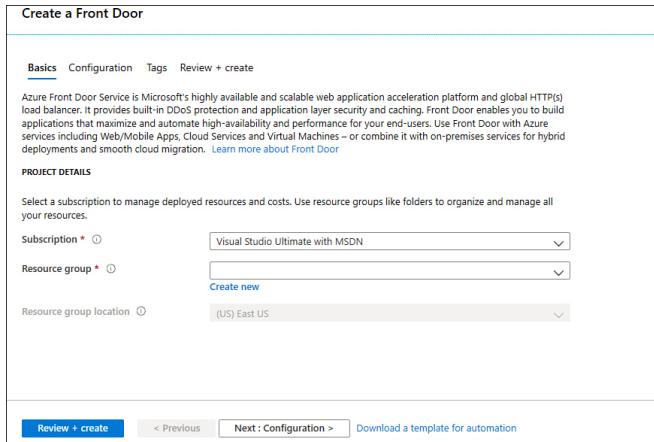


Figure 2-32 Azure Front Door creation page

4. In the **Subscription** drop-down menu, select the subscription that you want to use to create the Front Door.
5. In the **Resource Group** drop-down menu, select the resource group that you want for this Front Door.
6. Click the **Next: Configuration >** button; the **Configuration** tab appears, as shown in Figure 2-33.

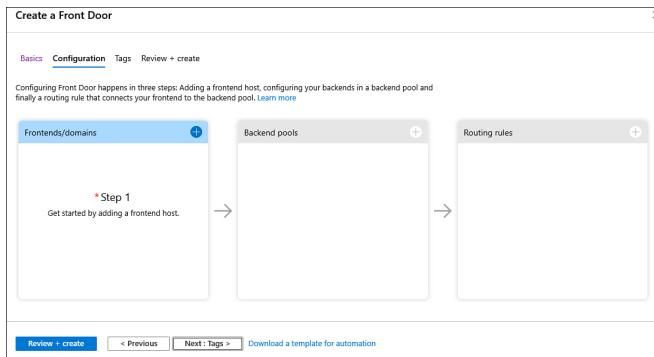


Figure 2-33 Initial Front Door configuration

7. Click the plus sign (+) in the first square, **Frontends/Domains**; the **Add Front End Host** blade appears, as shown in Figure 2-34.

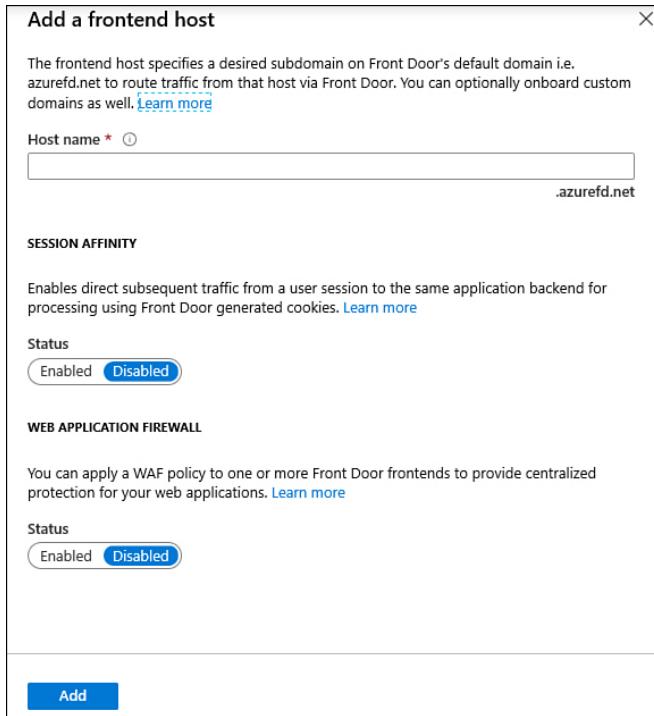


Figure 2-34 Add A Frontend Host

8. In the **Host Name** field, type a unique name for this front end.
9. Front Door forwards requests originating from the same client to different back ends based on load-balancing configuration, which means that Front Door doesn't use session affinity by default. However, some stateful applications usually prefer that subsequent requests from the same user land on the same back end that processed the initial request. In this case, you need to enable session affinity. For this example, leave the default selection in **Session Affinity (Enabled)**.
10. If you want to use Web Application Firewall (WAF) to protect your web application, you can take advantage of the centralized management provided by Front Door. For this example, leave the default **Disabled** setting for **Web Application Firewall** and click the **Add** button.
11. Click the plus sign (+) in the second square, **Back End Pools**; the **Add Back End Pool** blade appears, as shown in Figure 2-35.

Add a backend pool

A backend pool is a set of equivalent backends to which Front Door load balances your client requests. [Learn more](#)

Name *

BACKENDS

Backend host name	Status	Priority	Weight
Add a backend to get started			

+ [Add a backend](#)

HEALTH PROBES

Front Door sends periodic HTTP/HTTPS probe requests to each of your configured backends to determine the proximity and health of each backend to load balance your end user requests. [Learn more](#)

Status

Disabled Enabled

Path *

Protocol

HTTP HTTPS

Probe method

HEAD GET POST

Interval (seconds) *

LOAD BALANCING

Configure the load balancing settings to define what sample set we need to use to call the backend as healthy or unhealthy. The latency sensitivity with value zero (0) means always send it to the fastest available backend, else Front Door will round robin traffic between the fastest and the next fastest backends within the configured latency sensitivity. [Learn more](#)

Sample size *

Successful samples required *

Latency sensitivity (in milliseconds) *

Add

Figure 2-35 Add A Back End Pool

12. In the **Name** field, type a unique name for the back-end pool
13. In the **Back Ends** section, click **Add A Back End**; the **Add A Back End** blade appears, as shown in Figure 2-36.

Add a backend

← Go back to backend pool

Backends are your application servers where Front Door will route your client requests to. You can assign weights to your backends to define proportion of traffic to be sent and set priority for the backends to define active/stand-by kind of architectures. [Learn more](#)

Backend host type *

Backend host header ⓘ

HTTP port * ⓘ

80

HTTPS port * ⓘ

443

Priority * ⓘ

1

Weight * ⓘ

50

Status

Disabled **Enabled**

Add

Figure 2-36 Configuring a new backend

14. In the **Back End Host Type** drop-down menu, you can choose the type of resource you want to add. Select **App Service** in the drop-down menu.
15. Once you make this selection, the remaining parameters should be automatically filled with the default options. Review the values and click the **Add** button.
16. Now that you are back to the **Add Back End Pool** blade, review the options under the **Health Probes** section and notice that the default setting for **Probe Method** is **HEAD**. The **HEAD** method is identical to **GET**; the difference is that the server must not return a message-body in the response. This is also the recommended setting to lower load on and the cost of your back ends.
17. The **Load Balancing** settings for the back-end pool define how health probes are evaluated. These settings are used to determine whether the back end is healthy or unhealthy. The **Sample Size** is used to determine how many sample health probes are necessary to consider the state of the back end (health evaluation). The **Successful Samples Required** is the threshold for how

many samples must succeed to be considered successful. The **Latency Sensitivity** (in milliseconds) option is used when you want to send requests to back ends within the established latency measurement sensitivity range.

18. Leave the default selections and click the **Add** button.
19. Click the plus sign (+) in the third square; **Routing Rules**; the **Add Rule** blade appears, as shown in Figure 2-37.

The screenshot shows the 'Add a rule' configuration page. It includes fields for Name, Accepted protocol (HTTP and HTTPS), Frontends/domains (az500fd.azurefd.net), Patterns to Match (/* and /path), Route type (Forward selected), Backend pool (Az500BackendPool), Forwarding protocol (HTTPS only selected), URL rewrite (Disabled), Caching (Disabled), and an Add button at the bottom.

Add a rule

Name *

Accepted protocol ⓘ
HTTP and HTTPS

Frontends/domains
az500fd.azurefd.net

PATTERNS TO MATCH

Set this to all the URL path patterns that this route will accept. For example, you can set this to `/users/*` to accept all requests on the URL `www.contoso.com/users/*`. [Learn more](#)

/*
/path

ROUTE DETAILS

Once a route for a Front Door is matched, the configuration below defines the behavior of the route - forward and serve from the cache, or redirect. [Learn more](#)

Route type ⓘ
 Forward Redirect

Backend pool *

Az500BackendPool

Forwarding protocol ⓘ
 HTTPS only
 HTTP only
 Match request

URL rewrite ⓘ
 Enabled Disabled

Caching ⓘ
 Enabled Disabled

Add

Figure 2-37 Adding a new rule

20. In the **Name** field, type a unique name for this routing rule.
21. Under the **Patterns To Match** section, you can add a specific pattern that you want to use. When Front Door is evaluating the request, it looks for any routing with an exact match on the host. If no exact front-end hosts match, it rejects the request and sends a 400 Bad Request error. After determining the specific front-end host, Front Door will filter the routing rules based on the requested path. For this example, leave the default selections.

22. Under the **Route Details** section, you can configure the behavior of the route. In the **Route Type** option, you can select whether you want to forward to the back-end pool or redirect to another place. For this example, leave this set to **Forward**, which is the default. Enable the **URL Rewrite** option if you want to create a custom forwarding path. The **Caching** option is disabled by default, which means that requests that match to this routing rule will not attempt to use cached content. In other words, requests will always fetch from the back end. Leave all the default selections in this section and click the **Add** button.
23. Click **Review + Create** button, review the summary of your configuration, and click the **Create** button to finish.
24. Wait until the deployment is finished. Once it is finished, click the **Go To Resource** button to see the Front Door dashboard.

It will take a few minutes for the configuration to be deployed globally everywhere after you finish creating your Front Door.

Important Front Door Route

Routes for your Front Door are not ordered. A specific route is selected based on the best match.

Web application firewall

Web Application Firewall (WAF) can be used on Front Door. Azure also allows you to deploy WAF in other ways, so it is important to understand the design requirements before deciding which WAF deployment you should use.

Review the flowchart available at <http://aka.ms/wafdecisionflow> to better understand WAF's features, which include Azure Load Balancer, Application Gateway, and Azure Front Door. If your scenario has the following characteristic, WAF with Front Door is a good choice:

- Your app uses HTTP/HTTPS.
- Your app is Internet-facing.
- Your app is globally distributed across different regions.
- Your app is hosted in PaaS (such as an Azure App Service).

Consider deploying WAF on Front Door when you need a global and centralized solution. When using WAF with Front Door, the web applications will be inspected for every incoming request delivered by Front Door at the network edge.

Important WAF Integration with Front Door

If your deployment requires TLS offloading and package inspection, WAF natively integrates with Front Door, which allows you to inspect a request after it's decrypted.

Configure Web Application Firewall (WAF) on Azure Application Gateway

In a scenario where you need to protect your web applications from common threats such as SQL injection, cross-site scripting, and other web-based exploits, using Azure Web Application Firewall (WAF) on Azure Application Gateway is the most appropriate way address these needs. WAF on Application Gateway is based on Open Web Application Security Project (OWASP) core rule set 3.1, 3.0, or 2.2.9. These rules will be used to protect your web apps against the top 10 OWASP vulnerabilities, which you can find at <https://owasp.org/www-project-top-ten>.

You can use WAF on Application Gateway to protect multiple web applications. A single instance of Application Gateway can host up to 40 websites, and those websites will be protected by a WAF. Even though you have multiple websites behind the WAF, you can still create custom policies to address the needs of those sites. The diagram shown in Figure 2-38 has more details about the different components of this solution.

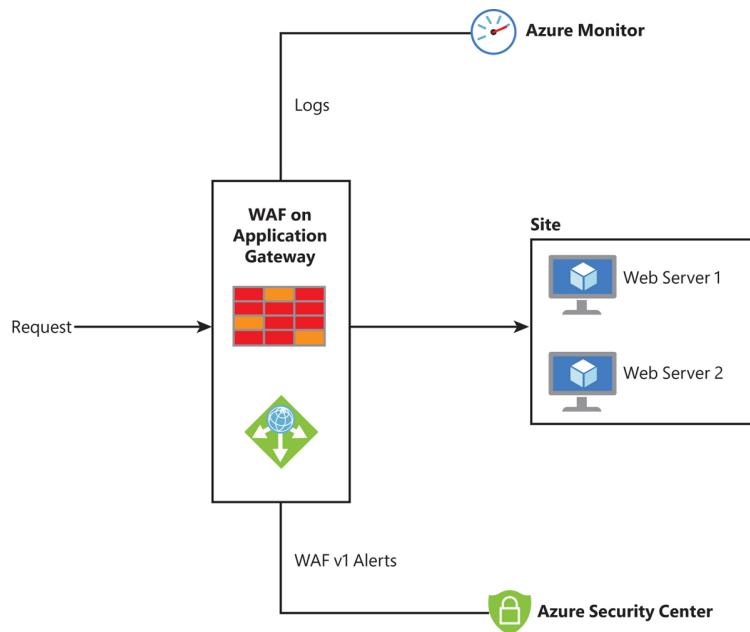


Figure 2-38 Different integration components for WAF on Application Gateway

In the example shown in Figure 2-38, a WAF Policy has been configured for the back-end site. This policy is where you define all rules, custom rules, exclusions, and other customizations, such as a file upload limit.

WAF on Application Gateway supports Transport Layer Security (TLS) termination, cookie-based session affinity, round-robin load distribution, and content-based routing. The diagram shown in Figure 2-38 also highlights the integration with Azure Monitor, which will receive all logs related to potential attacks against your web applications. WAV v1 alerts will also be streamed to Azure Security Center, and they will appear in the Security Alert dashboard.

Depending on the scenario requirement, you can configure WAF on the Application Gateway to operate in two different modes:

- **Detection mode** This mode will not interfere with traffic when suspicious activity occurs. Rather than blocking suspicious activity, this mode only detects and logs all threat alerts. For this mode to work properly, diagnostic logging and the WAF log must be enabled.

- **Prevention mode** As the name implies, this mode blocks traffic that matches the rules. Blocked requests generate a 403 Unauthorized Access message. At that point, the connection is closed, and a record is created in the WAF logs.

When reviewing the WAF log for a request that was blocked, you will see a message that contains some fields that are similar to this example:

[Click here to view code image](#)

```
Mandatory rule. Cannot be disabled. Inbound
Anomaly Score Exceeded (Total Inbound Score:
5 -
SQLI=0,XSS=0,RFI=0,LFI=0,RCE=0,PHPI=0,HTTP=0,SESS=0):
Missing User Agent Header;
individual paranoia level scores: 3, 2, 0, 0
```

The anomaly score comes from the OWASP 3.x rules, which have a specific severity: Critical, Error, Warning, or Notice. The previous message indicates that the total inbound score is 5, which translates to a severity equal to Critical. It is important to emphasize that the traffic will not be blocked until it reaches the threshold, which is 5. This means that if traffic matches the block rule but has an anomaly score of 3, it will not be blocked, though the message that you will see in the WAF log says that it is blocked. The severity levels are 5 (Critical), 4 (Error), 3 (Warning), and 2 (Notice).

Tip Application Gateway

To create an application gateway with a Web Application Firewall using the Azure portal, use the steps from this article:
<https://aka.ms/az500wafag>.

Configure Azure Bastion

Azure Bastion deployment is done per virtual network, which means that you provision the Azure Bastion service in the VNet and at that point, the RDP/SSH access will be available to all virtual machines that belong to the same VNet. The general architecture looks similar to Figure 2-39.

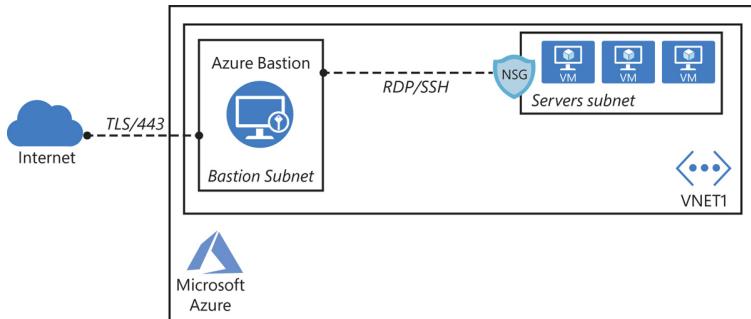


Figure 2-39 Core architecture for Azure Bastion deployment

Important Session Initiation

A session should be initiated only from the Azure portal. If you try to access the URL from another browser session or tab, you might receive the “Your session has expired” error.

When analyzing the scenario definition, you will identify clues that will lead you to use Azure Bastion. For example, in a scenario where Contoso administrators don't want to use a public IP on the VMs but need to provide external RDP access to those VMs. That's a typical scenario where Azure Bastion will be the best design choice. Another advantage of not exposing the public IP (v4 only) address is that your VM will not receive port scanning attacks.

Although Azure Bastion is going to receive external requests, you don't need to worry about hardening the service, since Azure Bastion is a fully managed PaaS service, and the Azure platform keeps Azure Bastion hardened and up to date for you. This approach also helps to prevent against zero-day exploits. Azure Bastion allows up to 25 concurrent RDP sessions and 50 concurrent SHC connections. Although this is the official limit, high-usage sessions can affect how Azure Bastion will answer to other connections, which means that it can allow less than the maximum if the usage is high.

To establish a connection to Azure Bastion, you need the Reader role on the virtual machine, Reader role on the NIC with private IP of the virtual machine, and Reader

role on the Azure Bastion resource. To create an Azure Bastion host using the portal, follow these steps:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **bastion**, and under **Services**, click **Bastions**.
3. On the **Bastions** page, click the **Add** button; the **Create A Bastion** page appears, as shown in Figure 2-40.

The screenshot shows the 'Create a Bastion' wizard in the Azure portal. The 'Basics' tab is active. The 'Project details' section includes a 'Subscription' dropdown set to 'Visual Studio Ultimate with MSDN' and a 'Resource group' dropdown with 'Create new' selected. The 'Instance details' section has a 'Name' field and a 'Region' dropdown set to 'West US'. Under 'Configure virtual networks', there's a 'Virtual network' dropdown with 'Create new' selected. The 'Public IP address' section shows 'Create new' is selected for the radio button. The 'Assignment' section shows 'Standard' as the SKU and 'Static' as the assignment type. At the bottom, there are buttons for 'Review + create', 'Previous', 'Next : Tags >', and 'Download a template for automation'.

Figure 2-40 Create A Bastion

4. Select the subscription and the resource group that you want to host your Azure Bastion.
5. In the **Instance Details** section, type the name of this Bastion, and select the region where it will reside.
6. Under **Configure Virtual Networks** section, select the virtual network in which the Bastion will be created, or if you don't have one available, you can click the **Create New** button and follow the steps to create a Bastion.
7. Select the Public IP address that will be used by the Bastion. You can either create one (default option) or use an existing one that is available.
8. Notice that the **Public IP Address SKU** option is prepopulated, and it doesn't allow you to change. That's because Azure Bastion only supports Standard Public IP SKU.

9. The **Assignment** option is prepopulated with **Static**. If you select **Use Existing IP Address**, this option will not be available because the setting established during the public IP creation will be used.
10. Click the **Review + Create** button.
11. Click the **Create** button.

At this point, the Bastion will be created, which usually takes five minutes to complete. After the Bastion is created, you will be able to connect to a VM using this Bastion. The option will appear when you click **Connect** in the VM blade, as shown in [Figure 2-41](#).

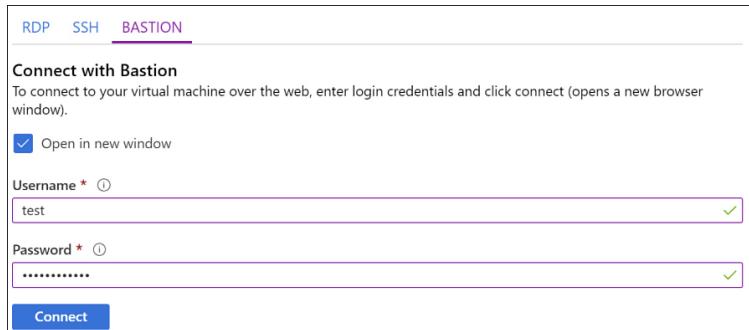


Figure 2-41 Accessing a VM using Azure Bastion

Configure resource firewall

In addition to Azure Firewall, you can also leverage the native firewall-related capabilities for different services. Azure Storage and SQL Database are examples of Azure services that have this functionality.

When you leverage this built-in functionality to harden your resources, you are adding an extra layer of security to your workload and following the defense in depth strategy, as shown in [Figure 2-42](#).

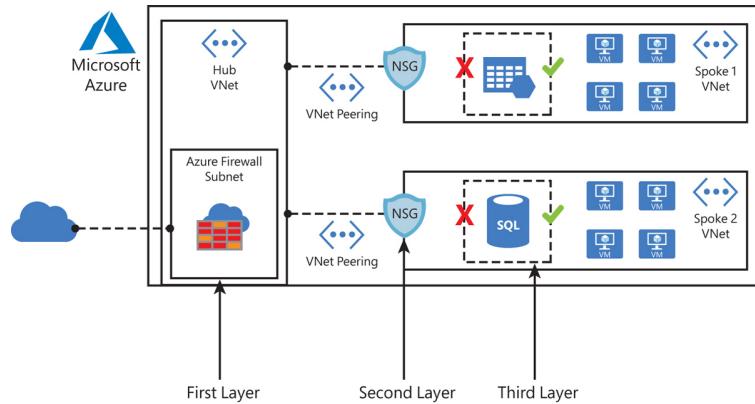


Figure 2-42 Multiple layers of protection to access the resource

Azure storage firewall

When you enable this feature in Azure Storage, you can better control the level of access to your storage accounts based on the type and subset of networks used. When network rules are configured, only applications requesting data over the specified set of networks can access a storage account.

You can create granular controls to limit access to your storage account to requests coming from specific IP addresses, IP ranges, or from a list of subnets in an Azure VNet. The firewall rules created on your Azure Storage are enforced on all network protocols that can be used to access your storage account, including REST and SMB.

Because the default storage accounts configuration allows connections from clients on any other network (including the Internet), it is a recommended that you configure this feature to limit access to selected networks. Follow these steps to configure Azure Storage firewall:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **storage**, and under **Services**, click **Storage Accounts**.
3. Click the storage account for which you want to modify the firewall settings.

- On the storage account page, under the **Settings** section in the left navigation pane, click the **Firewalls And Virtual Networks** option; the page shown in Figure 2-43 appears.

The screenshot shows the 'Firewalls and virtual networks' settings page for a storage account named 'yuridioatp'. The 'Allow access from' section has the 'All networks' radio button selected. The left sidebar lists other settings such as 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), and Firewalls and virtual networks.

Figure 2-43 Azure storage firewall and virtual network settings

- Under **Allow Access From**, click **Selected Networks**; the options shown in Figure 2-44 will become available.

The screenshot shows the 'Firewall settings' page. The 'Allow access from' section has the 'Selected networks' radio button selected. Below it, there's a note about firewall settings remaining in effect for up to a minute after saving updated settings. The 'Virtual networks' section allows adding existing or new virtual networks. The 'Firewall' section has an 'Address range' input field and a 'IP address or CIDR' dropdown. The 'Exceptions' section contains three checkboxes: 'Allow trusted Microsoft services to access this storage account', 'Allow read access to storage logging from any network', and 'Allow read access to storage metrics from any network'. There are also 'Save', 'Discard', and 'Refresh' buttons at the top.

Figure 2-44 Azure storage firewall settings

- Under the **Virtual networks** section, you could either add a new VNet or assign this storage account to a specific VNet.
- Under the **Firewall** section, you can harden the address range that can have access to this storage account. For that, you need to type the IP addresses or the range using CIDR format. Keep in mind that services deployed in the same region as the storage account use private Azure IP addresses for communication. Therefore, you cannot restrict access to specific Azure services based on their public outbound IP address range.
- Under the **Exceptions** section, you can enable or disable the following options:

- 1. Allow Trusted Microsoft Services To Access This Storage Account** Enabling this option will grant access to your storage account from Azure Backup, Azure Event Grid, Azure Site Recovery, Azure DevTest Labs, Azure Event Hubs, Azure Networking, Azure Monitor, and Azure SQL Data Warehouse.
 - 2. Allow Read Access To Storage Logging From Any Network** Enable this point if you want to allow this level of access.
 - 3. Allow Read Access To Storage Metrics From Any Network** Enable this option if you need the storage metrics to be accessible from all networks.
9. Once you finish configuring, click the **Save** button.

If you want to quickly deny network access to the storage account, you can use the Update-AzStorageAccountNetworkRuleSet cmdlet, as shown here:

[Click here to view code image](#)

```
Update-AzStorageAccountNetworkRuleSet -  
ResourceGroupName "MyRG" -Name "mystorage"  
-DefaultAction Deny
```

Azure SQL database firewall

When configuring your Azure SQL database, you can restrict access to a specific network by using the server-level firewall rules or database-level firewall rules. These rules can enable or disable access from clients to all the databases within the same SQL Database server. These rules are stored in the master database.

If your database is accessible from the Internet and a computer tries to connect to it, the firewall first checks the originating IP address of the request against the database-level IP firewall rules for the database that the connection requests. If the address isn't within a range in the database-level IP firewall rules, the firewall checks the server-level IP firewall rules.

The server-level firewall rules can be configured via the Azure portal, whereas the database-level firewall needs to be configured on the database itself by using the `sp_set_database_firewall_rule` SQL command. To configure the server-level firewall, follow these steps:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **database**, and under **Services**, click **SQL Databases**.
3. Click the database for which you want to modify the server-level firewall settings.
4. In the Overview page, click the **Set Server Rule** button, as shown in Figure 2-45.



Figure 2-45 Selecting the option to configure the server-level firewall

5. The Firewall settings page appears, as shown in Figure 2-46.

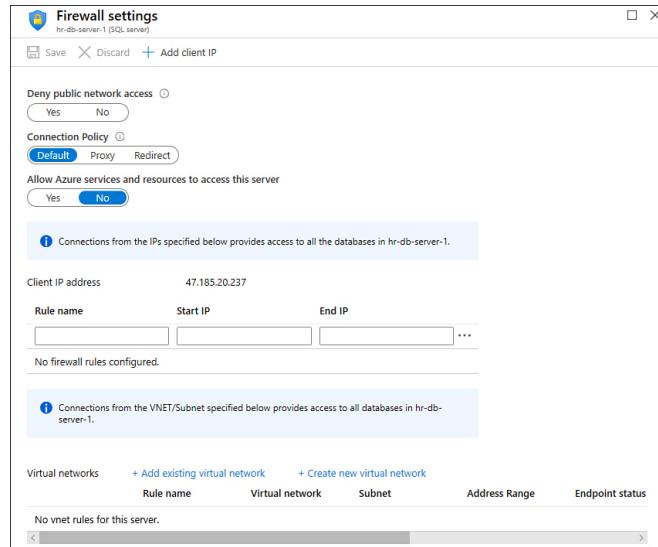


Figure 2-46 Server-level Firewall Settings options

6. Under **Deny Public Network Access** option, select **Yes** if you want to prohibit access from the Internet or **No** if you want to allow Internet access to this database.
7. The **Connection Policy** option allows you to configure how clients can connect to Azure SQL. The available options are

1. **Default** The default policy is basically a redirect for all client connections originating inside of Azure and proxy for all client connections originating outside.
 2. **Policy** By selecting this option, all connections are proxied via the Azure SQL Database gateways (which varies according to the Azure region). This setting will increase latency and reduce throughput.
 3. **Redirect** By selecting this option, all clients will establish connections directly to the node hosting the database, which reduces latency and improves throughput.
8. Under **Allow Azure Services And Resources To Access This Server**, you have the option to **Enable** or **Disable** this type of access.
9. Next are three fields, **Rule Name**, **Start IP**, and **End IP**, which allow you to create filters for client connections.
10. The last option that you can configure is the **Virtual Networks** setting, which allows you to either create or add an existing VNet.
11. Once you finish configuring, click the **Save** button.

Azure Key Vault Firewall

Just like the previous resources, Azure Key Vault also allows you to create network access restrictions by using Key Vault firewall, which applies to Key Vault's data plane. This means that operations such as creating a new vault or deleting or modify the settings won't be affected by the firewall rules. Below are two use-case scenarios for Azure Key Vault Firewall:

- Contoso needs to implement Azure Key Vault to store encryption keys for its applications. Contoso wants to block access to its keys for requests coming from the Internet.
- Fabrikam implemented Azure Key Vault, and now it needs to lock down access to its keys and enable access only to Fabrikam's applications and a short list of specific hosts.

To configure Azure Key Vault Firewall, you should first enable the Key Vault Logging using the following sequence of PowerShell commands:

[Click here to view code image](#)

```
$storagea = New-AzStorageAccount -  
ResourceGroupName ContosoResourceGroup -Name  
fabrikamkeyvaultlogs -Type Standard_LRS -Location
```

```
'East US'
$kvault = Get-AzKeyVault -VaultName
'ContosoKeyVault'
Set-AzDiagnosticSetting -ResourceId
$kvault.ResourceId -StorageAccountId $storagea.Id
-Enabled $true -Category AuditEvent
```

In this sequence, you will create a new storage account to store the logs, obtain the Key Vault information, and finally, configure the diagnostic setting for your Key Vault.

After finishing this part, you can go to Azure portal, open your Key Vault, and in the left navigation pane under the **Settings** section, click **Networking > Private Endpoint And Selected Networks**, as shown in Figure 2-47.

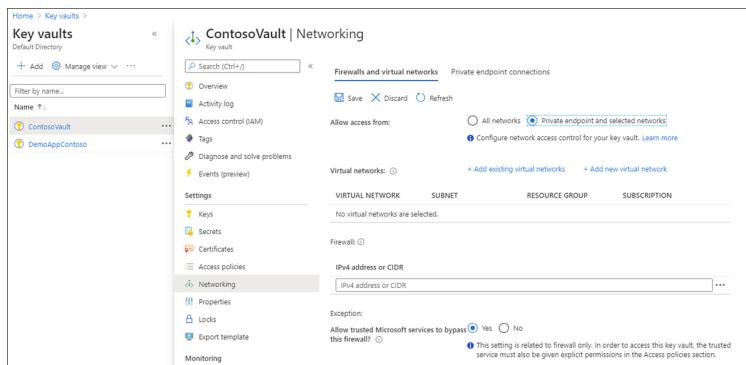


Figure 2-47 Azure Key Vault Firewall configuration

On this page, you can click the **Add Existing Virtual Networks** or **Add New Virtual Networks** options to start building your list of allowed virtual networks to access your Key Vault. Keep in mind that once you configure those rules, users can only perform Key Vault data plane operations when their requests originate from this list of allowed virtual networks. The same applies when users are trying to perform data plane operations from the portal, such as listing the keys.

Important IP Network Rules

If you are creating IP network rules, you can only use public IP addresses. Reserved IP address ranges are not allowed in IP rules.

Private networks include addresses defined with RFC 1918.

In Figure 2-47, notice the **Allow Trusted Microsoft Services To Bypass This Firewall** option, which is set to **Yes** by default. This will allow the following services to have access to your Key Vault regardless of the firewall configuration: Azure Virtual Machines deployment service, Azure Resource Manager template deployment service, Azure Application Gateway v2 SKU, Azure Disk Encryption volume encryption service, Azure Backup, Exchange Online, SharePoint Online, Azure Information Protection, Azure App Service, Azure SQL Database, Azure Storage Service, Azure Data Lake Store, Azure Databricks, Azure API Management, Azure Data Factory, Azure Event Hubs, Azure Service Bus, Azure Import/Export, and Azure Container Registry.

Azure App Service Firewall

You might also want to harden the network access for your apps that are deployed via Azure App Service. Although the terminology used in this section refers to “[Azure App Service Firewall](#),” what you are really implementing is a network-level access-control list. The access restrictions capability in Azure App Service is implemented in the App Service front-end roles. These front-end roles are upstream of the worker hosts where your code runs.

A common exam scenario for the implementation of this capability is when you need to restrict access to your app from certain VNets or the Internet. On the AZ-500 exam, make sure to carefully read the scenario because, in this case, you are adding restrictions to access the app itself, not the host.

To configure access restrictions on your Azure App Services, open the Azure portal, open the **App Services** dashboard, click your app service or Azure function, and

in the **Settings** section, click **Networking**. The **Access Restrictions** option is shown at the right (see [Figure 2-48](#)).

The screenshot shows the Azure App Services interface for an app named 'functionsyd'. On the left, there's a sidebar with options like 'Add', 'Manage view', 'Deployment slots', 'Deployment Center', 'Settings', 'Configuration', 'Authentication / Authorization', 'Application Insights', 'Identity', 'Backups', 'Custom domains', 'TLS/SSL settings', and 'Networking'. The 'Networking' option is highlighted. The main pane is titled 'functionsyd | Networking' and contains sections for 'Azure Front Door with Web Application Firewall' (with a 'Learn More' link) and 'Azure CDN' (with a 'Learn More' link). Below these is the 'Access Restrictions' section, which includes a sub-section for 'Define and manage rules that control access to your application.' There's also a 'Configure Access Restrictions' link.

Figure 2-48 Azure App Services access restriction

To start the configuration, click **Configure Access Restrictions** in the **Access Restriction** section. You will see the **Access Restriction** page, as shown in [Figure 2-49](#). The initial table is blank (no rules), and you can click **Add Rule** to start configuring your restrictions.

The screenshot shows the 'Access Restrictions' configuration page for the 'functionsyd' app. At the top, there are 'Remove' and 'Refresh' buttons. Below that is a section titled 'Access Restrictions' with a sub-section for 'Define and manage lists of allow/deny rules to control traffic to your app. Rules are evaluated in priority order. If there are no rules defined then your app will accept traffic from any address.' A 'Learn more' link is provided. The main area is a table with columns: 'Priority' (checkbox), 'Name' (text input), 'Source' (dropdown), 'Endpoint status' (dropdown), and 'Action' (checkbox). A single row is present with a Priority of 1, Name 'Allow all', Source 'Any', Action checked (green checkmark), and Endpoint status 'Allow'.

Figure 2-49 Adding Access Restrictions

It is recommended that you schedule a maintenance window to configure these restrictions because any operation (add, edit, or remove) in those rules will restart your app for changes to take effect.

Implement service endpoint

You can also have a VNet that has only PaaS services and allow these services to be accessible outside of the VNet

in which they reside. For example, the database admin needs to access the Azure SQL Database from the Internet. In this scenario, the database admin needs to create a service endpoint to allow secure access to the database.

At the time this chapter was written, the following Azure services supported service endpoint configuration:

- Azure Storage
- Azure SQL Database
- Azure SQL Data Warehouse
- Azure Database for PostgreSQL server
- Azure Database for MySQL server
- Azure Database for MariaDB
- Azure Cosmos DB
- Azure Key Vault
- Azure Service Bus
- Azure Event Hubs
- Azure Data Lake Store Gen 1
- Azure App Service
- Azure Container Registry

Important Network Updates

For the most updated list of supported service endpoint, see
<https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview>.

From a security perspective, service endpoints provide the ability to secure Azure service resources to your VNet by extending the VNet identity to the service. After enabling service endpoints in your VNet, you can add a VNet rule to secure the Azure service resources to your VNet. By adding this rule, you are enhancing the security by fully removing public Internet access to resources and allowing traffic only from your virtual network.

Another advantage of using service endpoint is the traffic optimization. Service endpoint always takes service

traffic directly from your VNet to the service on the Microsoft Azure backbone network, which means that the traffic is kept within the Azure backbone network. By having this control, you can continue auditing and monitoring outbound Internet traffic from your VNet without affecting service traffic.

Important Deployment Model

This feature is available only to virtual networks deployed through the Azure Resource Manager deployment model.

The VNet service endpoint allows you to harden the Azure service access to only allowed VNet and subnet access. This adds an additional level of security to the network and isolates the Azure service traffic. All traffic using VNet service endpoints flows over the Microsoft backbone, thus providing another layer of isolation from the public Internet. You can also fully remove public Internet access to the Azure service resources and allow traffic only from their virtual networks through a combination of IP firewall and access control list on the VNet, which protects the Azure service resources from unauthorized access.

To configure virtual network service endpoint, you will need to perform these two main actions:

- Enable service endpoint in the subnet
- Add a service endpoint to your VNet

If you are configuring Azure Storage, you also need to configure a service endpoint policy.

Note Vnet Service Policy

For more information on Azure VNet service endpoint policies for Azure Storage, see <https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoint-policies-overview>.

Enabling a service endpoint on the subnet can be done during the creation of the subnet or after the subnet is

created. In the properties of the subnet, you can select the service endpoint in the **Services** drop-down menu, as shown in [Figure 2-50](#).

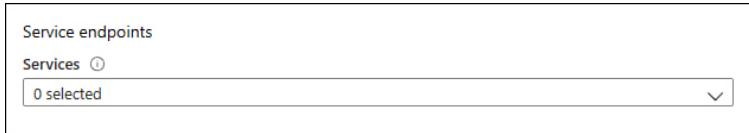


Figure 2-50 Service Endpoints configuration on the subnet

To configure virtual network service endpoint on your virtual network, use the following steps:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **virtual networks**; under **Services**, click **Virtual Networks**.
3. Click the virtual network for which you want to configure the service endpoint.
4. In the left pane, click **Service Endpoint**, as shown in [Figure 2-51](#).

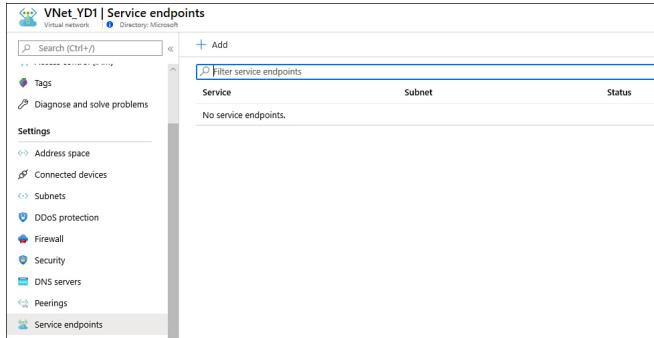


Figure 2-51 Configuring a VNet service endpoint

5. Click the **Add** button.
6. In the **Add Service Endpoints** page, click the drop-down menu and select the Azure Service that you want to add.

Implement DDoS

By default, Azure Distributed denial of service (DDoS) basic protection is already enabled on your subscription. This means that traffic monitoring and real-time mitigation of common network-level attacks are fully

covered and providing the same level of defense as the ones utilized by Microsoft's online services.

While the basic protection provides automatic attack mitigations against DDoS, there are some capabilities that are only provided by DDoS Standard tier. The organization's requirements will lead you to determine which tier you will utilize. In a scenario where Contoso needs to implement DDoS protection on the application level, it needs to have real-time attack metrics and resource logs available to its team. Contoso also needs to create post-attack mitigation reports to present to upper management. These requirements can only be fulfilled by the DDoS Standard tier. Table 2-2 provides a summary of the capabilities available for each tier:

Table 2-2 Azure DDoS Basic versus Standard

Capability	DDoS Basic	DDoS Standard
Active traffic monitoring and always-on detection	X	X
Automatic attack mitigation	X	X
Availability guarantee	Per Azure region.	Per application.
Mitigation policies	Tuned per Azure region volume.	Tuned for application traffic volume.
Metrics and alerts	Not available.	X
Mitigation	Not available.	X

flow logs		
Mitigation policy customization	Not available.	X
Support	Yes, but it is a best-effort approach. In other words, there is no guarantee support will address the issue.	Yes, and it provides access to DDoS experts during an active attack.
SLA	Azure region.	Application guarantee and cost protection.
Pricing	Free.	Monthly usage.

Tip Attacks Covered by Azure Ddos

For more information about the different types of attacks that are covered by Azure DDoS, visit <http://aka.ms/az500DDoS>.

To configure Azure DDoS, your account must be a member of the Network Contributor role, or you can create a custom role that has read, write, and delete privileges under

`Microsoft.Network/ddosProtectionPlans` and action privilege under

`Microsoft.Network/ddosProtectionPlans/join`. Your custom role also needs to have read, write, and delete privileges under

`Microsoft.Network/virtualNetworks`. After you grant access to the user, use the following steps to create a DDoS Protection plan:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **DDoS** and under **Services**, click **DDoS Protection Plans**.
3. On the **DDoS Protection Plans** page, click the **Add** button; the **Create A DDoS Protection Plan** page appears, as shown in

Figure 2-52.

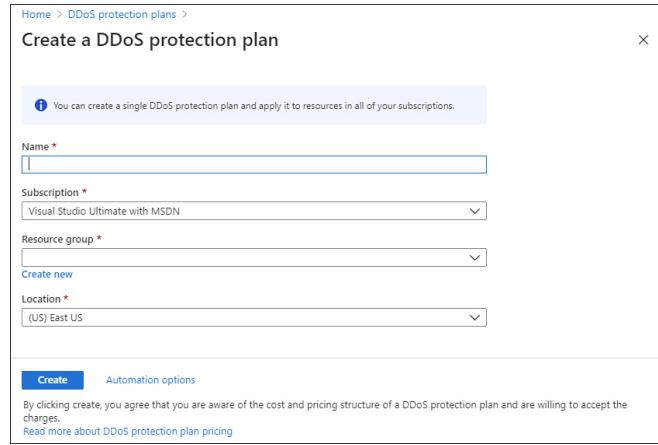


Figure 2-52 Create A DDoS Protection Plan

4. In the **Name** field, type the name for this DDoS protection.
5. In the **Subscription** field, select the appropriate subscription.
6. In the **Resource group** field, click the drop-down menu and select resource group that you want.
7. In the **Location** field, select the region for the DDoS.
8. Before you click **Create** button, read the note that is located under this button. This note emphasizes that by clicking **Create**, you are aware of the pricing for DDoS protection. Because there is no trial period for this feature, you will be charged during the first month of utilizing this feature.
9. After clicking **Create**, go to the search bar, type **network**, and click **Virtual Networks**.
10. Click the virtual network for which you want to enable the DDoS Standard.
11. In the left navigation pane, click the **DDoS Protection** option.
12. Click the **Standard** option, as shown in Figure 2-53.

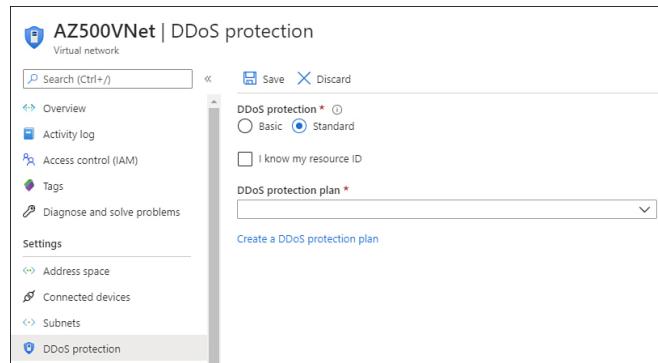


Figure 2-53 Enabling DDoS Standard on the VNet

13. Click the **DDoS Protection Plan** drop-down menu and select the DDoS protection plan that you created in step 9.

14. Click the **Save** button.

At this point, you can configure Azure Monitor to send alerts by leveraging DDoS protection metrics. To do that, open Azure Monitor, click **Metrics**, select the scope of the public IP address located in the VNet where DDoS Standard is enabled, click the **Metric** drop-down menu, and select **Under DDoS Attack Or Not**, as shown in Figure 2-54.

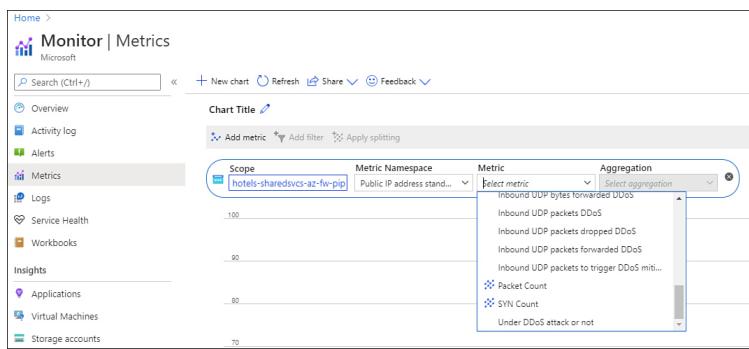


Figure 2-54 Monitoring DDoS activity

To access a DoS attack mitigation report, you need to first configure diagnostic settings. This report uses the Netflow protocol data to provide detailed information about the DDoS attack on your resource. To configure this option, click **Diagnostic Settings** in the **Settings** section in the Azure Monitor blade, as shown in Figure 2-55.

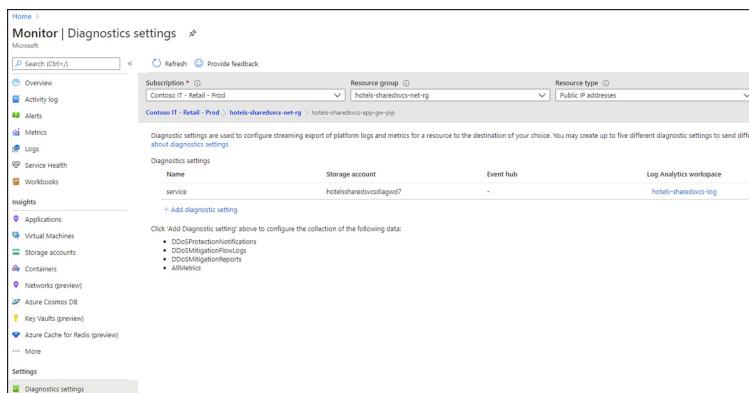


Figure 2-55 Configuring diagnostic logging

As you can see in the bottom part of the right blade, this page allows you to configure diagnostic logging for DDoSProtectionNotifications, DDoSMitigationFlowLogs, and DDoSMitigationReports. Just like any other diagnostic setting, you can store this data in a storage account, Event Hub, or in a Log Analytics workspace.



Exam Tip

For the AZ-500 exam, always make sure to review the details of the use case to ensure you are selecting the most appropriate option according to the scenario description.

Besides these options, is important to mention that Azure Security Center will also surface security alerts generated by DDoS Protection. There are two main alerts that could be triggered by this service and surfaced in Security Center:

- DDoS Attack detected for Public IP
- DDoS Attack mitigated for Public IP

SKILL 2.2: CONFIGURE ADVANCED SECURITY FOR COMPUTE

This section of the chapter covers the skills necessary to configure advanced security for compute, according to the Exam AZ-500 outline.

Configure endpoint security within the VM

Endpoint security is an imperative part of your security strategy, and these days, you can't have endpoint protection without an antimalware solution installed on your computer.

Consider a scenario in which you provision a new VM that doesn't have an endpoint protection configured. Wouldn't it be ideal to have a solution that alerts you to the fact that an endpoint protection is missing in that VM? This is exactly what happens when you have Azure Security Center (Free or Standard tiers) enabled in your subscription.

Follow these steps to access Security Center and review the endpoint protection recommendations:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **security** and under **Services**, click **Security Center**.
3. In Security Center main dashboard, under the **Resource Security Hygiene** section, click **Compute & Apps**.
4. In the resulting list, click the **Install Endpoint Protection Solution On Virtual Machines** option; the **Endpoint Protection Not Installed On Azure VMs** page appears, as shown in Figure 2-56.

Endpoint Protection not installed on Azure VMs				
		Filter	Install on 3 VMs	X
Virtual machine		↑↓	State	↑↓ Severity ↑↓
AZ500VM1	<input checked="" type="checkbox"/>		Open	High
AZ500VM2	<input checked="" type="checkbox"/>		Open	High
YD2020SRV16	<input checked="" type="checkbox"/>		Open	High

Figure 2-56 List of VMs that don't have an endpoint protection solution installed

5. Select the VM on which you want to install the endpoint protection and click the **Install On 1 VM** button. The **Select Endpoint Protection** page appears, as shown in Figure 2-57.



Figure 2-57 Selecting the available endpoint protection solution to install

6. Security Center automatically suggests that you install the Microsoft Antimalware for Azure, which is a free real-time protection that helps identify and remove viruses, spyware, and other malicious software. Click the **Microsoft Antimalware** option; the **Microsoft Antimalware** page appears, as shown in Figure 2-58.

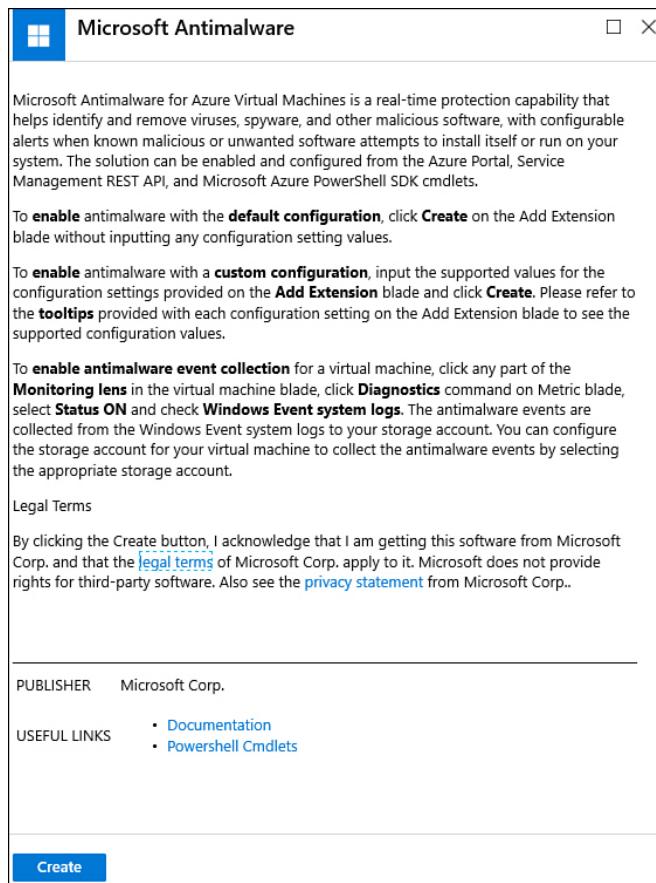
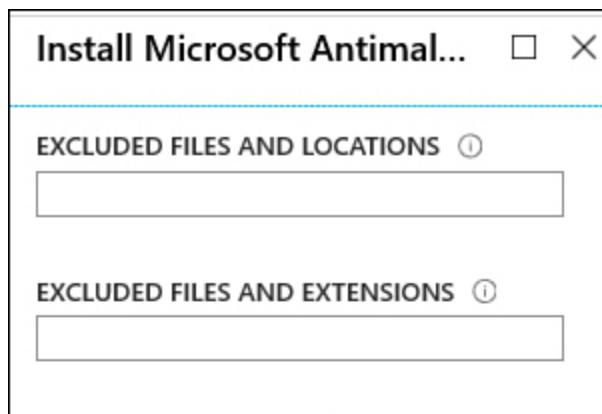


Figure 2-58 Microsoft Antimalware installation

7. Click the **Create** button; the **Install Microsoft Antimalware** blade appears, as shown in Figure 2-59.



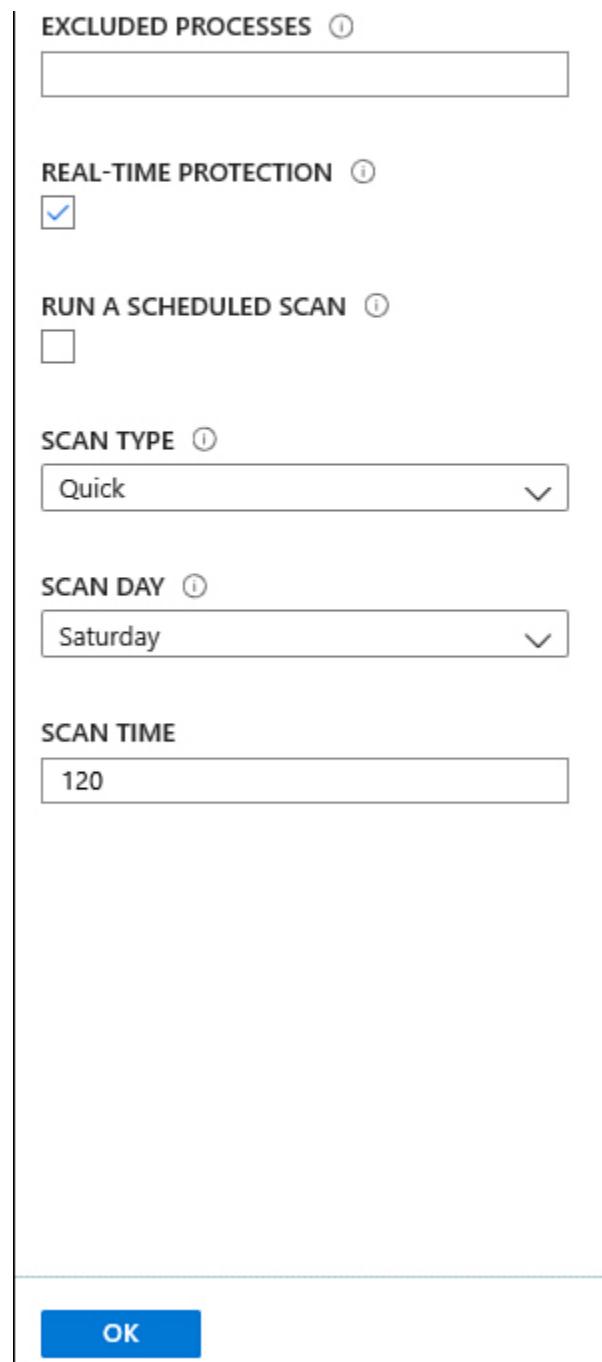


Figure 2-59 Installation options

8. If you need to create an endpoint protection exclusion list, this is where you would do that. For example, let's say you are aware that you want to avoid issues caused by antimalware scans of the files used by your app. You can add the paths used by this application in the exclusion list. This blade contains the following options:
 1. **Excluded Files And Locations** Here, you can specify any paths or locations to exclude from the scan. To add

multiple paths or locations, separate them with semicolons. This is an optional setting.

2. **Excluded Files And Extensions** This box lets you specify filenames or extensions to exclude from the scan. Again, to add multiple names or extensions, you separate them with a semicolon. Note that you should avoid using wildcard characters.
3. **Excluded Processes** Use this box to specify any processes that should be excluded from the scan. Again, use semicolons to separate multiple processes.
4. **Real-Time Protection** By default, this check box is enabled. Unless you have a good business reason to do otherwise, you should leave it that way.
5. **Run a Scheduled Scan** Selecting this check box enables you to run a scheduled scan.
6. **Scan Type** If you selected the **Run A Scheduled Scan** check box, you can use this drop-down menu to specify the type of scan. (A quick scan is run by default.)
7. **Scan Day** If you selected the **Run a Scheduled Scan** check box, you can use this drop-down menu to specify the day that the scan will run.
8. **Scan Time** If you selected the **Run a Scheduled Scan** check box, you can use this drop-down menu to specify what time the scan will run. The time is indicated in increments of 60 minutes (60 = 1 AM, 120 = 2 AM, and so on).
9. After you customize the options according to your needs, click the **OK** button.
10. After this step, the installation process will start. You can close the Security Center dashboard at this point.

Often, you will want to see an immediate reflection of the changes you made in the dashboard. However, be aware that the Security Center dashboard has different refresh times, which vary according to the objects. For example, operating system security configurations data are updated within 48 hours, and Endpoint Protection data is updated within 8 hours. This means that even the installation of the endpoint succeeds in the next five minutes after you started, the dashboard will only reflect that installation in the next refresh cycle.

Having said that, it is important to mention that, if the antimalware that was installed on the machine identifies

a malicious code running, it will immediately trigger an alert. This alert will appear in Security Center security alerts dashboard, as shown in [Figure 2-60](#).



Figure 2-60 The Alert that appears in Security Center when Microsoft Antimalware takes an action

When you open this alert, you will see more details about the operation, which includes the attacked resource, subscription, threat status, and file path, as shown in [Figure 2-61](#).

A screenshot of the Azure Security Center alert details page. The title is 'Antimalware Action Taken'. Below it, there's a 'Learn more' link and a 'General information' section. The 'General information' section contains the following details:

DESCRIPTION	Microsoft Antimalware has taken an action to protect this machine from malware or other potentially unwanted software.
ACTIVITY TIME	Tuesday, April 14, 2020, 1:09:06 PM
SEVERITY	Low
STATE	Active
ATTACKED RESOURCE	DIV4VM2016
SUBSCRIPTION	CONTOSO
DETECTED BY	Microsoft Antimalware
ACTION TAKEN	Blocked
ENVIRONMENT	Azure
RESOURCE TYPE	Virtual Machine
THREAT STATUS	Quarantined
CATEGORY	Virus
THREAT ID	2147519003
FILE PATH	C:\Users\yuridic\temp\iecar.com

Figure 2-61 Details about an alert triggered in Azure Security Center when malware is detected

Having an endpoint protection installed is only the first step to enhance the overall protection of your VM. There are many other aspects of VM security that need to be taken into consideration, and hardening is one of those. (See the next section.) Beyond hardening, what else can be implemented to secure a VM? Let's start with access control. In a scenario in which an organization has multiple subscriptions, you might need a way to manage access efficiently. Establishing a good access control policy is a one way to do just that.

In Azure, you can use Azure policies to create conventions for resources and create customized policies to control access. You can apply these policies to resource groups and the VMs that belong to those resource groups will inherit those policies. You can

implement those policies at the management group level if you have multiple subscriptions that should receive the same policy.

When configuring access control, always make sure to use the least-privilege approach. You can leverage built-in Azure roles to allow users to access and set up VMs. Instead of giving a higher level of access, you can assign a user to the Virtual Machine Contributor role, and that user will inherit the rights to manage VMs, though the user won't be able to manage the virtual network or storage account to which he or she is connected. The same applies for users who need access to Azure Security Center to visualize the recommendations for their VMs; they should have Security Reader role, which will enable them to see recommendations but will not allow them make changes to the configuration.

While the Security Center Free tier provides good insights regarding the current security posture of your workloads, you should also consider the threat detection for VMs that comes with Security Center Standard tier.

Security Center standard tier has Virtual Machine Behavioral Analysis (VMBA) that uses behavioral analytics to identify compromised resources based on an analysis of the virtual machine (VM) event logs, such as processing creation events and log in events. If your scenario requires detection of attacks against your VMs, Security Center standard tier must be enabled.

VMs threat detections in Security Center Standard tier are applicable for Windows and Linux operating systems. Figure 2-62 shows an example of a threat detection based on VMBA in Security Center. This alert appears in the **Security Alerts** dashboard in Security Center.

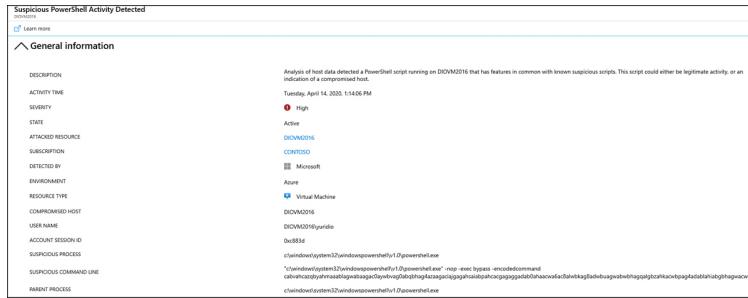


Figure 2-62 Example of a VM threat detection in Security Center

Threat detection is an important security control, though there are other security controls that must also be in place and that are categorized as proactive measures, or proactive security controls.

Disk encryption should also be applied to your VMs. Consider a scenario where the organization needs to ensure that encryption is in place no matter where the data is located (at rest or in-flight) and you need to quickly identify whether data is encrypted. Even the Security Center free tier can give you this level of visibility.

Security Center will trigger a recommendation when it identifies VMs that don't have disk encryption enabled. Another aspect of VM security is the identification of resource abuse. When VM processes consume more resources than they should, this could also be an indication of a suspicious activity. Without a doubt, performance issues could happen for a variety of issues, including an application that was not well-written. Performance issues might also happen because the VM is running out of resources because the valid load is high. (In this case, you need to upgrade the VM with more resources.) Whatever the cause may be, the bottom line is that a VM's performance can lead to service disruption, which directly violates the security principle of availability.

You can use Azure Monitor to obtain visibility of your VM's health. By leveraging Azure Monitor's features such

as resource diagnostic log files, you can identify potential issues that might compromise performance and availability. Azure Monitor and diagnostic logging are covered in more detail in Chapter 3, “Manage security operations.”

Configure system updates for VMs in Azure

Keeping the system up to date is another imperative measure for any organization that wants to implement host security. The good news is that in Azure you have two major services that can be used to ensure that your VMs are fully up to date.

Consider a scenario where you need to manage operating system updates for your Windows and Linux VMs, not only in Azure but also on-premises and in any other cloud environments. You can use the Update Management solution in Azure Automation to manage your VMs. Following are the components used by Update Management:

- **Log Analytics agent for Windows or Linux** This is the same agent used by Security Center, which means you should have it already installed if you are using Security Center.
- **PowerShell Desired State Configuration (DSC) for Linux** The management platform in PowerShell running on Linux.
- **Automation Hybrid Runbook Worker** Each Windows machine that is managed by the solution is listed in the Hybrid worker groups.
- **Microsoft Update or Windows Server Update Services (WSUS) for Windows machines** The update management platform managed by Microsoft (Microsoft Update) or managed by your organizations (WSUS).

Update management collection is done via a scan that is performed twice per day for each managed Windows server (clients are not supported) and every hour for Linux machines. The following versions of the operating systems are supported by this solution:

- Windows Server 2019 (Datacenter/Datacenter Core/Standard)

- Windows Server 2016 (Datacenter/Datacenter Core/Standard)
- Windows Server 2012 R2(Datacenter/Standard)
- Windows Server 2012
- Windows Server 2008 R2 RTM and SP1 Standard (assessment only, patching is not supported)
- CentOS 6 (x86/x64) and 7 (x64)
- Red Hat Enterprise 6 (x86/x64) and 7 (x64)
- SUSE Linux Enterprise Server 11 (x86/x64) and 12 (x64)
- Ubuntu 14.04 LTS, 16.04 LTS, and 18.04 (x86/x64)

You can enable Update Management solution directly from the VM's properties, which is a good approach if you only need to enable this solution for one VM. If you need to deploy to all VMs, you can select all VMs at once from the **Virtual Machines** dashboard and deploy to all VMs from there. VMs can be spread across up to three resources groups when enabling this solution for multiple VMs. Follow these steps to enable this feature for multiple VMs:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **virtual machine**, and under **Services**, click **Virtual Machines**.
3. Click the check box next to the field **Name** to select all VMs.
4. Click the **Services** button and click **Update Management**; the **Enable Update Management** page appears, as shown in Figure 2-63.

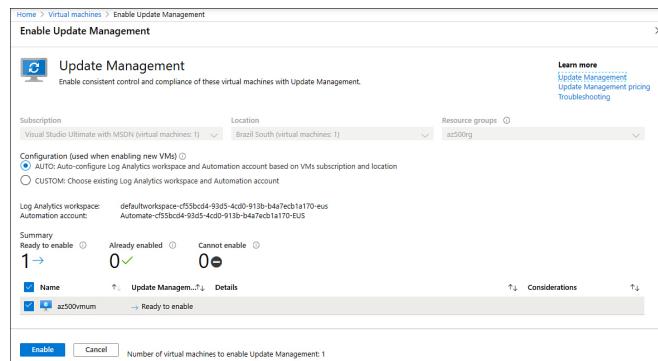


Figure 2-63 Enabling Update Management for VMs

5. Notice that the default configuration has the **AUTO** option selected. This option will auto-configure Log Analytics workspace and automation account based on your VMs subscription and location. If you already have VMs deployed with the Log Analytics

and the agent is already configured to report to a specific workspace, the auto-configuration won't work; you need to select **CUSTOM** and from there select the workspace where the VM resides as well as the Azure automation account that will be used by Updated Management.

- For this example, leave the default selection and click the **Enable** button.

The deployment of this solution can take some time depending on the amount of VMs that you select; wait until it is fully finished before proceeding.

Managing updates

Now that the Update Management solution is deployed to your VMs, you can access its dashboard to visualize the list of missing updates and schedule update deployments. To access the Update Management dashboard, use the following steps:

- Navigate to the Azure portal at <https://portal.azure.com>.
- In the search bar, type **automate** and under **Services**, click **Automated Accounts**.
- Click the automation account that is used by your Update Management solution.
- In the left pane, click **Update Management**, and if the scan is completed, the list of updates will appear, as shown in Figure 2-64.

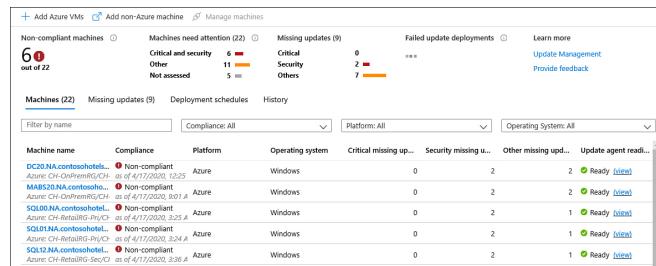


Figure 2-64 Update Management dashboard

- Click the **Missing Updates** tab to visualize the updates that are currently missing on the machines (see Figure 2-65).

Update name	Classification	Machines missing updates	Operating system	Information link
2020-04 Cumulative Update for Windows Server 2016 for x64-based Systems (...	● Security updates	6	Windows	KB4550929
2020-04 Servicing Stack Update for Windows Server 2016 for x64-based Syste...	● Security updates	5	Windows	KB4550994
Security Intelligence Update for Windows Defender Antivirus - KB2267602 (Ve...	▲ Definition updates	11	Windows	KB2267602
Security Intelligence Update for Windows Defender Antivirus - KB2267602 (Ve...	▲ Definition updates	4	Windows	KB2267602
Security Intelligence Update for Windows Defender Antivirus - KB2267602 (Ve...	▲ Definition updates	1	Windows	KB2267602
Security Intelligence Update for Windows Defender Antivirus - KB2267602 (Ve...	▲ Definition updates	1	Windows	KB2267602

Figure 2-65 Update Management dashboard

In the example given in the previous steps, you saw an environment that was already in production, with machines already reporting to Update Management and a deployment schedule already created. In a new deployment, you will see that there is a **Schedule Update Deployment** button in the main **Update Management** dashboard, as shown in Figure 2-66.



Figure 2-66 Option to schedule the deployment of the updates

Configure authentication for containers

There are multiple ways to configure authentication to secure Kubernetes clusters. You can leverage Azure RBAC to manage access based on users or groups and which resources they need to access, or you can also integrate with Azure Active Directory (AD). If you decide to use RBAC, you need to first assign permissions to users with Kubernetes RBAC. If you need the same permission applied to resources across the entire cluster, you should use a ClusterRole.

To enhance the AKS clusters security, you should take advantage of the integration with Azure AD, which allows you integrate on-premises identities into AKS clusters to provide a single source for account management and security. Azure AD authentication is provided to AKS clusters that have OpenID Connect, which is an identity layer built on top of the OAuth 2.0 protocol.

You should not think of one as being better than the other; in reality, you will use RBAC and Azure AD to make sure you have a stronger authentication method. The diagram shown in [Figure 2-67](#) shows an example of how both work together.

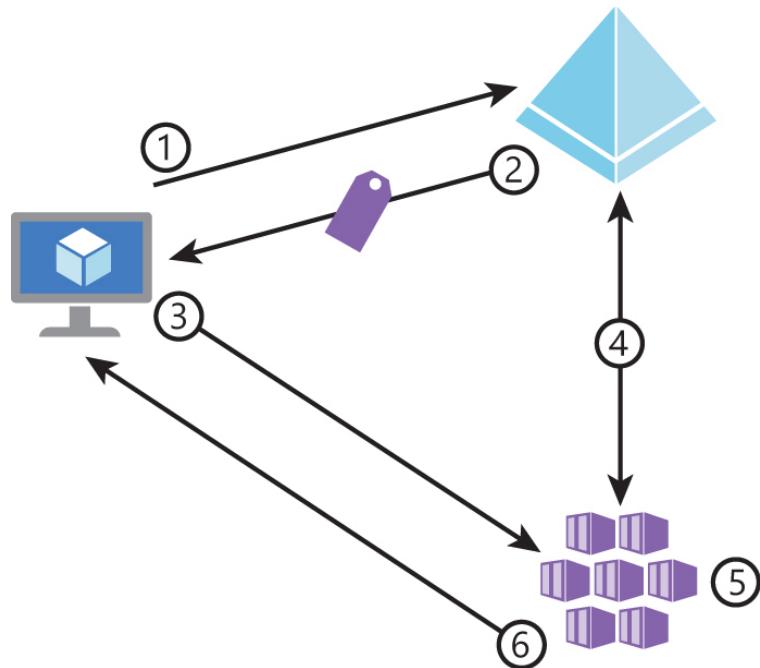


Figure 2-67 Authentication process

Step 1 of the authentication process shown in [Figure 2-67](#) is the client authenticating to Azure AD, which issues an access token (step 2). The user starts a task (such as creating a pod) that leverages this token (step 3). AKS validates this token with Azure AD (step 4), and during this validation, it retrieves the user's group membership. AKS RBAC and cluster policies are applied (step 5). Lastly in step 6, AKS answers the user's request, allowing or denying the request based on the validation process, which includes the Azure AD validation, the group membership (RBAC), and policies.

When you are creating your AKS Cluster, you can define whether the cluster will use RBAC in the Authentication tab, as shown in [Figure 2-68](#).

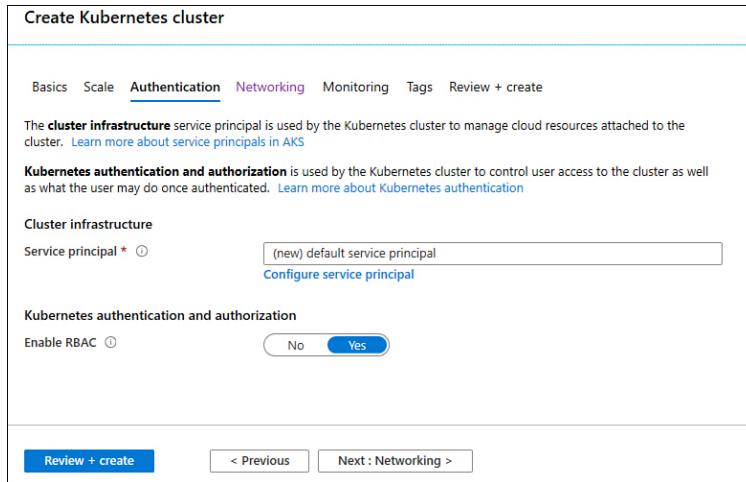


Figure 2-68 Authentication options during the AKS cluster creation

If you click **Configure Service Principal**, you have the option to either create a new one (which Azure will do for you) or use an existing one. If you choose to use an existing one, you need the service principal client ID, which is your AppID and the service principal client secret, which is the password value. Every service principal is associated with an Azure AD application. The service principal for an AKS cluster can be associated with any valid Azure AD application name. If you need to manually create a service principal, you can use the following Azure CLI command:

[Click here to view code image](#)

```
az ad sp create-for-rbac --skip-assignment --name myAKSClusterSP
```

As a security best practice, make sure to create roles and bindings that assign the least amount of privilege required. When possible, integrate with Azure AD so any change in user status or group membership is automatically updated and access to cluster resources is current.

Configure security for different types of containers

There are many built-in capabilities in AKS that help ensure that your AKS Cluster is secure. Those built-in capabilities are based on native Kubernetes features, such as network policies and secrets, with the addition of Azure components, such as NSG and orchestrated cluster upgrades.

The combination of these components is used to keep your AKS cluster running the latest OS security updates and Kubernetes releases, secure pod traffic, and provide access to sensitive credentials. [Figure 2-69](#) shows a diagram with the core AKS security components.

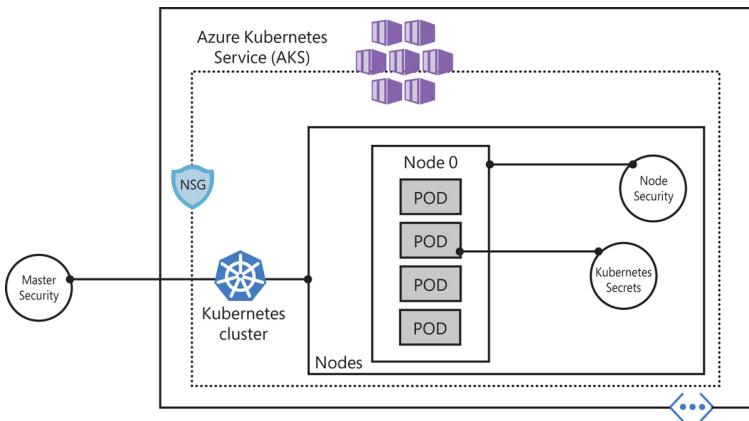


Figure 2-69 Core AKS security components

When you deploy AKS in Azure, the Kubernetes master components are part of the managed service provided by Microsoft. Each AKS cluster has a dedicated Kubernetes master. This master is used to provide API Server, Scheduler, and so on. You can control access to the API server using Kubernetes RBAC controls and Azure AD.

While the Kubernetes master is managed and maintained by Microsoft, the AKS nodes are VMs that you manage and maintain. These nodes can use Linux OS (optimized Ubuntu distribution) or Windows Server 2019. The Azure platform automatically applies OS security patches to Linux nodes on a nightly basis, but on Windows nodes, Windows Update does not automatically run or apply the latest updates. This means

that if you have Windows nodes you need to maintain the schedule around the update lifecycle and enforce those updates.

From the network perspective, these nodes are deployed into a private virtual network subnet with no public IP addresses assigned to it. SSH is enabled by default and should only be used for troubleshooting purpose because it is only available using the internal IP address. In [Figure 2-69](#), you also have an NSG, which can also be used to enhance the network protection.

AKS nodes use Azure Managed Disks and the data is automatically encrypted at rest within the Azure platform. To fulfill the security principle of availability, these disks are also securely replicated within the Azure datacenter.

Important Planning AKS

When you are planning AKS high availability, also take into consideration the process of upgrading an AKS Cluster. Read this article for more information about the upgrade process: <https://docs.microsoft.com/en-us/azure/aks/upgrade-cluster>.

The diagram shown in [Figure 2-69](#) shows the Kubernetes secret element, which is used to inject sensitive data into pods, such as credentials or keys. The use of secrets reduces the sensitive information that is defined in the pod or service YAML manifest. You can read more about secrets in Kubernetes at <https://kubernetes.io/docs/concepts/configuration/secret>.

Security Center for AKS

In addition to the native capabilities in Kubernetes and Azure that were described previously, you can enhance the security posture of your AKS deployment by leveraging Azure Security Center recommendations. In

Security Center, you will see recommendations such as the one shown in [Figure 2-70](#).

The screenshot shows the Azure Security Center (ASC) preview interface for a Kubernetes service named 'asc-preview'. At the top, it displays 'Total recommendations' as 3, with a breakdown: High (2), Medium (0), and Low (1). Below this, under 'Kubernetes service information', details are provided: Resource Name (asc-preview), Resource Group (asc-preview-rg), and Subscription (ASC DEMO). A section titled 'Recommendation list' shows three recommendations, all marked as High priority:

Recommendation	Status
Authorized IP ranges should be defined on Kubernetes Services	High
Audit diagnostic setting	Low
Pod Security Policies should be defined on Kubernetes Services (Preview)	High

Figure 2-70 Security Center recommendations for AKS

Security Center constantly monitors the AKS and Docker configurations and then generates security recommendations that reflect industry standards. In addition to that, if you upgrade Security Center to Standard Tier and enable the AKS threat detection, Security Center will continuously analyze raw security events, such as network data and process creation and the Kubernetes audit log.

Based on this information, Security Center might alert you for threats and malicious activity detected at the host and AKS cluster level. [Figure 2-71](#) shows an example of an alert that notifies you about a configuration that can lead to attackers using a namespace to hide malicious components.

PREVIEW - New container in the kube-system namespace detected
ASC

[Learn more](#)

General information

DESCRIPTION	Kubernetes audit log analysis detected a new container in the kube-system namespace that isn't among the containers that normally run in this namespace. The kube-system namespaces should not contain user resources. Attackers can use this namespace for hiding malicious components.
ACTIVITY TIME	Monday, March 2, 2020, 8:55:52 AM
SEVERITY	● Low
STATE	Active
ATTACKED RESOURCE	ASC
SUBSCRIPTION	ASC DEMO
DETECTED BY	■ Microsoft
ACTION TAKEN	Detected
ENVIRONMENT	Azure
RESOURCE TYPE	✖ Kubernetes Service
CONTAINER NAME	kube-system-container
CONTAINER IMAGE	nginx
POD NAME	kube-system-container

Figure 2-71 Security Center alert for AKS

Implement vulnerability management

When managing ACR, it is a good practice to implement a vulnerability assessment solution that scans all pushed images. You can leverage Security Center Standard tier with the ACR bundle enabled to have the vulnerability assessment functionality.

When this capability is enabled, Security Center scans the image that was pushed using a Qualys scanner, which is fully integrated with the Security Center standard tier, and there is no additional cost for the Qualys engine.

Figure 2-72 shows a diagram of how vulnerability management for ACR is done using Security Center.

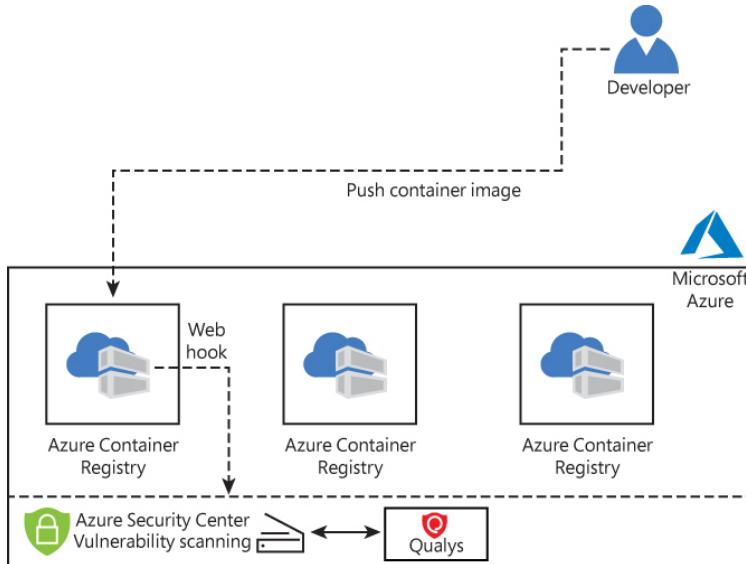


Figure 2-72 Vulnerability scanning process in Security Center

If an issue is found during this scanning process, Security Center generates an actionable recommendation with guidance for remediating the issue. [Figure 2-73](#) shows an example of the type of Security Center recommendations you might see.

Priority	Count
High	1
Medium	0
Low	2

Recommendation list

- Audit diagnostic setting (Low)
- [Preview]: Container Registry should use a virtual network ... (Low)
- Vulnerabilities in Azure Container Registry images should ... (High)

Figure 2-73 An ACR Security Center recommendation

Configure isolation for AKS

Container isolation is applicable for scenarios that you need to isolate workloads or teams. AKS provides capabilities for multitenant clusters and resource isolation. Natively, Kubernetes already creates a logical isolation boundary by using a namespace, which is the logical group of resources (such as pods).

Also, the following Kubernetes features should be used in scenarios that require isolation and multitenancy:

- **Scheduling** The AKS scheduler allows you to control the distribution of compute resources and to limit the impact of maintenance events. This component includes the use of features such as resource quotas and pod-disruption budgets.
- **Networking** AKS networking enables you to leverage the network policy's capability to allow or deny traffic flow to pods.
- **Authentication and authorization** As mentioned earlier in the chapter, the use of RBAC and Azure AD integration are imperative to enhance the security of your authentication and authorization.
- **Other features** These features include pod-security policies, pod-security contexts, scanning images, and runtimes for vulnerabilities.

Important Least Privilege

An important design consideration when planning your AKS is to provide the least number of privileges that are scoped to the resources each team needs.

There are two main types of isolation for AKS clusters: logical and physical. You should use logical isolation to separate teams and projects. Using logical isolation, a single AKS cluster can be used for multiple workloads, teams, or environments.

It is also recommended that you minimize the number of physical AKS clusters you deploy to isolate teams or applications. [Figure 2-74](#) shows an example of this logical isolation.

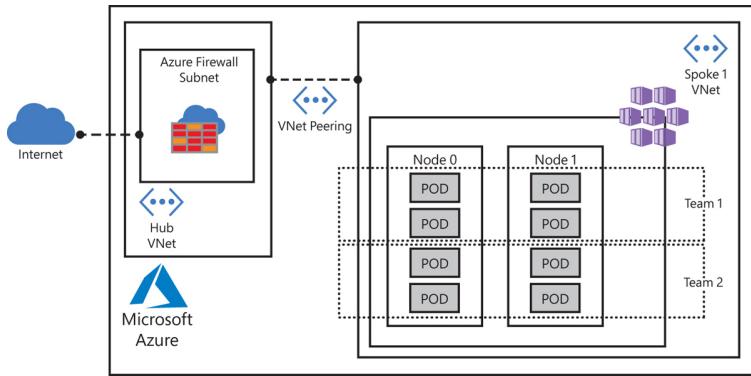


Figure 2-74 AKS logical isolation

Logical isolation can help minimize costs by enabling autoscaling and run only the number of nodes required at a time.

Physical isolation is usually selected when you have a hostile multitenant environment where you want to fully prevent one tenant from affecting the security and service of another. The physical isolation means that you need to physically separate AKS clusters. In this isolation model, teams or workloads are assigned their own AKS clusters. While this approach usually looks easier to isolate, it adds additional management and financial overhead.

Configure security for container registry

Azure Container Registry (ACR) is a private registry of Docker and Open Container Initiative (OCI) images, based on open-source Docker Registry 2.0. Developers can pull (download) images from an Azure container registry, and they can also push (upload) to a container registry as part of a container development workflow. ACR pricing tiers are

- **Basic** More suitable for developers that are learning about ACR
- **Standard** Increased storage and image throughput and more suitable for a production environment
- **Premium** More suitable for high-volume scenarios and high image throughput



Exam Tip

On the exam, you might need to select the best pricing tier (also known as a SKU) according to the given scenario.

You can use an Azure AD service principal to provide container image *docker push* and *pull* access to your container registry. Azure AD service principals provide access to Azure resources within your subscription.

Think of a service principal as a user identity for a service.

ACR also supports built-in Azure roles to provide different levels of permissions to an Azure container registry. For example, you can use role assignments to grant access AKS access to ACR. The built-in ACR roles are

- **Owner** Can access Resource Manager, create and delete registry, push and pull images, delete image data, and change policies
- **Contributor** Can do the same operations as the owner
- **Reader** Can access Resource Manager and pull images
- **ArcPush** Can push and pull images
- **ArcPull** Can pull an image
- **ArcDelete** Can delete image data
- **ArcImageSigner** Can sign images

To pull or push images to an Azure container registry, a client must interact over HTTPS with two different endpoints: the Registry REST API endpoint and the storage endpoint. By default, an ACR accepts connections over the Internet from hosts on any network. If you are using ACR Premium, you can leverage Azure VNet network access rules to control access to your ACR.

Implement Azure disk encryption

Data encryption at rest is an extremely important part of your overall VM security strategy. Security Center will even trigger a security recommendation when a VM is missing disk encryption. You can encrypt your Windows and Linux virtual machines' disks using Azure Disk Encryption (ADE). For Windows OS, you need Windows 8 or later (for client) and Windows Server 2008 R2 or later (for servers).

ADE provides operating system and data disk encryption. For Windows, it uses BitLocker Device Encryption; for Linux, it uses the DM-Crypt system. ADE is not available in the following scenarios:

- Basic A-series VMs
- VMs with less than 2 GB of memory
- Generation 2 VMs and Lsv2-series VMs
- Unmounted volumes

ADE requires that your Windows VM has connectivity with Azure AD to get a token to connect with Key Vault. At that point, the VM needs access to the Key Vault endpoint to write the encryption keys, and the VM also needs access to an Azure storage endpoint. This storage endpoint will host the Azure extension repository as well as the Azure storage account that hosts the VHD files.

Important URL Filtering

If the VM is hardened and there are Internet access restrictions, make sure that this VM can at least access the URI. See <http://aka.ms/az500kvfw>.

Group policy is another important consideration when implementing ADE. If the VMs for which you are implementing ADE are domain joined, make sure to not push any group policy that enforces Trusted Platform Module (TPM) protectors. In this case, you will need to make sure that the **Allow BitLocker Without A Compatible TPM** policy is configured. Also, BitLocker policy for domain-joined VMs with custom group policy

must include the following setting: Configure User Storage Of BitLocker Recovery Information / Allow 256-Bit Recovery Key.

Because ADE uses Azure Key Vault to control and manage disk encryption keys and secrets, you need to make sure Azure Key Vault has the proper configuration for this implementation. One important consideration when configuring your Azure Key Vault for ADE is that they (VM and Key Vault) both need to be part of the same subscription. Also, make sure that encryption secrets are not crossing regional boundaries; ADE requires that the Key Vault and the VMs are colocated in the same region. When configuring your Azure Key Vault, use `Set-AzKeyVaultAccessPolicy` with `-EnabledForDiskEncryption` to allow Azure platform to access the encryption keys or secrets in your key vault, as shown here:

[Click here to view code image](#)

```
Set-AzKeyVaultAccessPolicy -VaultName "<your -  
keyvault-name>" -ResourceGroupName  
"MyResourceGroup" -EnabledForDiskEncryption
```

While these are the main considerations to encrypt Windows VM, Linux VMs have some additional requirements. When you need to encrypt both data and OS volumes where the root (/) file system usage is 4 GB or less, you will need to have at least 8 GB of memory. However, if you need to encrypt only the data volume, the requirement drops to 2 GB of memory. The requirement doubles if Linux systems are using a root (/) file system greater than 4 GB, which means that the minimum memory requirement is `root file system usage * 2`.

More Info Supported Linux Distributions

To see the list of supported Linux distributions for ADE implementation, visit <http://aka.ms/az500ADElinux>.



Exam Tip

Understanding those considerations prior to implementing ADE is very important, mainly when reading a scenario in the AZ-500 exam. The scenario description will give you the requirements and the constraints, which means that in some scenarios, it won't be possible to implement ADE unless some other task is executed prior to the ADE implementation.

Assuming that you have the right prerequisites in place to implement ADE, you can use the `Set-AzVmDiskEncryptionExtension` PowerShell cmdlet to implement the encryption in a VM, as shown in the following example:

[Click here to view code image](#)

```
$AKeyVault = Get-AzKeyVault -VaultName MyAKV -  
ResourceGroupName MyRG  
Set-AzVmDiskEncryptionExtension -  
ResourceGroupName MyRG -VMName MyVM  
-DiskEncryptionKeyVaultUrl $AKeyVault.VaultUri -  
DiskEncryptionKeyId $AKeyVault.  
ResourceId
```

Wait a few minutes, and the output will show the field `.IsSuccessStatusCode` as `True`, and the `Statuscode` as `OK`. You can also check the encryption status using `Get-AzVmDiskEncryptionStatus` cmdlet. If it was encrypted successfully you should see a result similar to this:

[Click here to view code image](#)

<code>OsVolumeEncrypted</code>	: Encrypted
<code>DataVolumesEncrypted</code>	: NoDiskFound

```
OsVolumeEncryptionSettings :  
Microsoft.Azure.Management.Compute.Models.  
DiskEncryptionSettings  
ProgressMessage : Provisioning  
succeeded
```

More Info Disk Encryption

For more disk encryption scenarios for Windows VM, see
<http://aka.ms/az500ADEWin>.

Configure security for Azure App Service

Azure App Service is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends.

Azure App Service Environment (ASE) is an Azure App Service feature that provides an isolated and dedicated environment for securely running App Service apps in the cloud. You can create multiple ASEs to host multiple apps running in Windows, Linux, Docker, mobile, and function apps.

Important Pricing Tier

All pricing tiers run your apps on the shared network infrastructure in the Azure App Service, except for the Isolated pricing tier, which gives you complete network isolation by running your apps inside a dedicated App Service environment.

To configure security for Azure App Service, you need to understand the variety of options available. Azure App Service has built-in security controls that can be leveraged to enhance the overall security posture of your apps. Essentially, some of these controls are Azure components that were described throughout this chapter. **Table 2-3** provides a summary of the security controls that can be used with Azure App Service.

Table 2-3 Advantages and limitations

Layer	Security control	Description

Netw or k	Service Endpoint	You can use access restrictions to define a priority-ordered allow/deny list that controls network access to your app. This is an important practice to limit exposure to inbound network traffic.
	VNet injection support	This security control is used for ASE, which is a private implementation of App Service dedicated to a single customer and injected into that customer's VNet.
	Network Isolation and Firewalling support	You can configure network access control list (ACL) to lock down allowed inbound traffic.
	Forced tunneling support	Although ASE outbound dependency traffic must go through the VIP that is provisioned with the ASE, you can configure it to customize the network routing.
Moni to ri ng	Azure monitoring support	You can review quotas and metrics for an app and the App Service plan. You can also configure alerts and autoscale rules-based metrics.
	Control and management plane logging and audit	Because all management operations performed on App Service objects occur via Azure Resource Manager (ARM), you will be able to see historical logs of these operations. Keep in mind that there is no data-plane logging and auditing available for App Service.
Identit y	Authentication	Supports integration with Azure AD and other OAuth providers.
	Authorization	Controlled by Azure AD and RBAC.
Data a P ro	Server-side encryption at rest:	The App Service site file content is stored in Azure Storage, which automatically encrypts the content at rest, and customer's supplied secrets are encrypted at rest.

te ct io n	Microsof t- manage d keys	
	Server- side encrypti on at rest: custome r- manage d keys (BYOK)	Supports the storage of an application's secret in Key Vault, so that it can be retrieved during runtime.
	Encrypti on in transit	Supports the use of HTTPS for inbound traffic.
	API calls encrypte d	Also supported via calls over HTTPS.
C o nf ig u ra ti o n m a n a g e m e nt	Configur ation manag ement support	The state of an App Service configuration can be exported as an ARM template.

Besides the available security controls that are inherited from Azure, you should also ensure that you are always developing your apps using the latest versions of supported platforms, programming languages, protocols,

and frameworks. It is very important that throughout the development lifecycle, you properly configure the authentication for these apps. Always make sure that authentication is required and that anonymous access is disabled unless the scenario's description clearly states that it must be enabled. You can also enhance your authentication security by requiring clients to use a certificate to authenticate. This practice improves security by allowing connections only from clients that can authenticate using certificates that you provide.

As part of your secure configuration of App Service, make sure that data in transit is protected, which means that you should always redirect HTTP to HTTPS traffic and that you enforce the latest version of the TLS protocol. Communications from your Azure App Service and other Azure resources, such as Azure Storage, should also be encrypted. If the scenario description requires you to transfer files from your Azure app for another location using FTP, make sure that you are utilizing FTPS instead.

Some of the overall security recommendations for Azure app Service will also be surfaced in Azure Security Center, as shown in [Figure 2-75](#).

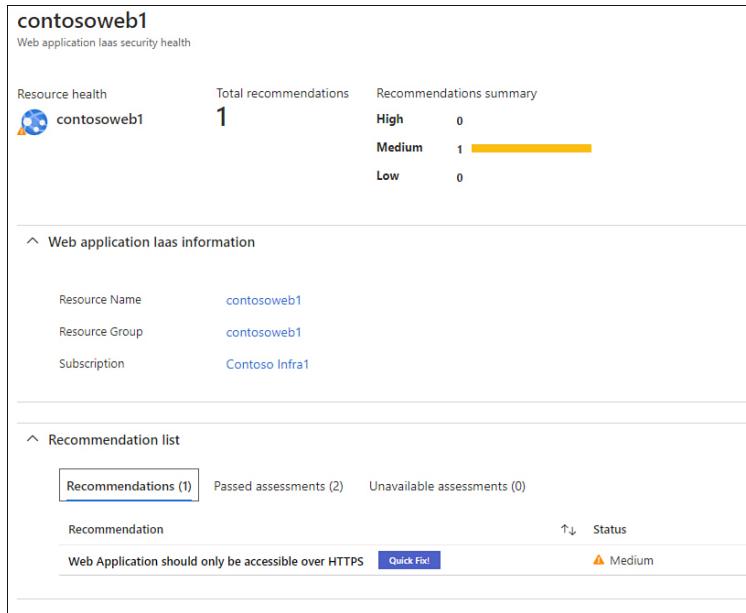


Figure 2-75 Security Center recommendations for App Service

Security Center will perform this security assessment on your apps, which is part of the Azure Security Center Free tier. However, if you upgrade to Standard tier, you will also get threat detection for App Service. Security Center’s App Service threat detection includes analytics and machine-learning models that cover all interfaces that allow customers to interact with their applications—whether it’s over HTTP or through one of the management methods.

Configure SSL/TLS certificates

To ensure that you are always protecting the data in transit, you should configure your App Service to use an SSL/TLS certificate. To create a TLS bind of your certificate to your app or enable client certificates for your App Service app, your App Service plan must be configured to the Basic, Standard, Premium, or Isolated tiers.

The App Service enable different scenarios to handle certificates, which includes the capability to buy a certificate; import an existing certificate from the App Service; upload an existing certificate that you might

have already; to import a certificate from Key Vault (from any subscription on the same tenant); or to create a free App Service custom certificate. (This last option does not provide support for naked domains.)

With the exception of buying a certificate—which is available via the **Buy Certificate** button—all other options are surfaced under the **Private Key Certificates (.pfx)** tab in the **TLS/SSL Settings** option in the right-hand navigation pane of the App Service that you selected. [Figure 2-76](#) shows an example of this tab.

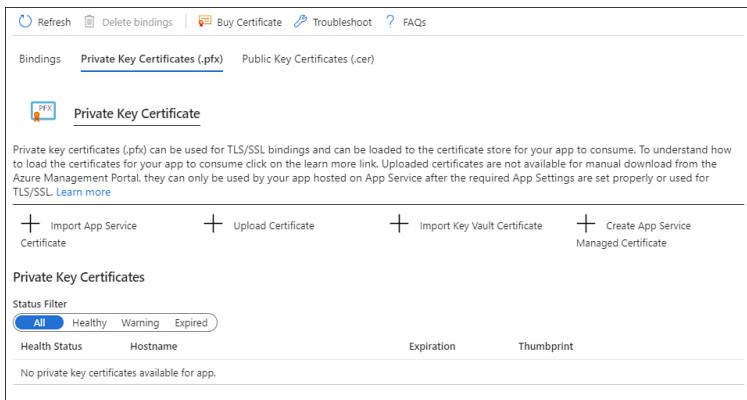


Figure 2-76 Options to configure a private key certificate for App Service

For the purpose of the AZ-500 exam, the scenario description is what leads you to choose one option over the other. For example, let's say that a Contoso administrator needs to secure data in transit for his or her App Service, but the administrator needs to save costs, leverage the existing Public Key Infrastructure (PKI) on-premises, and support naked domains. In this case, the most appropriate option would be to upload an existing certificate. This will save costs because it will leverage the existing PKI (which already met the second requirement) and it supports naked domains. When uploading an existing certificate, make sure you have the password for the protected PFX file; the private key must be at least 2048 bits long, and it must contain all intermediate certificates in the certificate chain.

Another important scenario is when you need to respond to requests to a specific hostname over HTTPS. In this case, you need to secure a custom domain in a TLS binding. In this scenario, you would use the **Add TLS/SSL Binding** option, which is available in the **Bindings** tab, as shown in Figure 2-77.

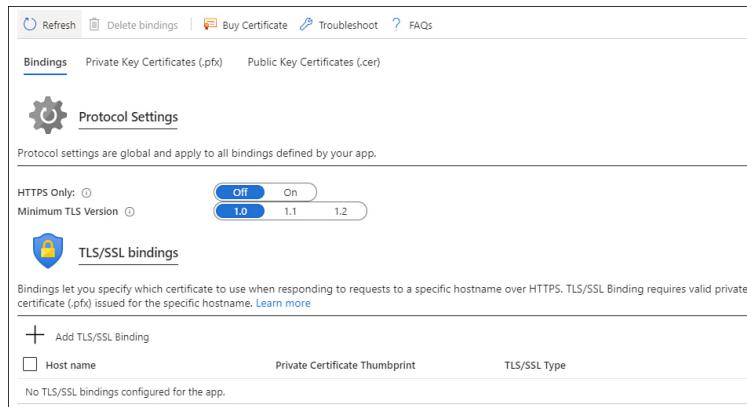


Figure 2-77 Options to add a TLS/SSL binding

The certificate that will be used to bind TLS/SSL needs to contain an `ExtendedKeyUsage` for server authentication object identifier (OID), which is `1.3.6.1.5.5.7.3.1`, and it must be signed by a trusted certificate authority. Also, notice that on this page, you can also configure your App Service to only answer to HTTPS, and you can configure the TLS version that will be used.

Tip Certificates

For the detailed steps to configure the different types of certificates, see <https://aka.ms/az500AppCertificates>.

Configure authentication

By default, authentication and authorization are disabled. Upon enabling it, every incoming HTTP request passes through it before being handled by your application code. The authentication and authorization module runs separately from your application code and is configured using app settings.

The authentication and authorization modules are responsible for handling the authentication of users based on the selected provider, and it validates, stores, and refreshes tokens. They also manage the authenticated session and injects identity information into request headers. To configure authentication in App Service, you need to switch the **App Service Authentication** toggle to **ON**, and under **Authentication / Authorization**, the authentication options will appear, as shown in [Figure 2-78](#).

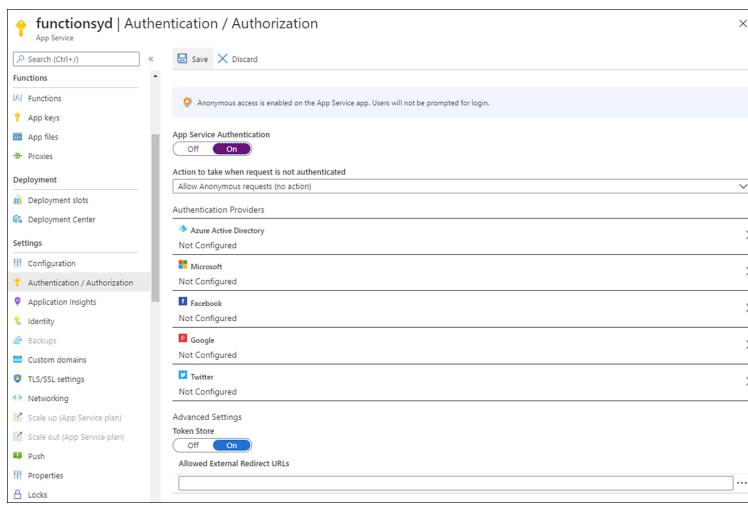


Figure 2-78 Authentication and authorization options

Because App Service uses federated identity, in which a third-party identity provider manages the user identities and authentication flow, the next step is to configure type of authentication provider that will answer to requests that are not authenticated. Click the **Action To Take When Request Is Not Authenticated** drop-down menu and select the appropriate option. The option that you selected in the drop-down menu should match with the provider that you select in the **Authentication Providers** section. Once you select the appropriate provider, its sign-in endpoint is available for user authentication and for validation of authentication tokens from the selected provider.

Tip End-to-End Authentication and Authorization

For an example of how to authenticate and authorize users end to end in Azure App Service, see <http://aka.ms/az500AppServiceAuth>.

If you select the **Allow Anonymous Requests (No Action)** option in the drop-down menu, this option will defer authorization of unauthenticated traffic to your application code; in other words, you need to write the authentication code in your app. If it is an authenticated request, App Service will pass along authentication information in the HTTP headers. Table 2-4 shows a summary of each identity provider:

Table 2-4 App Service identity providers

Identity provider	Sign-in endpoint	Configuration requirements
Azure AD	/.auth/login/aad	You can create a new Azure AD App or use an existing one. Allows you to enable Common Data Services (CDS) Permissions.
Microsoft Account	/.auth/login/microsoftaccount	Requires the Client ID and Client Secret. You can select different scopes that are responsible for enabling different operations.
Facebook	/.auth/login/facebook	Requires the App ID and App Secret. You can select different scopes that are responsible for enabling different operations.
Google	/.auth/login/google	Requires a Client ID and a Client Secret.
Twitter	/.auth/login/twitter	Requires an API key and an API secret.

Important Common Data Service (CDS)

Common Data Service (CDS) enables you to securely store and manage data that's used by your apps. Standard and custom entities within CDS provide a secure and cloud-based storage option for your data. For more information about CDS, see <https://docs.microsoft.com/en-us/powerapps/maker/common-data-service/data-platform-intro>.

If Contoso administrator's requirement is to securely store and manage the data that is used by the company's app, Azure AD is the identity provider that addresses this requirement because Azure AD allows you to use CDS.

Automatic updates

Since App Service is a Platform as a Service (PaaS), the operating system (OS) and application stack are managed for you by Azure, which means you don't need to worry about software update. Azure manages OS patching on two levels: the physical servers and the guest VMs that run the App Service resources. Both will follow the regular Microsoft Patch Tuesday update cycle, which is once a month, unless it is a zero day patch, which will be handled with higher priority and probably out of band (outside the regular Patch Tuesday cycle). When a new major or minor version is added to App Service, it is installed side by side with the existing versions.

App Service preserves its Service Level Agreement (SLA) even during the patch updates, which means that even if a patch requires a VM to restart, it will not affect App Service production because there always will be buffer in capacity.

Access to patches in the registry at

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Component Based Servicing\Packages are locked down, though basic info regarding OS and runtime updates can be queried using Kudu Console at
<https://github.com/projectkudu/kudu/wiki/Kudu->

console. For example, if you want to see the Windows version, you can access this URL:

<https://<appname>.scm.azurewebsites.net/Env.cshtml>.



Thought experiment

In this thought experiment, demonstrate your skills and knowledge of the topics covered in this chapter. You can find answers to this thought experiment in the next section.

Advanced security for compute at Tailwind Traders

You are one of the Azure administrators for Tailwind Traders, an online general store that specializes in a variety of products for the home. Tailwind Traders is deploying new VMs in Azure to increase the compute capacity because the company is forecasting an increase in online store shopping during the upcoming holiday season. Before releasing those VMs for use, they need to ensure that these VMs are configured to use security best practices, which include secure configurations, endpoint protection installation, and ensuring that the operating system is fully up to date.

Currently, Tailwind Traders does not have any cloud security posture management in place, but the company is interested in trying the Azure Security Center Free tier. To improve security, they also need continuously monitor those servers to identify potential attacks, and they want to receive an alert in case there are suspicious activities or indications of an attack against those

servers. Another goal of Tailwind Traders is to allow the Security Operation Center (SOC) analysts to have read-only access to the Security Center dashboard in order to view the alerts. With this information in mind, answer the following questions:

- 1.** Will Azure Security Center Free tier meet those requirements?
- 2.** What Azure role should the SOC analysts have to accomplish their goals?
- 3.** Where in Azure Security Center should the administrator go to identify whether the servers have an endpoint protection solution installed?

THOUGHT EXPERIMENT ANSWERS

This section contains the solution to the thought experiment. Each answer explains why the answer choice is correct.

- 1.** Azure Security Center Free tier will only accomplish partial results of the desired requirements. It will enable the administrator to see security recommendations and improve the security posture of the workloads, but to have continuous monitoring of threat detection, the administrator needs to upgrade Security Center to Standard tier.
- 2.** You should assign Security Reader role to the SOC analysts.
- 3.** To identify whether the servers have an endpoint protection solution installed, you should go to the **Recommendations** dashboard in Azure Security Center.

CHAPTER SUMMARY

- There are different types of Azure VPNs that will be selected according to the organization's requirement, including site-to-site VPN, point-to-site VPN, VNet-to-VNet, and multi-site VPN.
- Consider using ExpressRoute if your connectivity scenario requires a higher level of reliability, faster speeds, consistent latencies, and higher security than typical Internet connections.
- Network security group (NSG) in Azure allows you to filter network traffic by creating rules.

- Consider using Azure Firewall when your organization requires a fully stateful firewall, centralized management, with network- and application-level protection.
- Consider using Azure Front Door when your organization requirements include Azure deployment across different regions with a high-performance experience for applications and that it is resilient to failures.
- Use Azure Bastion to enable secure access via TLS to Internet users who need to access resources that are located in your Azure network.
- When you need resource-level filtering to enhance the security of your workloads, make sure to use a resource-level firewall.
- Enable Azure DDoS Standard when you need to tune application traffic volume and you want to ensure an SLA level that provides application guarantee and cost protection.
- To receive threat alerts in Azure Security Center, you need to upgrade to the Standard tier.
- You can use Azure Security Center to monitor the security posture of Azure Kubernetes and Azure Container registry.
- Azure Disk Encryption requires that your Windows VM has connectivity with Azure AD to get a token to connect with Key Vault.
- To ensure that you are always protecting the data in transit, you should configure your App Service to use an SSL/TLS certificate.

Chapter 3

Manage security operations

The main goal of security operations is to maintain and restore the security assurances of the systems as adversaries attack them. The National Institute of Standards and Technology (NIST) describes the tasks of security operations in their Cybersecurity Framework, which are Detect, Respond, and Recover. To be able to execute those functions in a cloud environment, you not only need the correct approach, but you also need to understand how the native tools work to provide you the data you need to limit the time and access an attacker can get to valuable systems and data.

Azure has native capabilities that you can leverage to continuously monitoring the security operations of your environment continuously, so you can quickly identify potential threats to your workloads.

Skills in this chapter:

- Skill 3.1: Monitor security by using Azure Monitor
- Skill 3.2: Monitor security by using Azure Security Center
- Skill 3.3: Monitor security by using Azure Sentinel
- Skill 3.4: Configure security policies

SKILL 3.1: CONFIGURE SECURITY SERVICES

Security operations start by ensuring that you have visibility and access to the underlying logs of the different services that you want to monitor. Azure Monitor can collect and store data from Azure applications, operating systems, Azure resources, Azure subscriptions, Azure tenant, and custom resources. This

section of the chapter covers the skills necessary to configure security services, which is based on Azure Monitor, according to the Exam AZ-500 outline.

Configure Azure Monitor

One common question when it comes to the usage of Azure Monitor is, “How do I enable it?” Azure Monitor is automatically enabled when you create a new Azure subscription. At that point, activity log and platform metrics are automatically collected. The other common question is, “Can Azure Monitor also monitor resources that are on-premises?” Although Azure Monitor implies (by the name) that the resources are in Azure, it also collects data from virtual machines and applications in other clouds and on-premises to monitor.

For this reason, before making any sort of configuration in Azure Monitor, is important to understand some foundational concepts of this platform. The section that follows covers some key principles.

Reviewing Azure Monitor concepts

The diagram shown in [Figure 3-1](#) helps you better understand the breadth of Azure Monitor and the different areas that it touches.

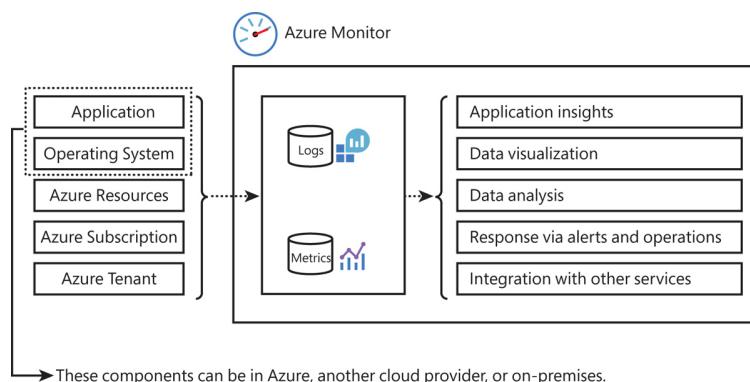


Figure 3-1 Architecture diagram of the Azure Monitor solution

On the left side of the diagram shown in [Figure 3-1](#), you have the different layers that represent the components that will generate logs, which can be ingested by Azure Monitor. From the application and operating system perspective, the machine can be physically located on-premises, in Azure or in another cloud provider. Aside from these data sources, you can also ingest data from different Azure resources, subscriptions, and the Azure tenant itself. This data is ingested into the Log Analytics Workspace, which is part of the Azure Monitor solution and once the data is there, you can query it using Kusto Query Language, which uses schema entities that are organized in a hierarchy similar to SQL's databases, tables, and columns.

The last three layers that appear in the left side of the diagram shown in [Figure 3-1](#) represent the three major layers in Azure where you can obtain logging information. The definition of each layer is shown here:

- **Azure Resources** Here, you will be able to obtain resource logs, which has operations that were executed in the data plane level of Azure. An example of that would be getting a secret from Azure Key Vault. These logs are also referred to as diagnostic logs.
- **Azure Subscription** Here, you will be able to obtain activity logs, which has operations that were executed in the management plane. You should review these logs when you need to determine the answer for the *what* (what operation was made), *who* (who made this operation), and *when* (when this operation was made). For example, if a VM was deleted, you should go to Azure Activity Log to find out the answer of the *what*, *who*, and *when* regarding the delete VM operation.
- **Azure Tenant** Here, you will be able to obtain the Azure Active Directory logs. In this layer you have the history of sign-in activity and audit trail of changes made in the Azure Active Directory.

Is very important to understand those layers when studying to the AZ-500 exam because you may have scenarios where you will need to select the right option regarding where to look for a specific information. For example, the Contoso administrator wants to identify the user who stopped the virtual machine two weeks ago; where he should be searching for this information? If you

answered Azure Activity Log, you are correct. As mentioned before, in Activity Log you will find management plane operations and the identification of the *what*, *who*, and *when* an operation was performed.

Metrics are another type of information that can be ingested. Metrics are numerical values that describe some aspect of a system at a particular point in time. Telemetry, such as events and traces, and performance data are stored as logs so that it can all be combined for analysis. This type of information can be used during scenarios where you need to collect security-related performance counters from multiple VMs and create alerts based on certain thresholds.

Because Azure Monitor starts collecting data from a resource upon the creation of that resource, it is important to know where to look when you need information about those resources. Many resources will have a summary of performance data that are relevant for that resource, and this is usually located in the **Overview** page of that resource. For example, in the **Overview** option of an Azure storage account, you will see insights regarding the average latency, egress data, and requests, as shown in [Figure 3-2](#).

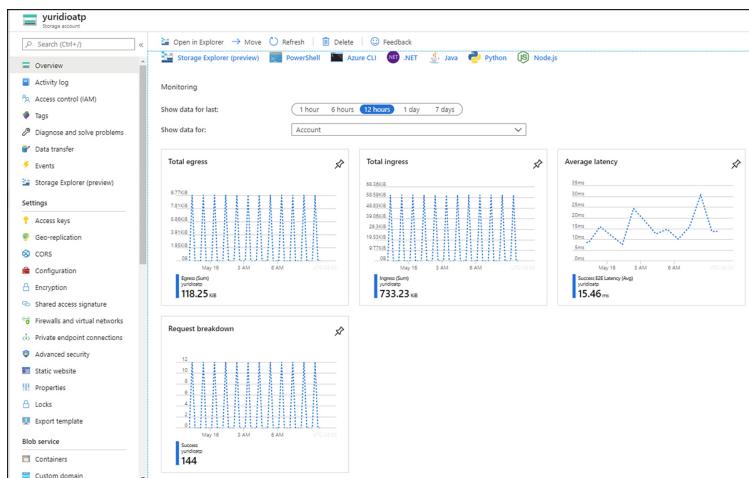


Figure 3-2 Summary of storage account performance insights

If you need to query logs that have operations that were executed in the management plane, you should use the Azure Activity Log. To access the Activity Log, follow these steps:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **activity** and under **Services**, click **Activity Log**. The **Activity Log** page appears, as shown in Figure 3-3.

The screenshot shows the 'Activity log' page in the Azure portal. At the top, there are buttons for 'Edit columns', 'Refresh', 'Diagnostics settings', 'Download as CSV', 'Logs', 'Pin current filters', and 'Reset filters'. Below these are search and quick insights fields. Filter options include 'Management Group: None', 'Subscription: Visual Studio Ultimate with MSDN', 'Timespan: Last 6 hours', and 'Event severity: All'. A 'Add Filter' button is also present. The main area displays a table with columns: Operation name, Status, Time, Time stamp, Subscription, and Event initiated by. A message '0 items.' and 'No results to display' is centered below the table.

Figure 3-3 Activity log initial page

3. Here, you can use the **Timespan** filter to adjust the timeline that you want to perform your query. For this example, this filter was changed for the last one hour, and after applying the change, the result appears, as shown in Figure 3-4.

The screenshot shows the 'Activity log' page after applying a filter for the last hour. The 'Timespan' dropdown now shows 'Last 1 hour'. The main area displays a table with one item: 'Delete Storage Account' with status 'Succeeded', timestamp '4 minutes ago', subscription 'Visual Studio Ultimate with MSDN', and initiator 'yundigories@hotmail.com'.

Figure 3-4 Activity log results after filtering

4. The result shows a summary of the operation, including the time, status, time stamp and who initiated the event. If you want more detailed information about the operation, you can expand the operation name field and click on it. There you will have the details of the operation in the JSON tab.

As mentioned in the previous section of this chapter, the other type of data that you may want to use is metric. If you are monitoring a virtual machine and you need more metrics beyond of the ones that appear in the **Overview** page, you can go to the **Metrics** page and from there customize the metrics that you want to monitor as shown in the example of Figure 3-5.

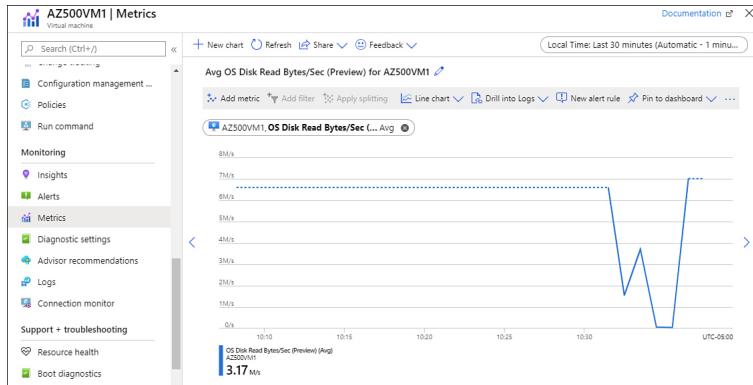


Figure 3-5 Visualizing VM metrics

Create and customize alerts

Another important feature in Azure Monitor is the capability to create alerts for different types of events. You can use the following type of data to generate alerts with the data that was collected for the past 30 days (by default):

- Metric values
- Log search queries
- Activity log events
- Health of the underlying Azure platform
- Tests for website availability

In Figure 3-5, you can see an option right above the New Alert Rule chart. This option enables you to go from this dashboard directly to the **Alert** dashboard and create an alert rule using the metric that is currently shown on screen, which, in this case, is to monitor OS Disk Read Bytes/Sec, as shown in Figure 3-6.

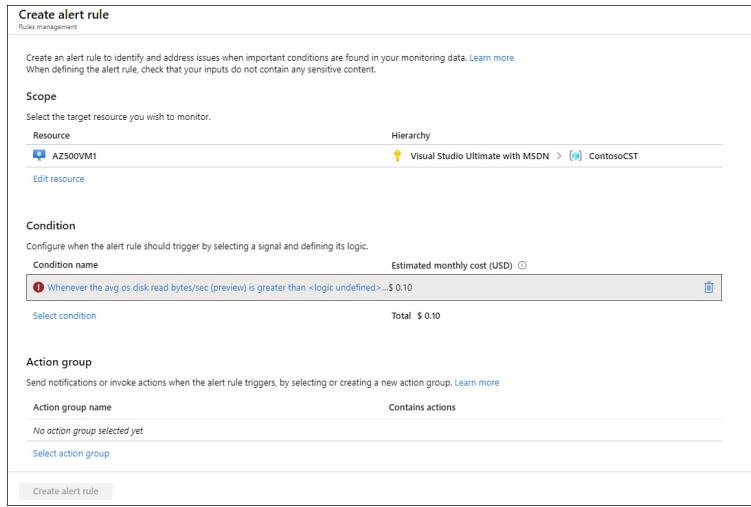


Figure 3-6 Creating an alert rule

The **Create Alert Rule** page has some important parameters that must be filled, but when you activate this page from the **Metrics** page where you already configured the metrics that you want to monitor, this page prepopulates the **Scope** (which is the target resource that you want to monitor) and the **Condition** (which is the rule logic that will be used to trigger the alert). While the scope has the resource that you want to monitor, the condition might need some adjustments according to your needs. To customize the condition, just click the condition name and the **Configure Signal Logic** blade appears, as shown in Figure 3-7.

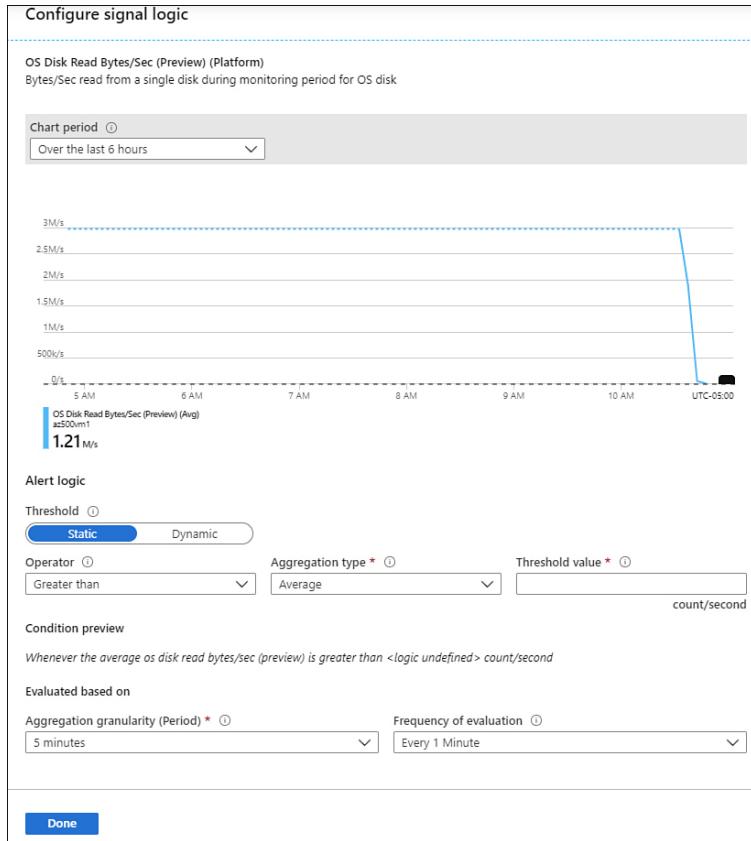


Figure 3-7 Customizing the alert logic

The first part of this blade has the performance counter name that you are using for this rule and a sample chart with data over the last 6 hours. The second part of this blade is where you configure the threshold. In the **Alert Logic** section, you can change the toggle to be **Static** (you provide a specific value as threshold) or **Dynamic** (which uses machine learning to continuously learn about the behavior pattern). In this case, the Contoso administrator wants to receive an alert if the average OS Disk Read Bytes/Sec counter is higher than 3 MB, which means **Static** is the best option to use. In this case, the operator remains `greater than`, the **Aggregation Type** remains `average`, and you just need to enter the value (in this case, 3) in the **Threshold Value** field. The **Condition Preview** section explains the logic so you can confirm that this is what you want to do. The **Evaluated Based On** section is where you can configure the **Aggregation Granularity (Period)**

option, which defines the interval over which the datapoints are grouped. You can also configure the **Frequency Evaluation**, which defines how often this alert rule should be executed. The **Frequency Evaluation** should be the same as the **Aggregation Granularity** or higher. Once you finish, click the **Done** button.

Next, configure the **Action Group** section, which allows you to configure the type of notification that you want to receive. To configure this option, click **Select Action Group**, and in the **Select An Action Group To Attach To This Alert Rule** blade, click the **Create Action Group** option; the **Add Action Group** blade appears, as shown in Figure 3-8.

The screenshot shows the 'Add action group' configuration dialog box. It includes fields for 'Action group name' (with a placeholder 'My first alert'), 'Short name' (placeholder 'Alert'), 'Subscription' (selected 'Visual Studio Ultimate with MSDN'), 'Resource group' (selected 'Default-ActivityLogAlerts'), and an 'Actions' section with a table for adding actions. The table has columns for 'Action name' (placeholder 'Unique name for the action') and 'Action Type' (dropdown menu). At the bottom, there's a note about email consistency and a link to 'Azure Privacy Statement' and 'Azure Alerts Pricing'. A blue 'OK' button is at the bottom left.

Figure 3-8 Action group configuration

On this blade, you should start by typing a name for this action group; this can be a long name that helps you identify what this group does. In the **Short Name** field, add a short name, which appears in emails or messages that might be sent by this alert. Select the subscription and resource group to where this action group resides;

under **Action Name**, type a name for the first action. Notice that there are many fields for actions; that's because you can have actions such as sending an email, sending a SMS message, or running a runbook, among others. In his case, the Contoso administrator wants to send an email to a distribution list and send an SMS message to the on-call phone. For the action type, select **Email/SMS Message/Push/Voice**, and the **Email/SMS Message/Push/Voice** blade appears. In this blade, type the email and the SMS number after that, click **OK** and then **OK** again.

To finish the alert creation, you just need to add an **Alert Rule Name** and a brief **Description** and then choose the **Severity** of the alert from the drop-down menu. The severity should represent the level of criticality that you want to assign for this rule. In this case, the Contoso administrator understands that when this threshold is reached, an important (not critical) alert should be raised, which, in this case, could be represented by **Sev 2**, as shown in Figure 3-9.

The screenshot shows the 'Alert rule details' configuration page. It includes fields for 'Alert rule name' (Contoso Disk Monitoring), 'Description' (Specify the alert rule description), 'Severity' (Sev 2), and a checked 'Enable alert rule upon creation' checkbox. A blue 'Create alert rule' button is at the bottom.

Figure 3-9 Configuring the alert rule details

Ideally, you should enable this rule upon creation, which is why the **Enable Alert Rule Upon Creation** check box is selected by default. To commit all the changes, click the **Create Alert Rule** button.

Important Activation Time

Usually, new metric rules take up to 10 minutes to activate.

Once you finish creating the rule, you should receive an email advising you that you were added to the action group. A sample of this email is shown in [Figure 3-10](#).

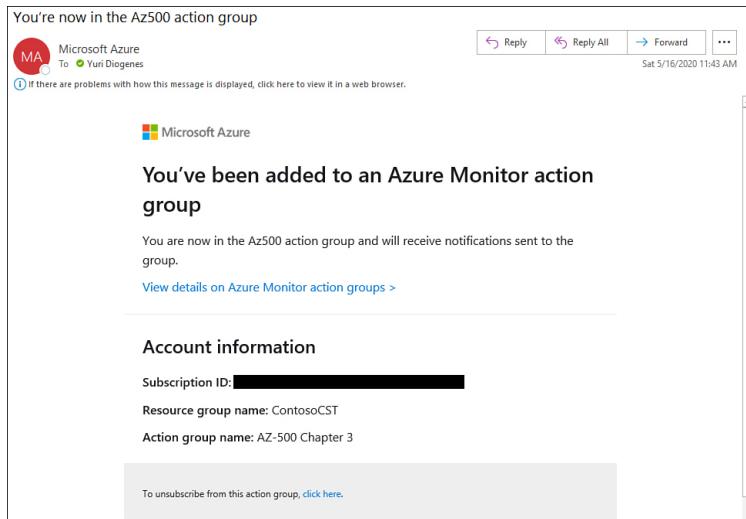


Figure 3-10 Email notification generated by Azure Monitor

You should also receive the SMS message. Notice that the short name that you used appears in the message, as shown in [Figure 3-11](#).

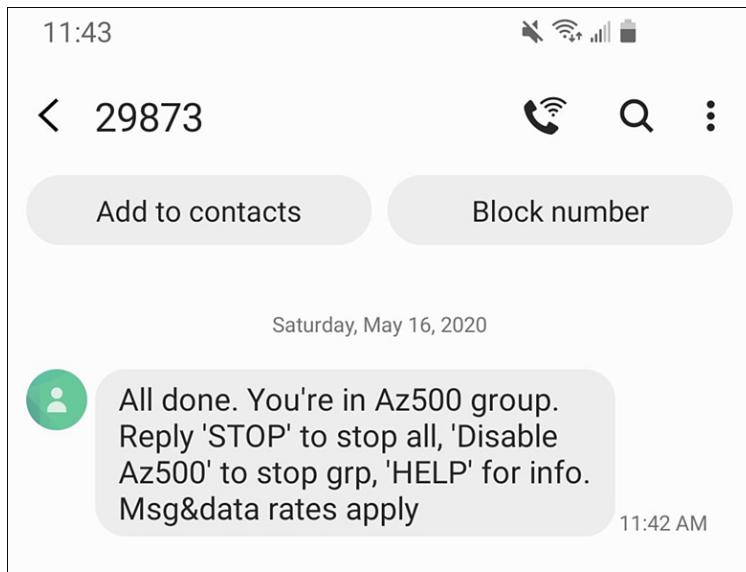


Figure 3-11 SMS notification generated by Azure Monitor

Now that you created an alert based on a metric that you used previously, the question is, “What if I need to change the alert rule?” If you want to be able both to see and change alerts, you can use the **Alerts** dashboard. Follow the steps below to access this dashboard.

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **alert**, and under **Services**, click **Alerts**.
3. Click the **Manage Alert Rules** button, and the **Rules** page appears, as shown in Figure 3-12.

The screenshot shows the 'Rules' page in the Azure portal. At the top, there are filters for Subscription (Visual Studio Ultimate with...), Resource group (20 selected), Resource type (Virtual machines), Signal type (All sources), and Status (Enabled). Below the filters, a message says 'Displaying 1 - 1 rules out of total 1 rules'. A search bar is present with the placeholder 'Search alert rules based on rule name and condition...'. The main table lists one rule:

Name	Condition	Status	Target resource	Target Resource Type	Signal type
Contoso Disk Monitoring	Whenever the average os disk r...	Enabled	AZ500vM1	virtual machines	Metrics

Figure 3-12 Activity log results after filtering

4. The alert rule that you created appears in the list. To edit the rule, you just need to click it. If you need to create a new alert rule, click the **New Alert Rule** button. Both steps will lead you to the **Create Alert Rule** page, which was previously shown in Figure 3-6.

Important RBAC Roles Required

The consumption and management of alert instances requires the user to have the built-in RBAC roles of either monitoring contributor or monitoring reader.

Once an alert is fired, the status of the alert is set to **New**, which means the rule was detected, but it hasn't been reviewed. Keep in mind that the **Alert State** is different and independent of the **Monitor Condition**. While the **Alert State** is set by the user, the **Monitor Condition** is automatically set by the system. When an alert is fired, the alert's **Monitor Condition** is set to **Fired**. When the underlying condition that caused the alert to fire clears (for example, if your condition was to send an alert if the CPU reaches 80 percent utilization, and then the CPU utilization dropped to 50 percent) the monitor condition is set to **Resolved**. You can see this

information in the email—assuming you configured the rule to send an email—as shown in [Figure 3-13](#).

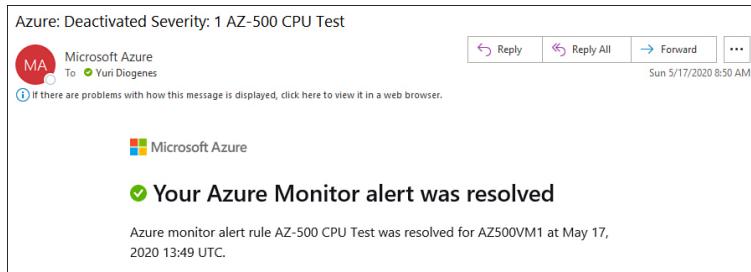


Figure 3-13 Email notification stating that an alert was resolved

Configure diagnostic logging and log retention

In Azure, each resource requires its own diagnostic setting. In these settings, you define the categories of logs and metric data that should be sent to the destinations defined in the setting. Also, you need to define the destination of the log, which includes sending it to the Log Analytics workspace, Event Hubs, and Azure Storage.

It is important to mention that each resource can have up to five diagnostic settings. This means that if the scenario requirement states that you need to send logs to Log Analytics workspace and Azure Storage, you will need two diagnostic settings. Follow these steps to configure the diagnostic settings:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **monitor** and under **Services**, click **Monitor**. The **Monitor | Overview** page appears.
3. In the left navigation pane, under **Settings**, click **Diagnostics Settings**; the **Monitor | Diagnostic** settings page appears, as shown in [Figure 3-14](#).

Figure 3-14 Diagnostics settings page in Azure Monitor

4. As you can see, all resources that can have diagnostic settings appear in this list. For this example, click the Front Door resource that was created in the previous chapter.
5. Click the **Add Diagnostic Setting** option; the **Diagnostics Settings** page appears, as shown in Figure 3-15.

Figure 3-15 Diagnostic settings for a Front Door resource

6. In the **Diagnostic Setting Name** field, type a comprehensive name for this setting.
7. For this specific resource, you have two types of logs. The first is a metric log, in which you can only select the ones that you need for your scenario; the second is the destination log, which can be Log Analytics, a storage account, or Event Hub.
8. In this case, the Contoso Administrator needs to be able to easily query Front Door access logs and WAF logs using a comprehensive query language. To meet this requirement, you need to select **Log Analytics**, which utilizes Kusto Query Language (KQL) to perform queries.
9. When you select the **Send To Log Analytics** option, you will see the option to select the subscription and the Log Analytics workspace that you want to utilize (assuming you have one). Make a selection and click the **Save** button.

- 10.** After saving, the **Save** button is no longer available, which indicates that the changes have been committed.

While the previous sample configuration describes the steps to configure Log Analytics workspace as the diagnostic settings destination, the overall settings can vary according to the destination. For example, if you select storage account, the options shown in [Figure 3-16](#) will appear.

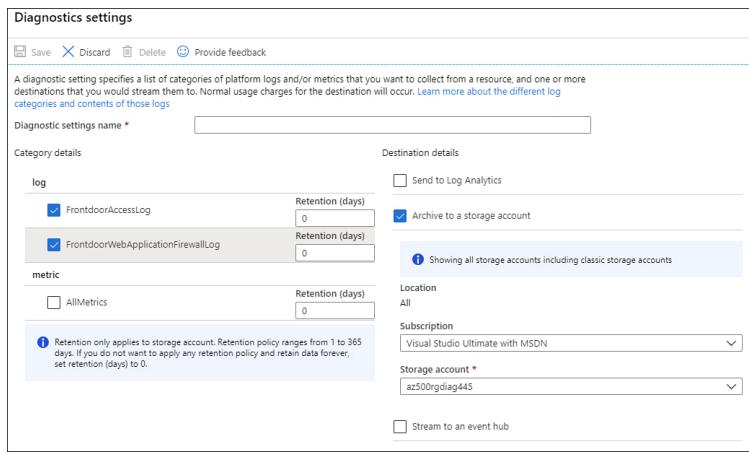


Figure 3-16 Storage account Diagnostic Settings

Notice that when configuring a storage account as your destination, you can customize the retention policy for each log. In a scenario where the requirement is to store the Front Door access logs for 50 days and the WAF logs for 40 days, the best destination for this setting is the storage account because it allows this type of granular configuration.

Consider selecting **Event Hub** as the destination when you need to stream the data to another platform. For example, you might do this if you need to send the Front Door (could be any other Azure resource) access logs to a third-party security information and event management (SIEM) solution, such as Splunk. In this case, using Event Hub is the best option because it allows the logs to be easily streamed to a SIEM solution.



Exam Tip

For the AZ-500 exam, make sure you understand the capabilities of each destination because the requirements of each scenario will lead to different storage options.

Monitoring security logs by using Azure Monitor

Because each Azure resource can have different sets of logs and configurations, you need to ensure that you are collecting all logs that affect your security monitoring.

For Platform as a Service (PaaS) services such as Azure Key Vault, you just need to configure the diagnostic settings to the target location (Log Analytics workspace, storage account, or Event Hub) where the log will be stored. For Infrastructure as a Service (IaaS) VMs, you need more steps because you want to ensure that you are collecting the relevant security logs from the operating system itself.

Data plane logs are the ones that will give you more information about security-related events in IaaS VMs. Assuming that you already have a Log Analytics workspace that will store this data, you will need to do the following actions to configure Azure Monitor to ingest security logs from VMs. First, enable the Log Analytics VM Extension and collect security events from the operating system. Once the data is collected, you can visualize it using the Log Analytics workspace and perform queries using KQL. Assuming that you already have a Log Analytics workspace created, follow these steps to configure this data collection:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **log analytics**, and under **Services**, click **Log Analytics Workspaces**.

3. On the **Log Analytics Workspaces** page, click the workspace in which you want to store the security logs.
4. In the left navigation pane of the workspace page, under **Workspace Data Sources**, click **Virtual Machines**.
5. Click the virtual machine that you want to connect to this workspace. Notice that the **Log Analytics Connection** status appears as **Not Connected**, as shown for the AZ500VM3 in Figure 3-17.

Name	Log Analytics Connection	OS
AZ500VM1	This workspace	Windows
AZ500VM2	Error	Windows
AZ500VM3	Not connected	Windows

Figure 3-17 Virtual Machines that are available in the workspace

6. On the VM's page, click the **Connect** button, as shown in Figure 3-18.

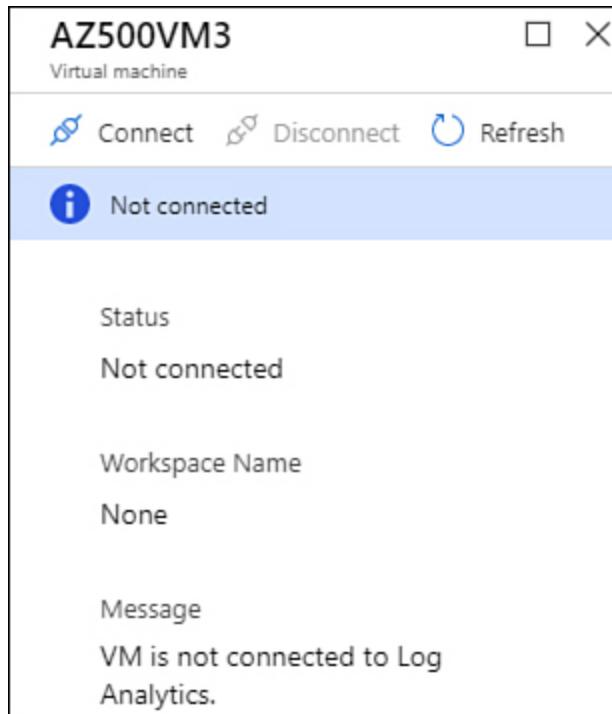


Figure 3-18 Connecting a VM to a workspace

7. At this point, the Log Analytics agent will be installed and configured on this machine. This process takes a few minutes, during which time the **Status** shows as **Connecting**. You can close this page, and the process will continue in background.
8. After the agent is installed, the status will change to **This Workspace**.

9. In the left navigation pane of the main workspace page, under **Settings**, click **Advanced Settings**.
10. On the **Advanced Settings** page, click **Data > Windows Event Logs**, as shown in Figure 3-19.

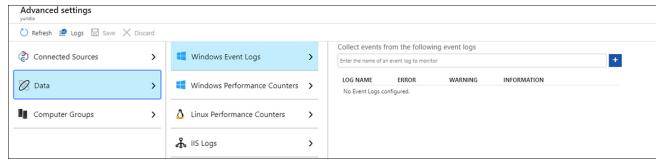


Figure 3-19 Configuring the data source for ingestion

11. In the **Collect Events From The Following Event Logs** field, type **System** and select **System** from the drop-down menu. Click the plus sign (+) to add this log. Leave the default options selected. If you have specific security events that you want to collect, type **security** and select the appropriate events.
12. Click the **Save** button.
13. Click **OK** in the pop-up window and close this page.

Azure Monitor also has solutions that can enhance the data collection for different scenarios. This can be extremely helpful for security monitoring. You can also leverage an Azure Resource Manager (ARM) template to deploy the agent in scale; when doing so, you will need two parameters: the workspace ID and the workspace key.

Security and Audit solution

Monitoring solutions leverage services in Azure to provide additional insight into the operation of an application or service. These solutions collect log data and provide queries and views to analyze collected data. Solutions require a Log Analytics workspace to store data collected by the solution and to host its log searches and views.

If you add the **Security And Audit** solution to your workspace, you automatically will be able to collect Windows security events that are configured according to audit policy best practices. This will allow you to search for specific events in your workspace. Follow

these steps to add the **Security And Audit** solution to your workspace:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **solutions**, and under **Services**, click **Solutions**.
3. On the **Solutions** page, click the **Add** button.
4. In the **Search The Marketplace** field, type **security and audit** and press Enter.
5. In the search results, click the **Security And Audit** tile.
6. Click the **Create** button.
7. Select the workspace to which you want to add this solution and click the **Create** button.

After the solution is added, you can open the Log Analytics workspace by choosing **Solutions** under **General** in the left navigation pane. See [Figure 3-20](#).

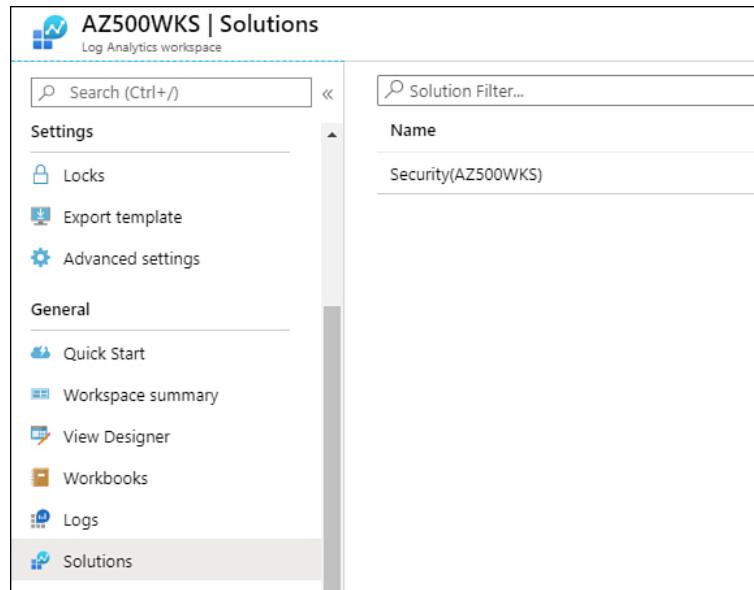


Figure 3-20 Log Analytics workspace where the new solution is shown

Searching security events in Log Analytics workspace

Now that the security events are stored in the workspace, you can start searching for events that might indicate suspicious activity. To access the logs in the workspace, go to the workspace's main page, and in the left

navigation pane, click the **Logs** option under **General**; the **New Query** page appears, as shown in Figure 3-21.

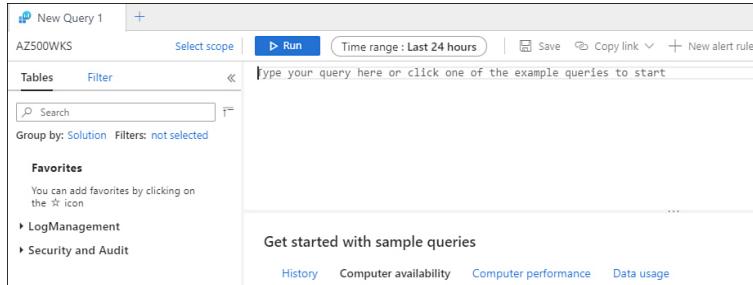


Figure 3-21 New Query page

The scenarios that follow provide more examples of how these queries can be useful when investigating security events related to authentication:

- Contoso's security admin is investigating a potential lateral movement in Contoso's network, and the admin knows that one of the ways to perform this lateral movement is by doing account enumeration. He would like to know all computers that were targeted by this enumeration. The query used to accomplish this task is shown here:

[Click here to view code image](#)

```
SecurityEvent | where EventID == 4799
```

The EventID 4799 is triggered every time a security-enabled local group membership is enumerated. The query result will list all computers that have this event.

- Fabrikam's security admin is investigating the use of a nonauthorized software in its environment. He wants to know when this software was launched. He is not sure what the exact command line is for the software, but he knows that it starts with CM. The query used to accomplish this task is shown here:

[Click here to view code image](#)

```
SecurityEvent | where EventID == 4688 and  
CommandLine contains "cm"
```

The EventID 4688 is triggered every time a new process is created, and the `CommandLine` attribute will evaluate the event's `CommandLine` field to verify whether it contains the string cm.

- Contoso's security admin received a request to report all successful anonymous login attempts coming from the network. The query used to accomplish this task is

[Click here to view code image](#)

```
SecurityEvent | where EventID == 4624 and  
Account contains "anonymous logon" and  
LogonType == 3
```

The EventID 4624 is triggered every time a successful log in occurs; the Account attribute will only filter for anonymous login. With the LogonType attribute set to 3, it will only filter for network attempts.

SKILL 3.2: MONITOR SECURITY BY USING AZURE SECURITY CENTER

In large organizations where it's necessary to have a centralized standard across multiple subscriptions, it is common to use Azure Management Groups to aggregate all subscriptions that share a common set of policies. Security Center enables you to have a centralized view across multiple subscriptions to ensure you have a better visibility of your cloud security posture. This section of the chapter covers the skills necessary to configure security policies in Security Center according to the Exam AZ-500 outline.

Evaluate vulnerability scans from Azure Security Center

Vulnerability assessment is a key component of any security posture management strategy. Security Center standard tier provides a built-in vulnerability assessment capability for your Azure VMs based on an industry lead vulnerability management solution, Qualys. This integration has no additional cost, as long as Security Center is using the standard tier pricing model. If you are using the Free tier, you will still receive a recommendation to install the vulnerability assessment in your machine, but this recommendation (which is not suggesting the built-in vulnerability assessment) requires

you to have the license for your vulnerability assessment solution, which can be Qualys or Rapid7.

Assuming you have the Standard tier enabled, Security Center will identify VMs that don't have a vulnerability assessment solution installed, and it will trigger a security recommendation suggesting the built-in Qualys extension be installed. This recommendation is similar to the example shown in [Figure 3-22](#).

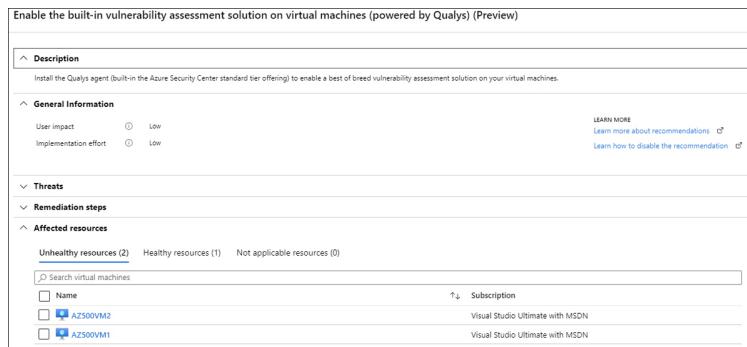


Figure 3-22 Recommendation to install the built-in Qualys extension

To install this vulnerability assessment solution, you need write permissions on the VM to which you are deploying the extension. Assuming that you have the right level of privilege, you will be able to select the VM from the list shown on the **Unhealthy Resources** tab and clicking in the **Remediate** button. This recommendation has the Quick-Fix capability, which means that you can trigger the extension installation directly from this dashboard. Like any extension in Azure, the Qualys extension runs on top of the Azure Virtual Machine agent, which means it runs as Local Host on Windows systems and as Root in Linux systems.

The VMs that already have the agent installed will be listed under the **Healthy Resources** tab. When Security Center cannot deploy the vulnerability scanner extension to the VMs, it will list those VMs on the **Not Applicable Resources** tab. VMs might appear in this tab if they are part of a subscription that is using the Free

pricing tier or if the VM image is missing the `ImageReference` class (which is used on custom images and VMs restored from backup). Another reason for a VM to be listed on this tab is if the VM is not running one of the supported operating systems:

- Microsoft Windows (all versions)
- Red Hat Enterprise Linux (versions 5.4+, 6, 7.0 through 7.7, 8)
- Red Hat CentOS (versions 5.4+, 6, 7.0 through 7.7)
- Red Hat Fedora (versions 22 through 25)
- SUSE Linux Enterprise Server (versions 11, 12, 15)
- SUSE OpenSUSE (versions 12, 13)
- SUSE Leap (version 42.1)
- Oracle Enterprise Linux (versions 5.11, 6, 7.0 through 7.5)
- Debian (versions 7.x through 9.x)
- Ubuntu (versions 12.04 LTS, 14.04 LTS, 15.x, 16.04 LTS, 18.04 LTS)

Important ASC Vulnerability Scanner

Security Center built-in vulnerability scanner solution does not integrate with the Qualys Console.

If you are deploying this built-in vulnerability assessment on a server that has restricted access to the Internet, it is important to know that during the set up process, a connectivity check is done to ensure that the VM can communicate with Qualys's cloud service on the following two IP addresses: 64.39.104.113 and 154.59.121.74.

Once the extension is installed in the target VM, the agent will perform the vulnerability assessment of the VM through a scan process. The scan result will be surfaced in another security recommendation, which is called **Remediate Vulnerabilities Found On Your Virtual Machines (Powered By Qualys)**. A sample of this recommendation is shown in Figure 3-23.

ID	Security Check	Category	Applies To	Severity
100319	Microsoft Internet Explorer Security Upd...	Internet Explorer	9 of 27 resources	High
91586	Microsoft Windows Security Update for ...	Windows	8 of 27 resources	High
100389	Microsoft Internet Explorer Security Upd...	Internet Explorer	8 of 27 resources	High
91582	Microsoft Windows Security Update for ...	Windows	8 of 27 resources	High

Figure 3-23 List of vulnerabilities found during the scan

On this page, you can see the list of findings in the **Security Checks** section. If you click a specific security check, Security Center will show another blade with the details of that vulnerability, which includes the **Impact; Common Vulnerabilities; Exposure (CVE)** (located under **General Information** section); the **Description** of the type of threat; the **Remediation** steps; **Additional References** for this security check; and the list of **Affected Resources**. See Figure 3-24.

Name	Subscription
vm4	ASC DEMO

Figure 3-24 Vulnerability details blade

The deployment of these recommended remediations are done out-of-band; in other words, you will deploy them outside Security Center. For example, if a security check requires you to install a security update on your target computer, you will need to deploy that security update using another product, such as Update Management. Some other remediations will be more about security best practices. For example, the security check 105098 (Users Without Password Expiration) recommends that you create a password policy that has an expiration date. This is usually deployed using Group Policy in Active Directory.

Vulnerability scanning for SQL

Another category of vulnerability scanning that is natively available in Security Center is the vulnerability assessment for SQL Servers. This capability is part of the integration of Security Center with SQL Advanced Data Security (ADS) feature. You can enable this feature in the Security Center **Pricing Tier** setting, which will enable ADS for all databases in the subscription, or you can enable it only on the databases that you want to have this capability.

When you enable ADS, threat protection is available for SQL. Threat protection for Azure SQL Database detects anomalous activities that indicate unusual and potentially harmful attempts to access or exploit databases. For example, an alert that may be generated by this feature is the possible vulnerability to SQL Injection. This alert might indicate a possible vulnerability to SQL injection attacks. Usually there are two possible reasons for a faulty statement: a defect in application code might have constructed the faulty SQL statement, or the application code/stored procedures didn't sanitize user input.

When Security Center identifies that there are databases that don't have this feature enabled, it will trigger a security recommendation, as shown in [Figure 3-25](#).

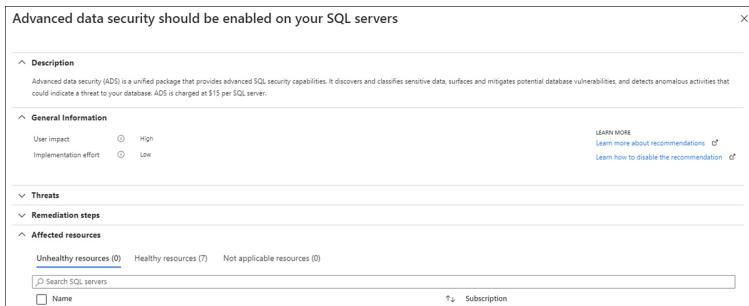


Figure 3-25 Security recommendation to enable ADS

After this feature is enabled, Security Center also will indicate that you also need to enable the vulnerability assessment for your SQL servers (see [Figure 3-26](#)).

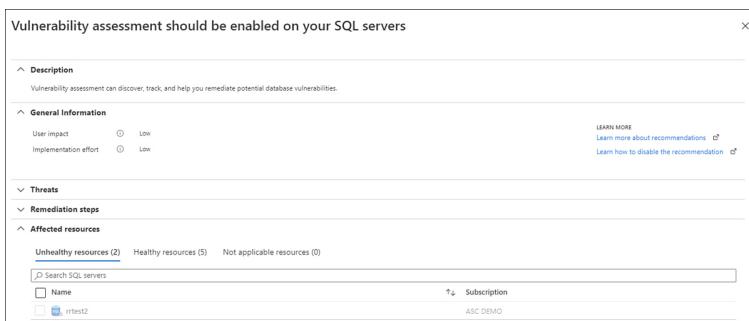


Figure 3-26 Security recommendation to enable vulnerability assessment in SQL



Exam Tip

For the AZ-500 exam, make sure to remember the vulnerability scanning options and that the built-in vulnerability assessment for VMs in Security Center can be deployed using the Quick-Fix feature.

Configure Just-In-Time VM access by using Azure Security Center

When the scenario requires that you reduce the attack surface of IaaS VMs, you should ensure that you are leveraging a Security Center Standard tier capability called Just-In-Time (JIT) VM access. The intent of this capability is to ensure that management ports are not exposed to the Internet all the time. Because the majority of the attacks against IaaS VMs will try to utilize techniques such as RDP or SSH brute force, VMs that have those management ports open will be more susceptible to being compromised.

When you enable JIT VM access, Security Center hardens the inbound traffic to your Azure VMs by creating a Network Security Group (NSG) rule. This rule is based on the selected ports on the VM to which inbound traffic will be locked down. Security Center configures the Network Security Groups (NSGs) and Azure Firewall to allow inbound traffic to the selected ports and requested source IP addresses or ranges, for the amount of time that was specified. After the time has expired, Security Center restores the NSGs to their previous states.

Important Established Connections

When the time expires, the connections that are already established will not be interrupted.

To configure or edit a JIT VM access policy for a VM, you will need write access for the scope of the subscription or resource group for the following objects:

- Microsoft.Security/locations/jitNetworkAccessPolicies/write
- Microsoft.Compute/virtualMachines/write

The user who is requesting access to a VM configured with JIT will need read access on the scope of the

subscription or resource group for the following objects:

- Microsoft.Security/locations/jitNetworkAccessPolicies/initiate/action
- Microsoft.Security/locations/jitNetworkAccessPolicies/*/read
- Microsoft.Compute/virtualMachines/read
- Microsoft.Network/networkInterfaces/*/read

If you want to allow a user to have read access to the JIT policy, you can assign the Security Reader role to the user. If you need a deeper level of customization, you can assign read access for the following objects:

- Microsoft.Security/locations/jitNetworkAccessPolicies/read
- Microsoft.Security/locations/jitNetworkAccessPolicies/initiate/action
- Microsoft.Security/policies/read
- Microsoft.Compute/virtualMachines/read
- Microsoft.Network/*/read

Follow these steps to configure JIT VM access in Security Center:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **security**, and under **Services**, click **Security Center**.
3. In the left navigation pane, in the **Advanced Cloud Defense** section, click **Just In Time VM Access**. The **Security Center | Just In Time VM Access** page appears, as shown in Figure 3-
27.

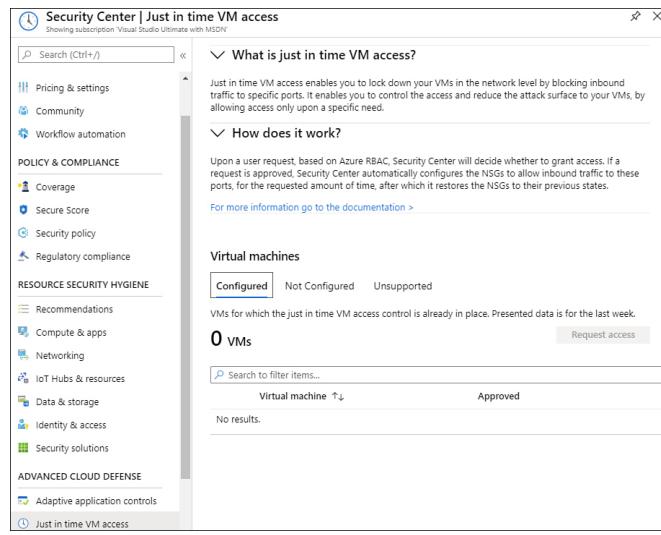


Figure 3-27 JIT VM Access main page

4. In the example shown in Figure 3-27, there are no VMs configured (on the **Configured** tab). If you click the **Not Configured** tab, you should see all the VMs that can support this solution but that have not yet been configured. On the **Unsupported** tab, you will see all VMs that can't use this feature, which include VMs that are missing an NSG, classic VMs, or VMs that are in a subscription that is using the free tier.
5. Click the **Not Configured** tab, select the VM on which you want to enable JIT, and click **Enable JIT On 1 VM** button. The **JIT VM Access Configuration** page appears, as shown in Figure 3-28.

JIT VM access configuration					
AZ500VM3					
+ Add Save Discard					
Configure the ports for which the just in time VM access will be applicable					
Port	Protocol	Allowed source IPs	IP range	Time range (hours)	
22 (Recommended)	Any	Per request	N/A	3 hours	...
3389 (Recommended)	Any	Per request	N/A	3 hours	...
5985 (Recommended)	Any	Per request	N/A	3 hours	...
5986 (Recommended)	Any	Per request	N/A	3 hours	...

Figure 3-28 Ports available to configure JIT

6. You can select one of the default ports according to the protocol for which you want to allow access: 22 (SSH), 3389 (RDP), and 5985/5986 (WinRM). You can also click the **Add** button if you want to customize the port on which you want to allow inbound traffic. For this example, click **3389** and the **Add Port Configuration** blade appears, as shown in Figure 3-29.

Add port configuration X

Port *

Protocol
Any TCP UDP

Allowed source IPs
Per request CIDR block

IP addresses (i)

Max request time
 3 (hours)

Discard OK

Figure 3-29 Port configuration

7. On this blade, you can customize the **Port** as well as the **Protocol** type, the **Allowed Source IP** ranges that are allowed to access

(which could be the request IP or a block of IP addresses), and the time range (**Max Request Time**) for which this rule will stay enabled. After finishing those configurations click the **OK** button.

8. If you are not using the other ports, you can select each of the unused ports, click the ellipsis at the end of each port, and select **Delete**.
9. Click the **Save** button to commit the changes.

If you want to see the changes that JIT VM access made to your VM, open the VM's properties and click

Networking. The example shown in [Figure 3-30](#) shows a new rule (the first rule in the list) that was created by JIT to deny access to those ports. Because this rule is managed by JIT, do not make any manual changes to it.

Inbound port rules	Outbound port rules	Application security groups	Load balancing
Network security group AZ500VM3-nsg (attached to network interface: az500vm3210)			Add inbound port rule
Impacts 0 subnets, 1 network interfaces			
Priority	Name	Port	Protocol
1000	▲ SecurityCenter-JITRule_37335191_C... 22.33.0.59:5986	Any	Any
1001	▲ RDP	3389	TCP
65000	AllowVnetInBound	Any	Any
65001	AllowAzureLoadBalancerInBound	Any	AzureLoadBalancer
65500	DenyAllInBound	Any	Any

Figure 3-30 Inbound port rules with the addition of the JIT VM access rule

Now that JIT is configured, let's see how to request access to a VM with JIT enabled. Use the following steps to perform this action:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **security**, and under **Services**, click **Security Center**.
3. In the left navigation pane, in the **Advanced Cloud Defense** section, click **Just In Time VM Access**.
4. On the **Security Center | Just In Time VM Access** page, under the **Configured** tab, select the VM for which you enabled JIT, and click the **Request Access** button, as shown in [Figure 3-31](#).

Virtual machines								
Configured	Not Configured	Unsupported	VMs for which the just in time VM access control is already in place. Presented data is for the last week.					
1 VMs								
Request access								
Search to filter items...								
Virtual machine ↑↓	Approved	Last access ↑↓	Connection details	Last user ↑↓	...			
<input checked="" type="checkbox"/> AZ500VM3	0 Requests	N/A	-	N/A	...			

Figure 3-31 Requesting access to a VM using JIT

5. On the **Request Access** page, you have the option to select the **Port** that you want to access, the **Allowed Source IP**, and the **Time Range (Hours)**, as shown in [Figure 3-32](#).



Figure 3-32 Customizing the access

6. Select **RDP**, leave the other options with the default selection, and click the **Open Ports** button.

Now you can initiate an RDP session to this VM. When you do that, go back to Security Center and notice that the status of the VM changed to show that the connection is currently active and who initiated this session. See [Figure 3-33](#).

Virtual machine	Approved	Last access	Connection details	Last user
AZ500VM3	1 Requests	Active now	Ports: 3389	live.com@yuridiogenes@hotmail.com

Tip Connection Icon Might Vary
The icon that appears in the Connections Details column can vary because it can also be the Azure Firewall icon.

Figure 3-33 VM status showing details about the connection

In a scenario where a VM has JIT enabled and is located in a subnet with a user-defined route that points to an Azure Firewall as a default gateway, you might experience problems accessing the VM using JIT. This issue happens because of the asymmetric routing behavior, which means that the request comes in via using the virtual machine public IP address, where JIT has opened the access. However, the response (return path) is via the Azure Firewall, which evaluates the request, and because there is no established connection, it drops the packet. In scenarios like this, you need to

move the resource to a subnet that doesn't have a user-defined route.

Configure centralized policy management by using Azure Security Center

Security Center recommendations are derived from Azure Policy. By default, Security Center has an initiative called ASC Default, which is assigned to the subscription once you activate Security Center for the first time. The activation process happens in the background, and it is triggered when you visit Security Center dashboard for the first time.

To recap some important concepts regarding Azure Policy, and how these policies are correlated with Security Center, review the diagram shown in [Figure 3-34](#).

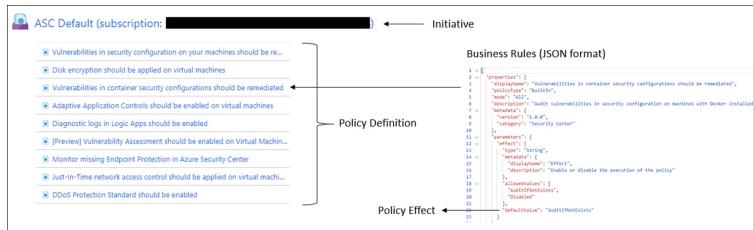


Figure 3-34 Correlation between Security Center initiative and Azure Policy

The ASC Default initiative has multiple policy definitions that can be accessed individually using Azure Policy. Policy definitions are used to compare the properties of Azure resources with business rules, which in this case are implemented in JSON. Each policy definition in Azure Policy has a single effect, also called policy effect. That effect determines what happens when the policy rule is evaluated to match. Security Center uses the following effects: Audit, AuditIfNotExists, and Disabled. This means that Security Center is not used for policy enforcement, but it is used for security monitoring and compliance visualization. Policy

enforcement is covered in “Skill 3.4 Configure security policies,” later in this chapter.

Security Center policies can be customized, which means that if you have a scenario where the organization is using a third-party multi-factor authentication (MFA) solution instead of Azure MFA, you can disable the MFA recommendation in Security Center because this recommendation is based on a check to determine whether you are using Azure MFA. While it is recommended to always use the default settings for these policies, there will be scenarios in which you might have to customize and change the effect from `AuditIfNotExists` to `Disabled`.

Implementing centralized policy management

Let’s start reviewing a fictitious scenario for Fabrikam: Consider a scenario in which Fabrikam has a single Azure tenant with multiple business units across the company. Each business unit has its own subscription and follows the policy standards that were established according to its branch, which is based on Fabrikam’s geolocation across the globe. In this scenario, Fabrikam wants to have centralized policy management for all its business units according to the standards followed by each unit’s branch and country.

To accommodate the requirements of this scenario, you should create one Management Group to represent the branch office in each country and move the subscriptions of each business unit in that branch so that it is under this management group. Once you have this structure, you can assign the Security Center policy to the management group level. It would look similar to the diagram shown in [Figure 3-35](#).

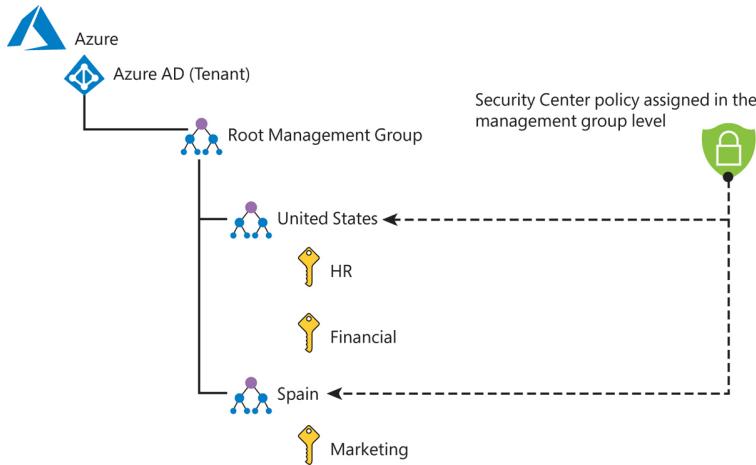


Figure 3-35 Centralized management structure

To make changes to the Security Center initiative, you need to have Security Admin role privileges. You can also make changes if you are the subscription owner. Both Contributor and Reader have access to all Azure Policy operations in which read access is required. Contributors can trigger resource remediation, but they can't create definitions or assignments. Using the scenario described previously in which multiple business units were required, if you want to restrict users in each business unit to only be able to see (read-only operation) the policies, you can assign them to the Security Reader role.

Because security recommendations in Security Center are derived from Azure Policy, you might have a situation in which you need to customize the policy so that the default effect is **Disabled**. Consider a scenario in which Fabrikam is using an endpoint protection solution that is not supported by Security Center. Fabrikam keeps receiving the **Install Endpoint Protection Solution On Virtual Machines** security recommendation. Fabrikam understands that this recommendation is a false positive for its environment because Fabrikam has an endpoint protection installed. However, because it is not supported by Security Center, this recommendation keeps triggering. In this scenario, Fabrikam can change the default effect to **Disabled**. Use the following steps to configure this change in Security Center:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **security**, and under **Services**, click **Security Center**.
3. In Security Center's main dashboard, under the **Policy & Compliance** section, click **Security Policy**.
4. Click the subscription for which you want to change the policy.
5. On the **Security Policy** page, click the **View Effective Policy** button, as shown in Figure 3-36.

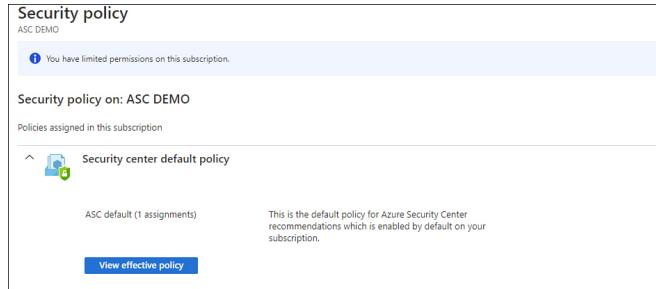


Figure 3-36 Visualizing the security policy in Security Center

6. On the next Security Policy page that appears, you will see all policies that are currently in use. This page is mostly for read-only purposes; if you need to make changes to the policy, click the **[Preview]: Enable Monitoring In Azure Security Center** link for the policy, as shown in Figure 3-37.

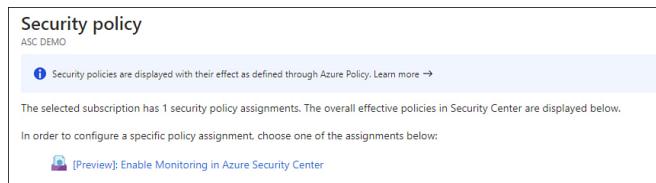


Figure 3-37 Accessing the default policy

7. On the next page, click the **Parameters** tab, as shown in Figure 3-38.

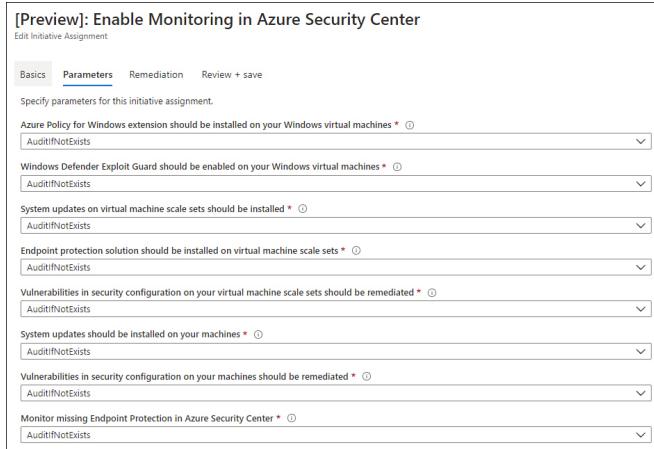


Figure 3-38 Accessing the default policy

8. On this tab is a list of parameters for this initiative, which represents the Security Center recommendations. The goal, in this case, is to disable the missing endpoint protection recommendation. Click the **Monitor Missing Endpoint Protection in Azure Security Center** drop-down menu and select **Disabled**.
9. Click the **Review + Save** button and then click the **Save** button to commit the changes.

Make sure to document the changes you made to the default Security Center initiative, specifically regarding policies that have been disabled. Document the rationale behind your reasoning for disabling the policy and who disabled it.

Configure compliance policies and evaluate for compliance by using Azure Security Center

While Security Center recommendations will cover security best practices for different workloads in Azure, there are some organizations that also need to be compliant with different industry standards. Security Center's Standard tier helps simplify the process for meeting regulatory compliance requirements by using the Regulatory Compliance dashboard.

The Regulatory Compliance dashboard view can help focus your efforts on the gaps in compliance with a

standard or regulation that is relevant for your organization. By default, Security Center provides support for the following regulatory standards: Azure CIS, PCI DSS 3.2, ISO 27001, and SOC TSP. Use the following steps to access the **Regulatory Compliance** dashboard:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **security**, and under **Services**, click **Security Center**.
3. In the Security Center main dashboard, under the **Policy & Compliance** section, click **Regulatory Compliance**; the **Regulatory Compliance** dashboard appears, as shown in Figure 3-39.

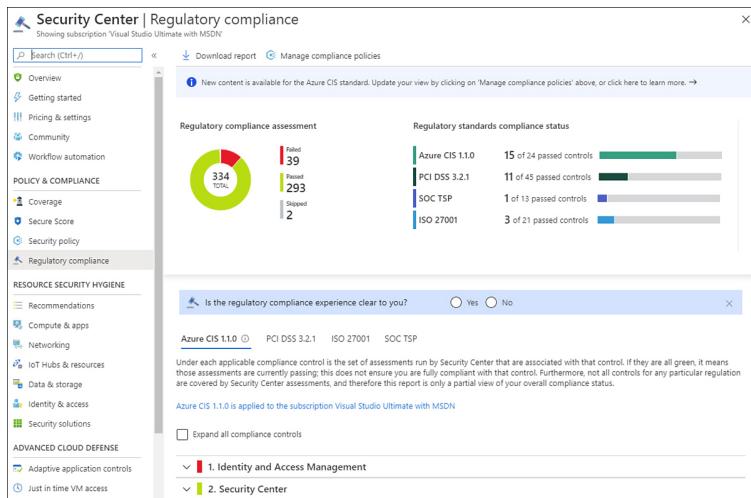


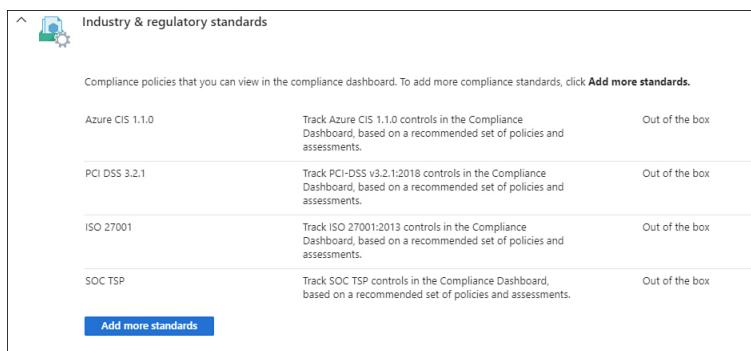
Figure 3-39 Regulatory Compliance dashboard

The top part of the dashboard shows a brief summary of the assessment's status and the individual status of each regulatory standard compliance. In the second part of the dashboard are four tabs. The default tab selection is Azure CIS 1.1. The others are PCI DSS 3.2, ISO 27001, and SOC TSP. You can navigate through the items to see which ones need attention (shown in red) and which ones successfully passed the assessment (shown in green). Also, notice that some controls are unavailable. These controls don't have Security Center assessments associated with them, and no further action is required.

To improve your compliance status, you need to follow the same approach that you did for the security recommendations. In other words, you need to remediate the assessment to comply with the requirements. Assessments are updated approximately every 12 hours, so if you remediate an assessment, you will only see the effect on your compliance data after the assessments run.

In some scenarios, the organization will need to comply with different industry standards. Microsoft is constantly reviewing new standards and making them available in the Azure platform, which means that in addition to the industry standards that come out of the box in Security Center, you can add NIST SP 800-53 R4, SWIFT CSP CSCF-v2020, UK Official and UK NHS, Canada Federal PBMM, and Azure CIS 1.1.0 (new)—an updated version of Azure CIS 1.1.0.

To add a new compliance standard, you need to be the subscription owner or policy contributor. Assuming you have the right privilege, you can just click the **Manage Compliance Policies** button in the **Regulatory Compliance** dashboard, and then on the Security policy page, click the subscription to which you want to add the standard. In the resulting page, click the **Add More Standards** button, as shown in Figure 3-40.



The screenshot shows the 'Industry & regulatory standards' section of the Azure Regulatory Compliance dashboard. It lists several compliance standards with their descriptions and status. At the bottom, there is a blue button labeled 'Add more standards'.

Standard	Description	Status
Azure CIS 1.1.0	Track Azure CIS 1.1.0 controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	Out of the box
PCI DSS 3.2.1	Track PCI-DSS v3.2.1:2018 controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	Out of the box
ISO 27001	Track ISO 27001:2013 controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	Out of the box
SOC TSP	Track SOC TSP controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	Out of the box

Add more standards

Figure 3-40 Adding more standards to the Regulatory Compliance dashboard

After you click the **Add More Standards** button, you will have the option to click the **Add** button for each new industry standard available on the list, as shown in [Figure 3-41](#).

Add regulatory compliance standards		
Click Add on the standards that you want to add to the regulatory compliance dashboard and then assign it to the subscription. After completing the assignment, the custom policies will be available in the Regulatory compliance dashboard.		
<input type="text"/> Search to filter items...		
Name	Description	Add
NIST SP 800-53 R4	Track NIST SP 800-53 R4 controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>
UK OFFICIAL and UK NHS	Track UK OFFICIAL and UK NHS controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>
Canada Federal PBM	Track Canada Federal PBM controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>
Azure CIS 1.1.0 (New)	Track Azure CIS 1.1.0 (New) controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>
Azure Security Benchmark	Track Azure Security Benchmark controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>
HIPAA	Track HIPAA controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>
SWIFT CSP CSCF v2020	Track SWIFT CSP CSCF v2020 controls in the Compliance Dashboard, based on a recommended set of policies and assessments.	<input type="button" value="Add"/>

Figure 3-41 Available regulatory compliance standards

Important Standards Cannot be Removed

You cannot remove the out-of-the-box industry standards from the dashboard; you can only add more standards to the dashboard.

Once you add the new standard, a new tab will be created in the main Regulatory Compliance dashboard. There are some scenarios in which you might need to send a summary report of your regulatory compliance status to someone. If you need to do this, you can use the **Download Report** button on the main **Regulatory Compliance** dashboard.

SKILL 3.3: MONITOR SECURITY BY USING AZURE SENTINEL

Azure Sentinel is a Microsoft Security Information and Event Management (SIEM) and Security Orchestration, Automation, and Response (SOAR) solution. You can use this solution to ingest data from different data sources, create custom alerts, monitor incidents, and respond to alerts. This section of the chapter covers the skills necessary to configure container security according to the Exam AZ-500 outline.

Introduction to Azure Sentinel's architecture

To help you to better understand Azure Sentinel's architecture, first, you need to understand the different components of the solution. The major Azure Sentinel components are diagrammed in Figure 3-42.

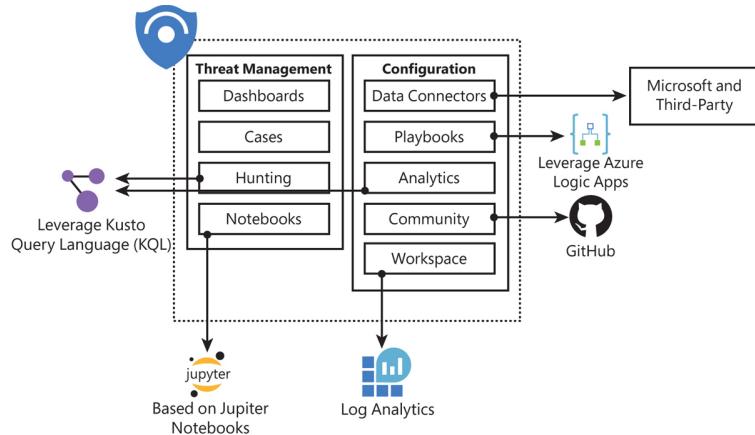


Figure 3-42 Major components of Azure Sentinel

The components shown in Figure 3-42 are presented in more detail in the following list:

- **Dashboards** Built-in dashboards provide data visualization for your connected data sources, which enables you to deep dive into the events generated in those services.
- **Cases** An aggregation of all the relevant evidence for a specific investigation. It can contain one or multiple alerts, which are based on the analytics that you define.
- **Hunting** A powerful tool to investigators and security analytics who need to proactively look for security threats. The searching capability is powered by Kusto Query Language (KQL).
- **Notebooks** By integrating with Jupyter notebooks, Azure Sentinel extends the scope of what you can do with the data that was collected. It combines full programmability with a collection of libraries for machine learning, visualization, and data analysis.
- **Data Connectors** Built-in connectors are available to facilitate data ingestion from Microsoft and partner solutions.
- **Playbook** A collection of procedures that can be automatically executed upon an alert that is triggered by Azure Sentinel. Playbooks leverage Azure Logic Apps, which help you automate and orchestrate tasks and workflows.

- **Analytics** Enables you to create custom alerts using Kusto Query Language (KQL).
- **Community** The Azure Sentinel Community page is located on GitHub (<https://aka.ms/ASICommunity>), and it contains Detections based on different types of data sources that you can leverage to create alerts and respond to threats in your environment. It also contains hunting queries samples, Playbooks, and other artifacts.
- **Workspace** Essentially, a Log Analytics workspace is a container that includes data and configuration information. Azure Sentinel uses this container to store the data that you collect from the different data sources.

The sections that follow assume that you already have a workspace configured to use with Azure Sentinel.

Important Sentinel is not Covered in Depth

This book does not cover Azure Sentinel entirely; it only covers the topics that are relevant for the AZ-500 exam. To learn more, see, *Microsoft Azure Sentinel: Planning and implementing Microsoft's cloud-native SIEM solution*, published by Microsoft Press.

Configure Data Sources to Azure Sentinel

The first step to configure a SIEM solution such as Azure Sentinel is ensuring that the data relevant for your requirements is ingested. For example, if you need to collect data related to conditional access policies and legacy authentication-related details using sign-in logs, you need to configure the Azure Active Directory (AD) connector. Azure Sentinel comes with a variety of connectors that enable you to start ingesting data from those data sources with just a couple of clicks. Keep in mind that you need to have those services enabled to start ingesting data using these connectors. Use Table 3-1 to identify some use-case scenarios and to determine which connector is available for each scenario:

Table 3-1 Azure Sentinel connectors and use-case scenarios

Scenario | Connector

You need to gain insights about app usage; conditional access policies; legacy authentication-related details; and activities like user, group, role, app management.	Azure AD
You need to get details of operations such as file downloads, access requests sent, and changes to group events, and you need to set mailbox and details of the user who performed the actions.	Office 365
You need to gain visibility into your cloud apps; get sophisticated analytics to identify and combat cyberthreats; and control how your data travels.	Microsoft Cloud App Security
You need to gain insights into subscription-level events that occur in Azure, including events from Azure Resource Manager operational data; service health events; write operations taken on the resources in your subscription; and the status of activities performed in Azure.	Azure Activity
You need to gain visibility about users at risk, risk events, and vulnerabilities.	Azure AD Identity Protection
You need to gain insights into your security state across hybrid cloud workloads; reduce your exposure to attacks; and respond to detected threats quickly.	Azure Security Center

The connectors shown in this table are considered service-to-service integrations. Also, there are connectors to external solutions using API and others that can perform real-time log streaming using the Syslog protocol via an agent. Following are some examples of external connectors (non-Microsoft) that use agents:

- Check Point
- Cisco ASA
- DLP solutions
- DNS machines - agent installed directly on the DNS machine
- ExtraHop Reveal(x)
- F5
- Forcepoint products
- Fortinet
- Linux servers
- Palo Alto Networks
- One Identity Safeguard
- Other CEF appliances
- Other Syslog appliances
- Trend Micro Deep Security
- Zscaler

To configure data connectors, you will need the right level of privilege. The necessary roles for each connector are determined per connector type. For example, to configure the Azure AD connector, you will need the following permissions:

- **Workspace** Read and write permissions are required.
- **Diagnostic Settings** Required read and write permissions to AAD diagnostic settings.
- **Tenant Permissions** Required Global Administrator or Security Administrator roles on the workspace's tenant.

Note Azure AD Logs

To ingest Azure AD logs into the Azure Sentinel workspace, you will also need an Azure AD P1/P2 License.

While this connector has a decent list of permission requirements, some others will be simpler. For example, to configure the Azure Activity connector, you just need read and write permissions in the workspace. The requirements for each connector will be available on the connector's page in Azure Sentinel.

For this initial scenario, let's say that Fabrikam wants to ensure that all events from Azure Resource Manager operational data; service health events; write operations taken on Fabrikam's subscription resources; and the status of activities performed in Azure are ingested in Azure Sentinel. To accomplish that, you need to configure the Azure Activity connector. Follow these steps:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **sentinel**, and under **Services**, click **Azure Sentinel**.
3. On the Azure Sentinel workspaces page, click the workspace that you want to use with Azure Sentinel; the **Azure Sentinel | Overview** page appears (see Figure 3-43).

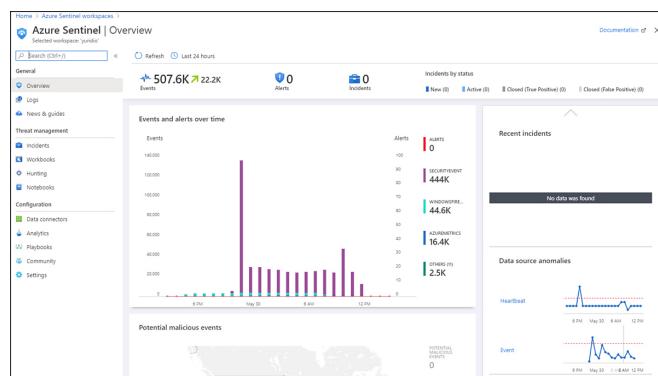


Figure 3-43 Azure Sentinel Overview page

Important Azure Sentinel Dashboard

If this is the first time you've launch Azure Sentinel after configuring the workspace, your dashboard will not have any data because data collection is not configured yet.

4. In the left navigation pane, under **Configuration**, click **Data Connectors**.
5. On the **Data Connectors** page, click **Azure Activity**.
6. On the **Azure Activity** blade, click the **Open Connector Page** button, as shown in Figure 3-44.

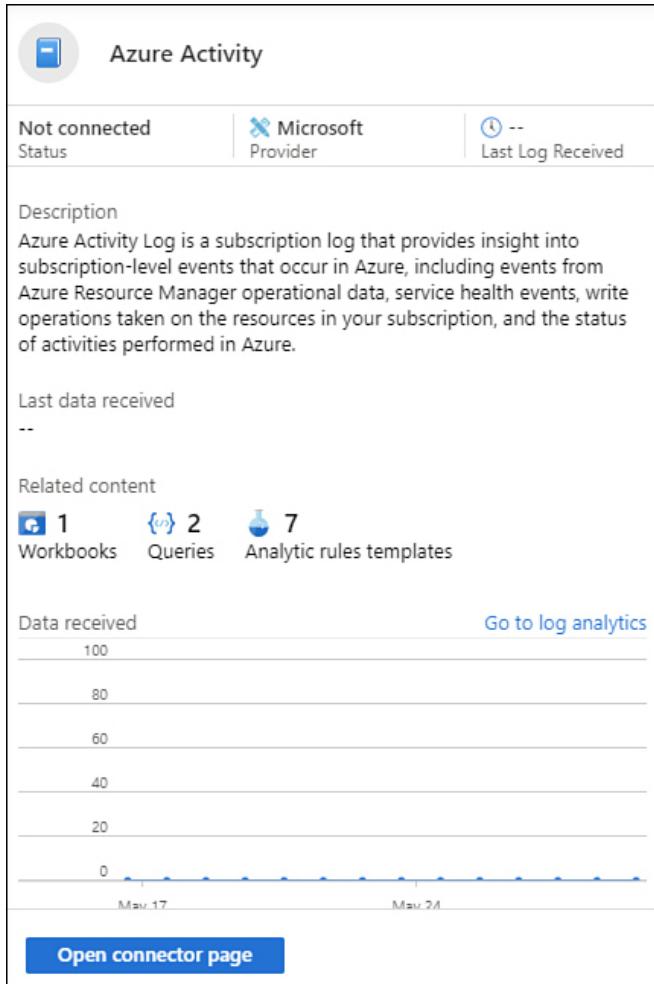


Figure 3-44 Azure Activity blade

7. On the **Azure Activity** page, click the **Configure Azure Activity Logs** link, as shown in Figure 3-45.

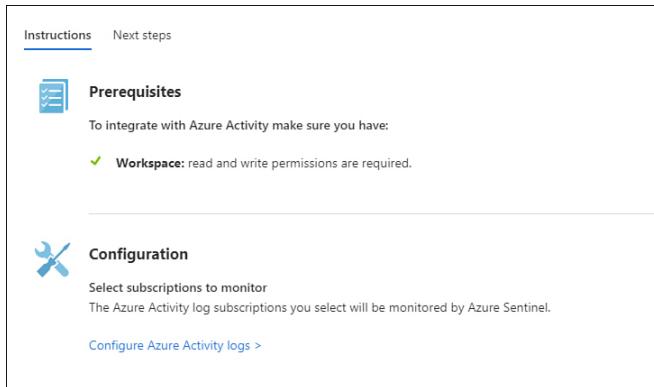


Figure 3-45 Azure Activity data connector configuration

8. On the **Azure Activity Log** blade, click the subscription that you want to connect, and in the **Subscription** blade that appears, click the **Connect** button, as shown in [Figure 3-46](#).

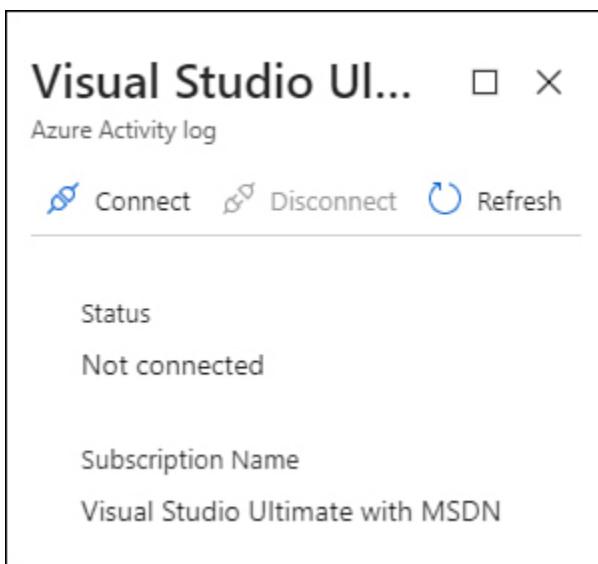


Figure 3-46 Subscription blade

9. Once it finishes connecting, click the **Refresh** button to update the button's status. You will see that now the **Disconnect** button is available.
10. Close the **Subscription** blade, close the **Azure Activity Log** blade, and close the **Azure Activity** connector page.
11. When you return to the **Azure Sentinel | Data Connectors** page, make sure to click the **Refresh** button to update the Azure Activity data connector status.

The core steps to configure Azure Sentinel data connectors are very similar, though depending on the connector, you might need to execute more steps. This is

true mainly for external connectors and services in different cloud providers. For example, if you need to connect to Amazon AWS to stream all AWS CloudTrail events, you will need to perform some steps in the AWS account.

Create and customize alerts

After the different data sources are connected to Azure Sentinel, you can create custom alerts, which are called Analytics. There are two types of analytics that can be created: a scheduled query rule and a Microsoft incident creation rule. A scheduled query rule allows you to fully customize the parameters of the alert, including the rule logic and the alert threshold. A Microsoft incident creation rule allows you to automatically create an incident in Azure Sentinel for an alert that was generated by a connected service. This type of rule is available for alerts generated by Azure Security Center, Azure Security Center for IoT, Microsoft Defender Advanced Threat Protection, Azure AD Identity Protection, Microsoft Cloud App Security, and Azure Advanced Threat Protection.

When considering which one you need to utilize, make sure to understand the prerequisites for the scenario because those requirements will determine the type of rule that you need to create. For example, if the requirement is to customize the alert with parameters that will determine the query scheduling and alert threshold, then the best option is the scheduled query rule. For this scenario, Fabrikam wants to create a medium severity alert every time a VM is deleted and an incident should be created for further investigation. Follow these steps to create a scheduled query rule:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **sentinel**, and under **Services**, click **Azure Sentinel**.

3. On the Azure Sentinel workspaces page, click the workspace that you want to use with Azure Sentinel; the **Azure Sentinel | Overview** page appears.
4. In the left navigation pane, under **Configuration**, click **Analytics**.
5. Click the **Create** button and select the **Scheduled Query Rule** option. The **Analytic Rule Wizard – Create New Rule** page appears, as shown in Figure 3-47.

The screenshot shows the 'Analytic rule wizard - Create new rule' page. At the top, there's a breadcrumb trail: Home > Azure Sentinel workspaces > Azure Sentinel | Analytics >. The main title is 'Analytic rule wizard - Create new rule'. Below the title, there are tabs: General (which is selected), Set rule logic, Incident settings (Preview), Automated response, and Review and create. A subtitle says 'Create an analytic rule that will run on your data to detect threats.' Under 'Analytic rule details', there are four sections: 'Name *' with a text input field, 'Description' with a text input field, 'Tactics' with a dropdown menu showing '0 selected', and 'Severity' with a dropdown menu showing 'Medium'. Below these is a 'Status' section with two buttons: 'Enabled' (which is highlighted) and 'Disabled'. At the bottom right is a blue button labeled 'Next : Set rule logic >'.

Figure 3-47 Create New Rule page

6. In the **Name** field, type a name for this analytic.
7. Optionally, you can write a full description for this analytic and select the tactic. The **Tactics** drop-down menu contains a list of the different phases available in the cyber kill chain. You should select the appropriate phase for the type of alert that you want to create; for this example, select **Impact**.
8. The **Severity** drop-down menu contains a list of all available levels of criticality for the alert. For this example, leave it set to **Medium**.
9. Because you want to activate the rule after creating it, leave the **Status** set to **Enabled**.
10. Click the **Next: Set Rule Logic** button; the **Set Rule Logic** tab appears, as shown in Figure 3-48.

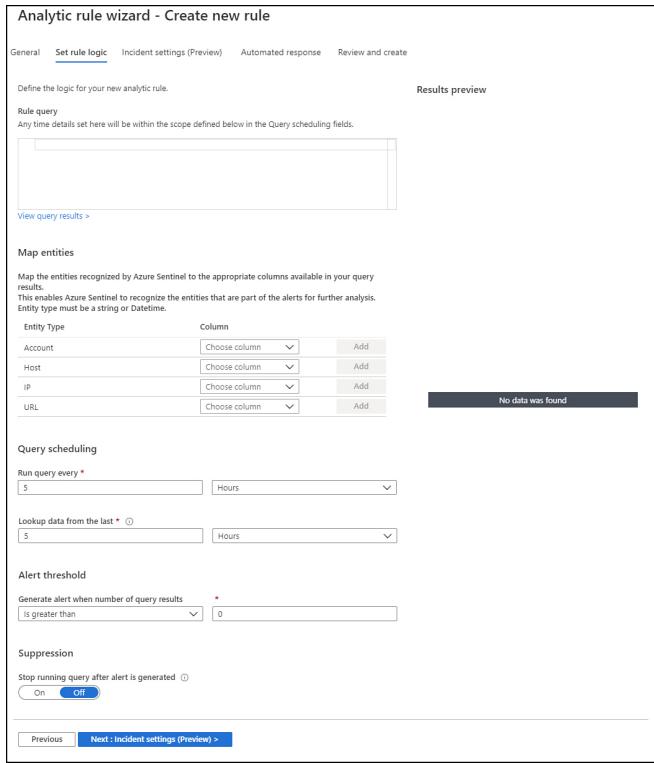


Figure 3-48 Configuring the rule logic

11. In the **Rule query** field, you need to type the KQL query. Because Fabrikam wants to receive an alert when VMs are deleted, type the following sample query:

[Click here to view code image](#)

```
AzureActivity
| where OperationNameValue contains
"Microsoft.Compute/virtualMachines/delete"
```

12. In some scenarios, you might need to customize the **Map Entities** options to enable Azure Sentinel to recognize the entities that are part of the alerts for further analysis. For this scenario, you can leave the default setting.
13. Under **Query scheduling**, the first option is to customize the frequency with which you want to run this query. Because this scenario does not have a specifically defined frequency, leave it set to run every 5 hours.
14. Next, you can customize the timeline in which you want to run this query, under the **Lookup Data From The Last** option. By default, the query will run based on the last 5 hours of data collected. Since in this scenario it was not specifically specified, leave as is.
15. Under **Alert Threshold**, you have the **Generate Alert When Number Of Query Results** drop-down menu. Because this scenario calls for an alert to be generated every time a VM is

deleted, you should leave this set to the default setting, **Is Greater Than 0**.

16. Under **Suppression**, you could choose to stop the query after the alert is generated. In this scenario, leave the default selection, which is **Off**.
17. Click the **Next: Incident settings (Preview)** button; the **Incident Settings** tab appears, as shown in Figure 3-49.

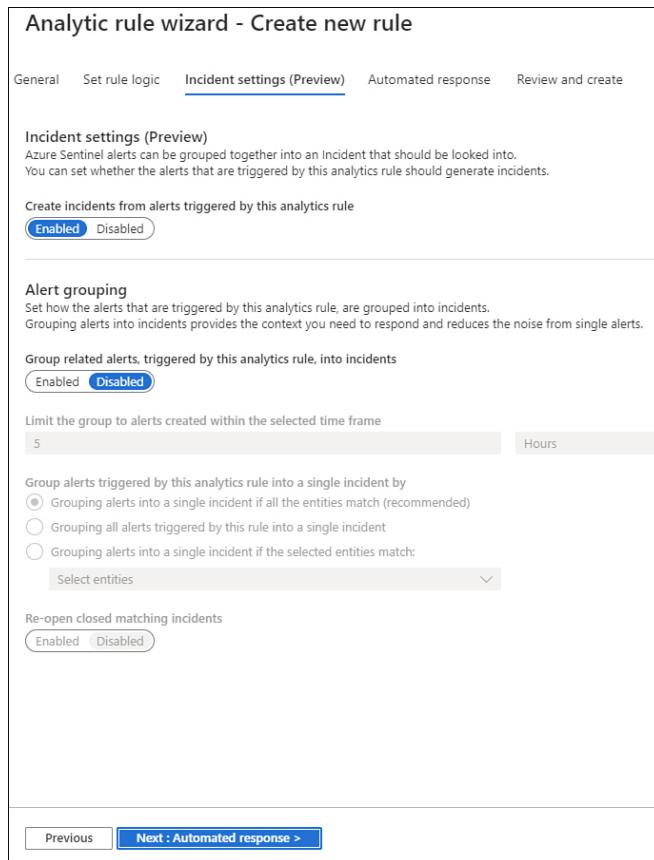


Figure 3-49 Configuring incident settings

18. Leave the **Create Incidents From Alerts Triggered By This Analytics Rule** option selected (which is the default setting) because the scenario requires an incident to be created.
19. Under **Alert Grouping**, you can configure how the alerts that are triggered by this analytics rule are grouped into incidents. For this scenario, leave the default selection, which is **Disabled**.
20. Click the **Next: Automated Response** button; the **Automated Response Tab** appears, as shown in Figure 3-50.

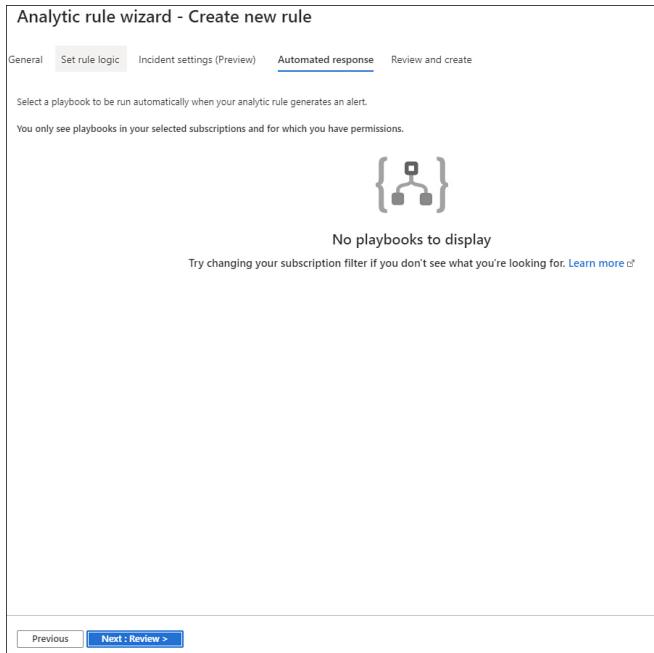


Figure 3-50 Configuring an Automated Response

21. The **Automation Response** tab contains a list of all Azure Logic Apps available. In a new deployment, it is common to see an empty tab because there will be no Logic Apps available. You will learn more about automated responses in the next section of this chapter.
22. Click the **Next: Review** button, review the options, and click the **Create** button.
23. After the rule is created, you will be taken back to the **Azure Sentinel | Analytics** page; the rule appears in the **Active Rules** list. If you click it, you will see the parameters of the rule, as shown in Figure 3-51.

The screenshot shows the configuration page for a custom alert named "VM Deletion".

- Severity:** Medium
- Status:** Enabled
- Id:** e9fb688d-ec9c-4e81-b814-0bccdbce22b2
- Description:** -
- Tactics:** Impact
- Rule query:**

```
AzureActivity  
| where OperationNameValue contains "Microsoft.Com"
```
- Rule period:** Last 5 hours data
- Rule frequency:** Every 5 hours
- Rule threshold:** Trigger alert if query returns more than 0 results
- Suppression:** Not configured

Edit button at the bottom left.

Figure 3-51 Custom alert after creation

While this rule was created specifically for a particular scenario, you can utilize existing templates, which are located on the **Rule Templates** tab in the main **Azure Sentinel | Analytics** page. You can create a scheduled rule type based on different known types of attacks. For example, if you have a scenario in which you need to detect distributed password cracking attempts in Azure AD, you can just create a rule based on the available template, as shown in [Figure 3-52](#).

Home > Azure Sentinel | Analytics >

Analytic rule wizard - Create new rule from template

Distributed Password cracking attempts in AzureAD

General Set rule logic Incident settings (Preview) Automated response Review and create

Create an analytic rule that will run on your data to detect threats.

Analytic rule details

Name *

Description

Tactics

Severity

Status Enabled Disabled

Next : Set rule logic >

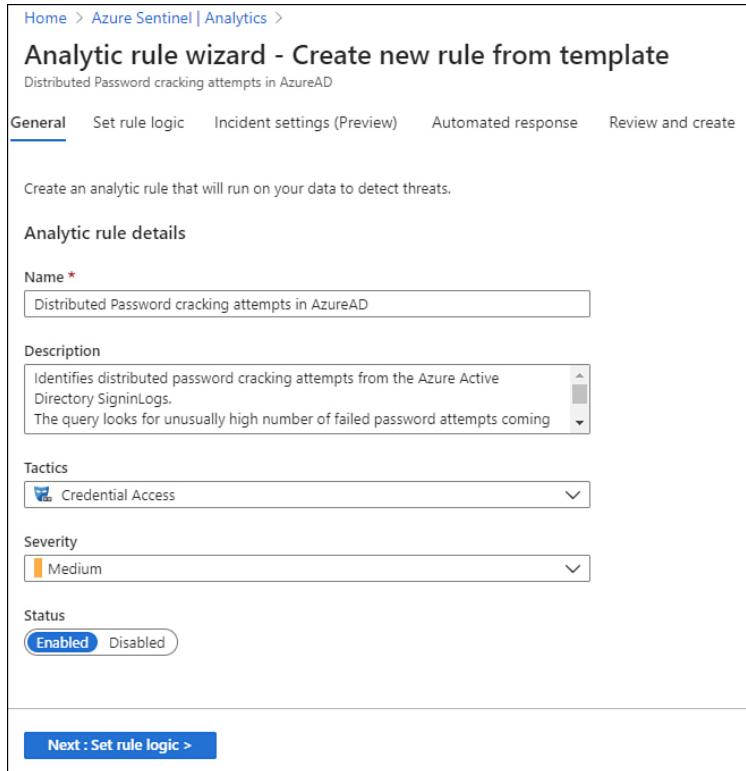


Figure 3-52 Creating an alert based on a template

There are other scenarios in which you might need to simply create an incident in Azure Sentinel based on an alert triggered by a connected service. For example, you might want to create an incident every time an alert is triggered from Azure Security Center. The initial steps are the same. The difference is that in step 5 of the earlier instructions, you would select the **Microsoft Incident Creation** rule. When this option is selected, you will see the **Analytic Rule WizardCreate New Rule** page, as shown in Figure 3-53.

Home > Azure Sentinel | Analytics >

Analytic rule wizard - Create new rule

Create an analytic rule that creates incidents based on alerts generated in another Microsoft security service.

Analytic rule details

Name *

Description

Status

Enabled Disabled

Analytic rule logic

Microsoft security service *

Filter by severity

Any Custom

Include specific alerts

Only create incidents from alerts that contain the following text in the alert name

Exclude specific alerts

Only create incidents from alerts that do not contain the following text in the alert name

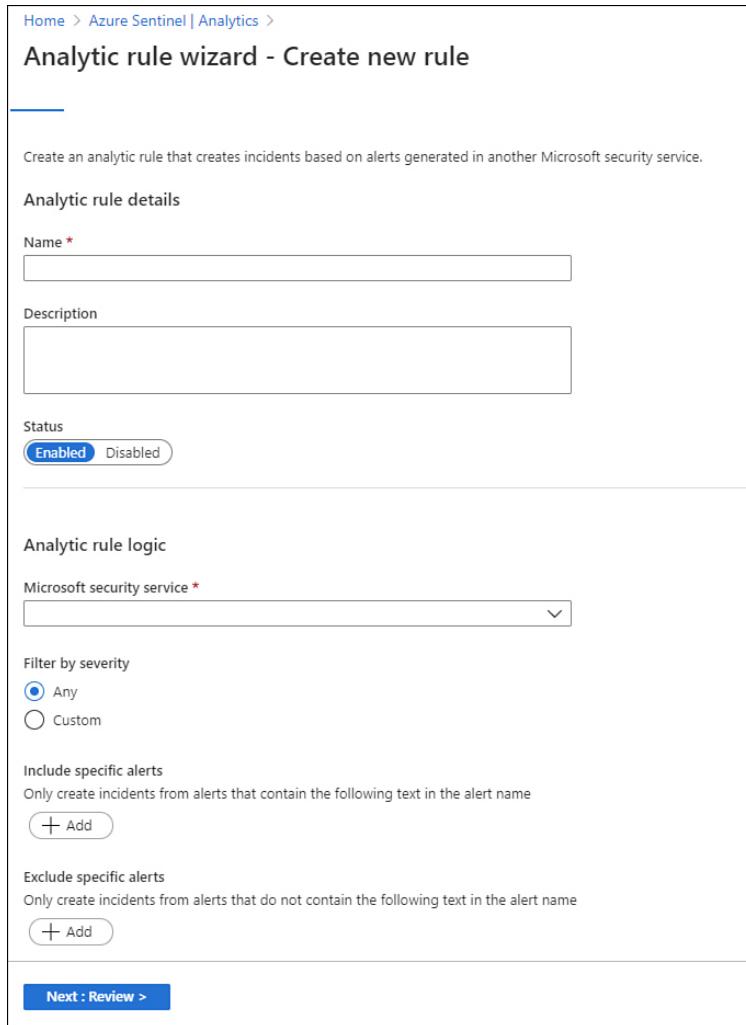


Figure 3-53 Creating an alert based on a connected service

In the **Microsoft Security Service** drop-down menu, you can select the connected service that you want to use as the data source. For example, if you select **Azure Security Center** from the list and you do not customize the included or excluded alerts, Azure Sentinel will create an incident for all alerts triggered by Azure Security Center.

Configure a Playbook for a security event by using Azure Sentinel

Security Playbooks enable you to create a collection of procedures that can be executed from Azure Sentinel when a certain security alert is triggered. Azure Logic

Playbooks. Before creating a Playbook, you should have in mind what you want to automate. Before you start configuring a Playbook, make sure to answer at least the following questions:

- For which alert should I automate a response?
- What steps should be automated if the conditions for this alert are true?
- What steps should be automated if the conditions for this alert are false?

Note Additional Charges Apply

Playbooks leverage Azure Logic Apps, so charges apply on top of your Azure Sentinel pricing.

You can use the Logic App Contributor role or the Logic App Operator role to assign explicit permission for using Playbooks. To create a Playbook, you will need Azure Sentinel Contributor and Logic App Contributor privileges. For this scenario, Contoso wants to send an email to a distribution list that alerts recipients if a VM has been deleted. Follow these steps to create a Playbook that will be used for this automation:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **sentinel**, and under **Services**, click **Azure Sentinel**.
3. On the Azure Sentinel workspaces page, click the workspace that you want to use with Azure Sentinel; the **Azure Sentinel | Overview** page appears.
4. On the left navigation pane, under **Configuration**, click **Playbooks**.
5. Click the **Add Playbook** button; the **Logic App** page appears, as shown in Figure 3-54.

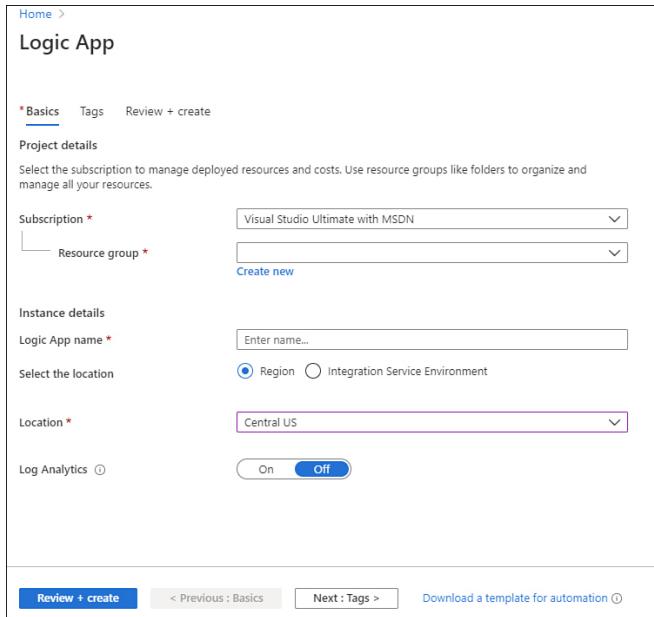


Figure 3-54 Configuring a Logic App

6. Select the subscription and resource group where the Logic App will be located.
7. In the **Logic App Name** field, type a name for this automation.
8. In the **Location** field, select the Azure location where this Logic App will reside.
9. Optionally, you can push the Logic App runtime events to a Log Analytics workspace. For this scenario, leave the default selection and click the **Review + Create** button.
10. On the **Review + Create** tab, click the **Create** button.
11. Click the **Go To Resource** button to open the **Logic Apps Designer** page.
12. Under **Templates**, click the **Blank Logic App** tile.
13. In the **Search Connectors And Triggers** field, type **Azure Sentinel**, and select **When A Response To An Azure Sentinel Alert Is Triggered**.

Important Logic Apps Triggers

Although Logic Apps has many triggers, only Azure Sentinel product-specific triggers may be used when creating your Playbook.

14. Click the **New Step** button; a list of actions appears, as shown in Figure 3-55.

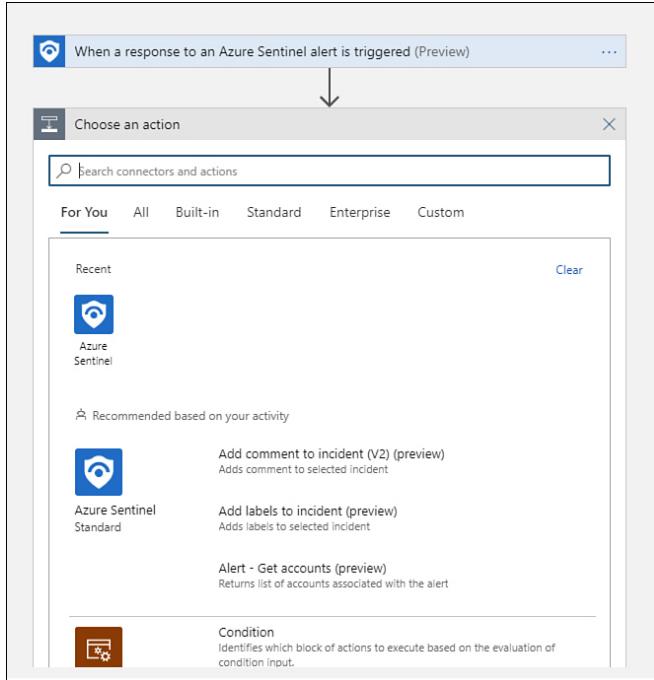


Figure 3-55 Choosing the initial action to be executed

15. In the **Search Connectors And Actions** field, type **email** and select **Office 365**.
16. Select **Send An Email (v2)**; the options shown in Figure 3-56 will appear.

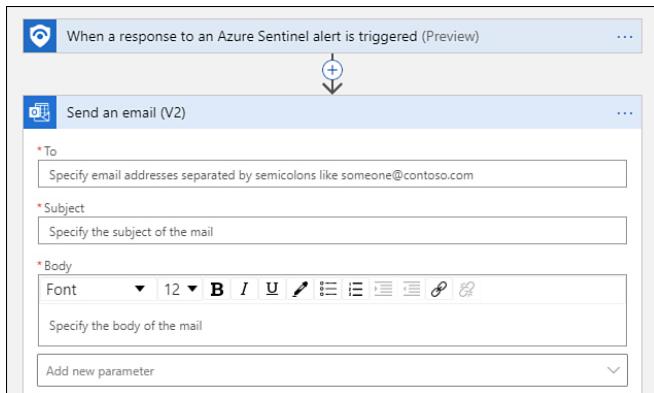


Figure 3-56 Entering the email options

17. Enter the **To** and **Subject** parameters for this email.
18. Click the **Body** field and click **Add Dynamic Content**; a floating menu containing the options to add dynamic content appears, as shown in Figure 3-57.

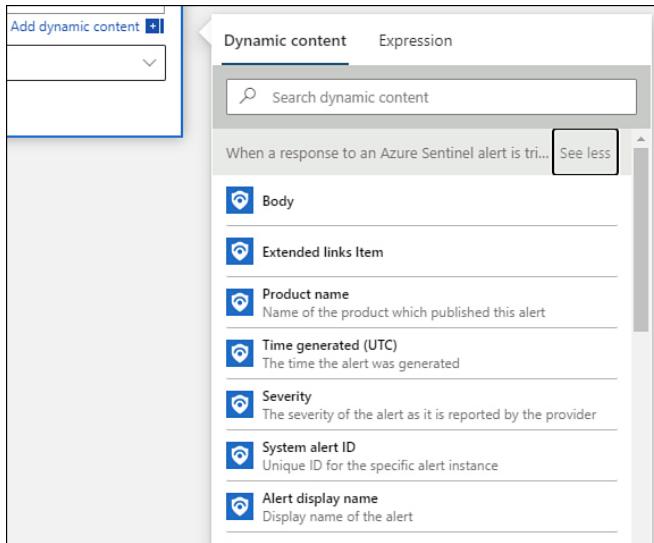


Figure 3-57 Dynamic Content options

19. You can select any dynamic content that you want to add to the body of the email. This helps to enrich the email content by adding alert-related information. For example, you could enter **Alert Severity:** and add the **Severity** field from the dynamic content next to the text.
20. Once you finish adding the dynamic content, click the **Save** button.
21. Close the Logic Apps Designer.
22. Open Azure Sentinel again and click **Analytics**.
23. Click the analytic rule that you created, and on the right side, click the **Edit** button.
24. Click the **Automated Response** tab and notice that the Logic App that you created appears in the list, as shown in Figure 3-58.

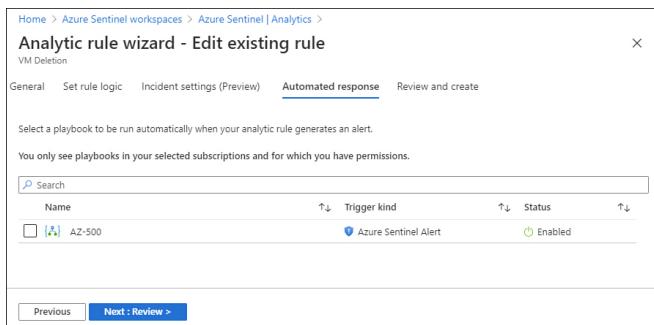
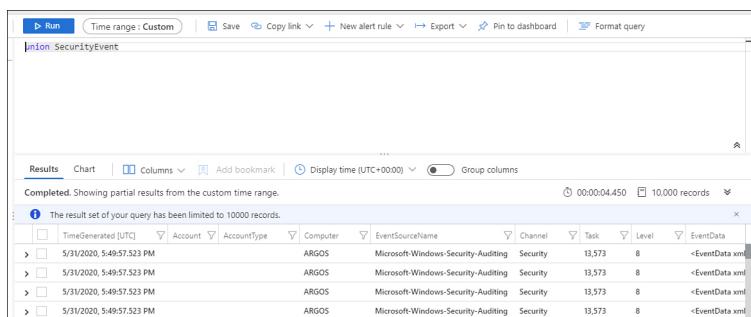


Figure 3-58 Playbook selection for an existing rule

25. Select the Playbook that you created and click the **Next : Review >** button.
26. Click the **Save** button.

Evaluate results from Azure Sentinel

Besides the main overview dashboard available in Azure Sentinel that brings charts and a summary of how the events and alerts, you can also perform direct queries in the Log Analytics workspace or visualize the collected data using Workbooks. If you need to quickly visualize security events, you just need to click the **SecurityEvent** option in the **Events And Alerts Over Time** tile; the Log Analytics workspace appears with the query result, as shown in Figure 3-59.



The screenshot shows the Azure Log Analytics workspace interface. At the top, there are buttons for 'Run' (with a 'Time range : Custom' dropdown), 'Save', 'Copy link', 'New alert rule', 'Export', 'Pin to dashboard', and 'Format query'. Below this is a search bar containing the query: 'union SecurityEvent'. The main area displays a table titled 'Results' with the following columns: TimeGenerated [UTC], Account, AccountType, Computer, EventSourceName, Channel, Task, Level, and EventData. A note at the top of the table says 'Completed. Showing partial results from the custom time range.' and 'The result set of your query has been limited to 10000 records.' The table contains four rows of data, all from 'ARGOS' with 'Microsoft-Windows-Security-Auditing' as the EventSourceName and 'Security' as the Channel. The Task column shows values like 13,573 and 8, and the Level column shows values like 8 and 1. The EventData column shows XML snippets starting with '<EventData xml='.

TimeGenerated [UTC]	Account	AccountType	Computer	EventSourceName	Channel	Task	Level	EventData
5/31/2020, 5:49:57.523 PM	ARGOS			Microsoft-Windows-Security-Auditing	Security	13,573	8	<EventData xml='
5/31/2020, 5:49:57.523 PM	ARGOS			Microsoft-Windows-Security-Auditing	Security	13,573	8	<EventData xml='
5/31/2020, 5:49:57.523 PM	ARGOS			Microsoft-Windows-Security-Auditing	Security	13,573	8	<EventData xml='
5/31/2020, 5:49:57.523 PM	ARGOS			Microsoft-Windows-Security-Auditing	Security	13,573	8	<EventData xml='

Figure 3-59 Security Events

When accessing the information directly from the Log Analytics workspace you can leverage KQL search to explore further the information that you are trying to find out. This type of approach to query data freely using the Log Analytics workspace is more used in investigation scenarios (reactive).

Important No Automated Investigation in Sentinel

Although Azure Sentinel has investigation capabilities, it doesn't have automated investigation. This feature is available only in Microsoft Defender Advanced Threat Protection (ATP).

For more proactive scenarios, one option is to use Azure Workbooks. Azure Sentinel Workbooks provide interactive reports that can be used to visualize your security and compliance data. Workbooks combine text, queries, and parameters to make it easy for developers to create mature visualizations, advanced filtering, drill-down capabilities, advanced dashboard navigations, and

more. To leverage a specific Workbook template, you must have at least Workbook Reader or Workbook Contributor permissions on the resource group of the Azure Sentinel workspace.

Using a Workbook is a great choice for monitoring scenarios where you need data visualization through a dashboard with specific analytics for each data source. Another use case scenario is when you want to build your custom dashboard with data coming from multiple data sources.

For example, if you need to evaluate Azure Activity Log data that is being ingested in Azure Sentinel using the **Azure Activity** connector, you can use the **Azure Activity Workbook**. In the main Azure Sentinel dashboard, under **Threat Management**, click **Workbooks**. Next, click the **Azure Activity** option and click the **View Template** button at the right; the Azure Activity Workbook appears, as shown in [Figure 3-60](#).

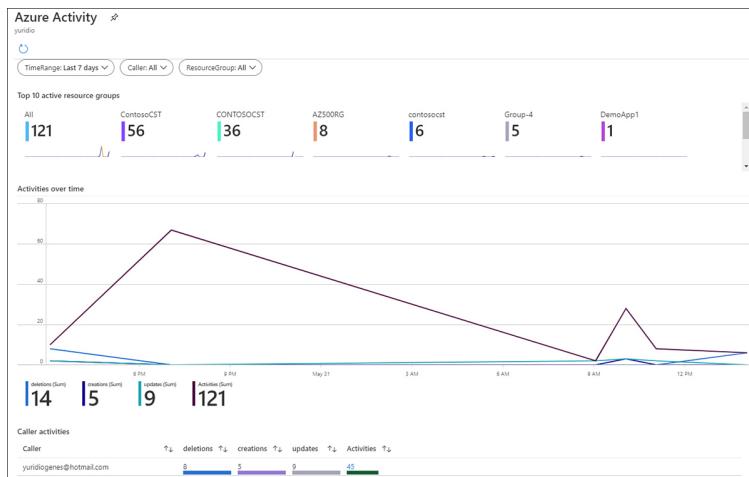


Figure 3-60 Security Events

Leveraging the correct option to evaluate results in Azure Sentinel can help you save time identifying the relevant information.

Incidents

Another way to evaluate results in Azure Sentinel is by looking at incidents. When an incident is created based on an alert that was triggered, you can review this incident in the dashboard, and you can remediate the incident using a Playbook that you previously created. Also, you can investigate the incident.

To access the incidents dashboard, click **Incidents** under the **Threat Management** section on the main Azure Sentinel page. Figure 3-61 shows an example of an incident that was created based on the alert that you created earlier in this chapter.

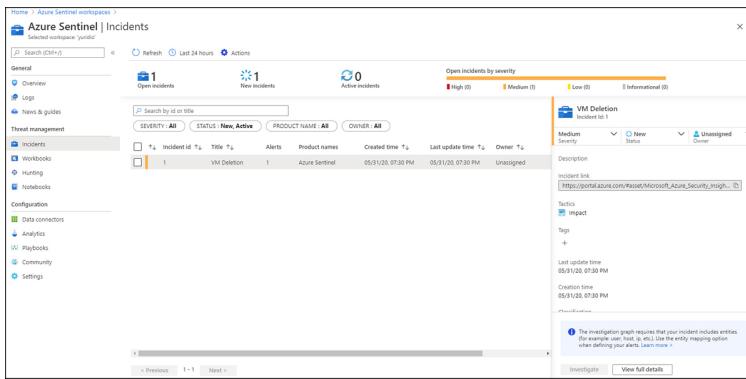


Figure 3-61 Visualizing an incident in Azure Sentinel

When an incident is selected, you will see a summary of the incident details in the right pane. As you triage the incident, you can change the incident's severity, the incident status (for example changing to **Active**, if it is an ongoing investigation), and assign the incident to an owner. (By default, the owner is shown as **Unassigned**.) To see more details about the incident, click the **View Full Details** button. Figure 3-62 shows an example of a full incident.

The screenshot shows the Azure Sentinel Incident view for an incident titled 'VM Deletion'. The left pane displays the incident details, including the severity (Medium), status (New), and owner (Unassigned). It also lists tactics (Imped2), tags, and evidence. The right pane shows a table of alerts with one entry: 'VM Deletion' (Severity: Medium, Alert ID: bed77899-4460-5d89-f545...), associated with 'Azure Sentinel' (Product Name), 5 Events, and a creation time of 05/31/20 07:30 PM. The timeline shows the last update at 05/31/20 07:30 PM and the creation time at 05/31/20 07:30 PM. A note at the bottom of the evidence section states: 'The investigation graph requires that your incident includes entities (for example user, host, ip, etc.). Use the entity mapping option when defining your alerts. Learn more.' A 'Investigate' button is present at the bottom.

Figure 3-62 A full incident

Depending on the artifacts that are available about the incident, you will also have access to the Investigation dashboard. Notice in [Figure 3-62](#), the **Investigate** button is disabled because there is nothing else to investigate on this incident. (Deleting an alert is a single action.)

Threat hunting

Threat hunting is the process of iteratively searching through a variety of data with the objective to identify threats in the systems. Threat hunting involves creating hypothesis about the attackers' behavior and researching the hypotheses and techniques that were used to determine the artifacts that were left behind.

In a scenario in which a Contoso administrator wants to proactively review the data that was collected by Azure Sentinel to identify indications of an attack, the threat hunting capability is the recommended way to accomplish this task. Proactive threat hunting can help to identify sophisticated threat behaviors used by threat actors even when they are still in the early stages of the attack. To access the threat **Hunting** dashboard, click **Hunting** in the **Threat Management** section on the main Azure Sentinel page. [Figure 3-63](#) shows an example of this dashboard.

The screenshot shows the Azure Sentinel Hunting interface. On the left, there's a navigation sidebar with options like General, Threat management, Incidents, Workbooks, Notebooks, Configuration, Data connectors, Analytics, Playbooks, Community, and Settings. The main area has tabs for General, Threat management, Incidents, and Workbooks, with 'Hunting' selected. It displays a list of 'Total queries' (82) and 'My bookmarks' (0). Below this is a search bar and a 'Search queries' input field. A 'FAVORITES' button is highlighted. The main pane shows a table of results with columns for 'Query', 'Provider', 'Data Source', 'Results', and 'Tactics'. One row is expanded to show details about 'Changes made to AWS IAM policy'. This expanded view includes a 'Description' section with text about IAM policy changes, a 'Created time' section with '9/1/2019', and a 'Query' section with a complex PowerShell-like query. At the bottom right of the expanded view is a 'Run Query' button.

Figure 3-63 Hunting capability in Azure Sentinel

To start hunting, you just need to select the predefined query, which was created for a specific scenario, and click the **Run Query** button in the right-hand pane. This pane shows a summary of the results. Click the **View Results** button to see the full details of the query.

SKILL 3.4: CONFIGURE SECURITY POLICIES

While security monitoring is critical for any organization that wants to continue improving their security posture, governance is foundational for any organization that wants to establish standards of deployment and ensure that security is applied in the beginning of the deployment pipeline. This section of the chapter covers the skills necessary to configure security settings using Azure Policy and Azure Blueprint according to the Exam AZ-500 outline.

Configure security settings by using Azure Policy

The first step to achieve governance in Azure is to ensure that you are leveraging Azure Policy for policy enforcement. You can also enforce data residency and sovereignty using Azure Policy. For example, if you need to enforce that all new resources be created to use a specific region, you will use Azure Policy to enforce that. As

mentioned earlier in this chapter, from the centralized management perspective, it's always recommended that you assign a policy to a management group and move the subscriptions that you want to inherit that policy to that management group.

There are many built-in roles grant permission to Azure Policy resources. You can use the Resource Policy Contributor role, which includes most Azure Policy operations. The Owner role has full rights to perform all actions and both Contributor and Reader roles have access to all Azure Policy Read operations. You can use the Contributor role to trigger resource remediation, but you can't use it to create definitions or assignments.

When you are enforcing policies, you need to ensure that your policy initiative is using the right type of effect. If the scenario's requirement is that you avoid certain workloads to be provisioning if certain attributes are not set, your policy effect should be **Deny**. The **Deny** attribute is used to prevent a resource request that doesn't match defined standards through a policy definition and fails the request.

If your scenario's requirement is to change parameters if they were not set during provision time, then your policy effect should be `DeployIfNotExists`. For example, if a Contoso administrator wants to deploy Azure Network Watcher when a virtual network is created, the administrator should enforce the `DeployIfNotExists` effect for that policy. `DeployIfNotExists` runs about 15 minutes after a resource provider has handled a create or update resource request and has returned a success status code. When you configure a policy with this type of effect, you also create a remediation task, and the goal of this remediation task is to configure the resource with the parameter that you want.

Another common scenario is to update tags on a resource during creation or update. For example, Contoso administrator needs to update cost center for all resources during the creation time. For this scenario you need to use the `Modify` effect. Just like the `DeployIfNotExists` effect, you also need to configure a remediation task to run the desire change. Keep in mind that when you are creating this remediation task for both effects, you will need to check the **Create A Managed Identity** option. You can use the identity to authenticate to any service that supports Azure AD authentication—including Key Vault—with any credentials in your code.

Follow the steps below to configure policy enforcement using Azure Policy:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **policy**, and under **Services**, click **Policy**.
3. On the **Policy** page, click **Assignments** under **Authoring** in the left pane. Figure 3-64 shows an example of the Assignments page.

Name	Scope	Type	Policies	Category
Enable Monitoring in Azure Security Center	Teal Management Group	Initiative	103	Security Center
Deploy Diagnostic Settings for Network Security Groups	Visual Studio Ultimate with MSDN/AZS...	Policy	1	Monitoring
ASC DataProtection	Visual Studio Ultimate with MSDN	Initiative	1	Security Center
A security contact email address should be provided for...	Visual Studio Ultimate with MSDN	Policy	1	Security Center

Figure 3-64 Policy assignments page

4. Notice that on this page, you can assign an initiative or a policy. For this example, click the **Assign Policy** button. The **Assign Policy** page appears (see Figure 3-65).

Figure 3-65 Selecting the policy to assign

5. On the **Basics** tab, you have the option to select the **Scope** in which this policy should be assigned. If your scenario requires centralized management, you can change it here to assign to a management group. If the scenario requires that you assign only to the subscription level, then leave the default selection.
6. In the **Exclusion** field, you can optionally select resources that you want to exclude from this policy. For example, if you have certain resource groups that should be exempted from this policy, add those resource groups in this list.
7. In the **Policy Definition** field, click the ellipsis to open the policies that are available.
8. On the **Available Definitions** blade, a list of all policy definitions is shown. For this example, type **SQL** in the **Search** field.
9. Select the **Deploy SQL DB Transparent Data Encryption** policy and click the **Select** button.
10. Notice that both the **Policy Definition** and **Assignment Name** fields have been populated with the name of the policy.
11. Click the **Parameters** tab and notice that for this policy, there are no parameters or effects.
12. Click the **Remediation** tab to configure the additional options.

Figure 3-66 shows the available options.

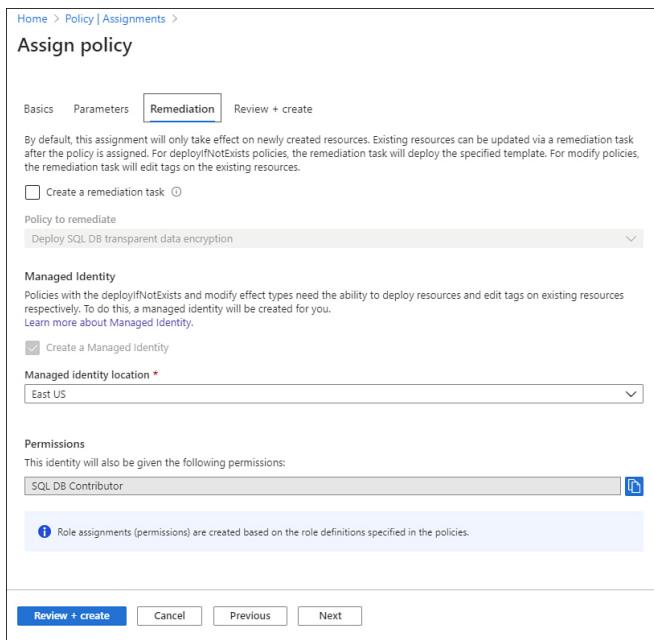


Figure 3-66 Configuring remediation tasks

13. Click the **Create A Remediation Task** check box.
14. The **Policy To Remediate** drop-down menu will automatically select the policy that needs to be used for remediation.
15. Notice that the **Create A Managed Identity** is automatically selected, and the **Managed Identity Location** is also selected.
16. The **Permission** section also automatically shows that the identity that is used will be given the **SQL DB Contributor** permission.
17. Click the **Review + Create** button.
18. Click the **Create** button.

Now that the policy and the remediation task are created, you have the full extent of policy enforcement. You can monitor the compliance of this policy by using the **Overview** dashboard in Azure Policy, and then click the policy to see more details about the assignment. [Figure 3-67](#) shows the **Assignment Details** dashboard.

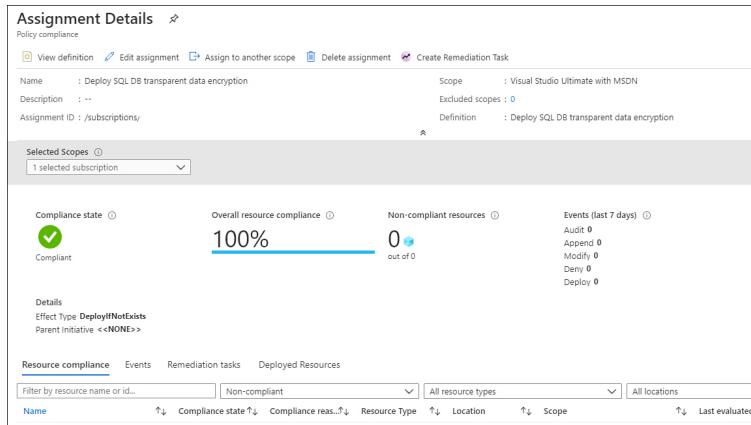


Figure 3-67 Assignment Details dashboard

In Figure 3-67, notice that the **Effect Type** is `DeployIfNotExists`, even though you didn't have to manually set this effect. That's because this policy is already preconfigured with this effect only, and if you open the JSON code for this policy, you will see that this effect is hard coded there.

Configure security settings by using Azure Blueprint

Azure Blueprints enable you to define a repeatable set of Azure resources that implement and adhere to an organization's standards, patterns, and requirements. It is very important for you to understand when to use a blueprint instead of a policy. Blueprints are used to orchestrate the deployment of various resource templates and other artifacts, such as role assignments, policy assignments, Azure Resource Manager templates, and resource groups.

The main difference between a blueprint and a policy is that a blueprint is a package for composing focus-specific sets of standards, patterns, and requirements related to the implementation of Azure cloud services, security, and design. Another characteristic of the blueprint is that you can reuse them to maintain consistency and compliance. A policy can be included in this package as an artifact for the blueprint. Both can be utilized in scenarios where

you have multiple subscriptions and want to maintain governance. From the lifecycle perspective, a blueprint has these major stages:

- **Blueprint creation** This initial step is where you create the blueprint from scratch (blank) or by using a sample.
- **Draft** After creating a new blueprint, the blueprint status changes to `draft`, which means that it was created, but has not been published yet.
- **Published** After finalizing the draft you can publish the first version of the blueprint.
- **Assignment** After a blueprint is published, you can assign it to your subscription.
- **Rewrites** You can change the blueprint versions, which allows you to keep your blueprint up to date.
- **Deletion** If you no longer need a blueprint, you can delete the assignment and then delete the blueprint.

You can create a new blueprint based on your scenario's requirements, or you can create one based on the existing samples available. Follow these steps to create a new blueprint and publish it:

1. Navigate to the Azure portal at <https://portal.azure.com>.
2. In the search bar, type **blueprint**, and under **Services**, click **Blueprints**.
3. On the **Blueprints | Getting Started** page, click the **Create** button in the **Create A Blueprint** section. The **Create Blueprint** page appears, as shown in Figure 3-68.

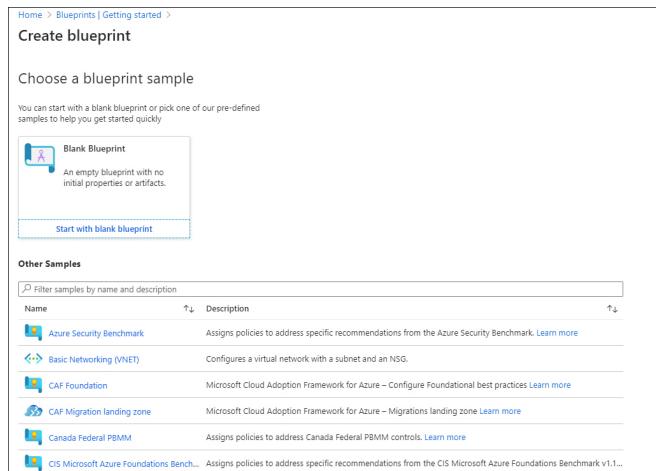


Figure 3-68 Create Blueprint

4. Click the **Start With Blank Blueprint** option; the screen shown in Figure 3-69 appears.

The screenshot shows the 'Create blueprint' interface. At the top, there's a breadcrumb navigation: Home > Blueprints | Getting started >. Below it is the title 'Create blueprint'. There are two tabs: 'Basics' (which is selected) and 'Artifacts'. The 'Blueprint name' field is marked with a red asterisk and contains the placeholder 'Resource name'. The 'Blueprint description' section has a placeholder 'What is the purpose of this blueprint?'. The 'Definition location' section includes a field with a red asterisk and an ellipsis button, with a note explaining that it determines the scope where the blueprint is saved. At the bottom, there are three buttons: 'Save Draft' (gray), 'Discard' (gray), and 'Next : Artifacts >' (blue).

Figure 3-69 Creating a new blank blueprint

5. In the **Blueprint Name** field, type the name of the blueprint.
6. Click the ellipsis in the **Definition Location** option and select the subscription that you want to use for this blueprint.
7. Click the **Next: Artifacts** button.
8. On the **Artifacts** tab, click the **+ Add Artifact** button.
9. On the **Add Artifact** blade, select **Policy Assignment** from the **Artifact Type** drop-down menu.
10. On the **Policy Definitions** tab, select **Deploy Log Analytics Agent For Windows VMs** and click the **Add** button.
11. Click **Add Artifact** again and select **Role Assignment**.
12. In the **Role** drop-down menu, select **Contributor** and click the **Add** button.
13. Click the **Save: Draft** button.
14. On the main **Blueprint** dashboard, click the **Apply** button in the **Apply To A Scope** section.

- 15.** From the **Scope** option, click the ellipsis and select the target subscription. You will see the **Blueprint Definitions** page with the blueprint that you created, which is currently in draft mode, as shown in Figure 3-70.

Blueprint definitions					
+ Create blueprint		Blueprints	Search		
Scope	Visual Studio Ultimate with	All	Search by blueprint name		
Name	Latest Version	Unpublished changes	Last modified	Definition location	
8942500	Draft	Yes	6/1/2020	Visual Studio Ultimate with MSDN	

Figure 3-70 Existing blueprint in draft mode

- 16.** Click the blueprint you created, and from the page that opens, click the **Publish Blueprint** button.
- 17.** In the **Version** field, type a version control for this blueprint. Optionally, you can type a note about the changes in this version in the **Change Notes** field.
- 18.** Click the **Publish** button.

Now that the blueprint is created and published, you can assign it to the subscription. To do that, click the **Assign Blueprint** button in the properties of the blueprint.

Figure 3-71 shows an example of this page.

Assign blueprint

Basics

Subscription(s) Visual Studio Ultimate with MSON

Assignment name *

Location * West US 2

Blueprint definition version * AZ500V1

Lock Assignment

Don't Lock Do Not Delete Read Only

The assignment is not locked. Users, groups, and service principals with permissions can modify and delete deployed resources.
[Learn more](#)

Managed Identity System assigned User assigned

By clicking "Assign" with a system assigned identity, you agree to grant the Azure Blueprints service temporary Owner access to this subscription so that we can properly deploy all Artifacts. We will automatically remove this access when the blueprint assignment process is finished.

Artifact parameters

Artifact / Parameter	Parameter Value
<input checked="" type="checkbox"/> Subscription	
<input checked="" type="checkbox"/> [User group or application name] : Contributor	[User group or application name] ([User group or application name] : Contributor) : <input type="button" value="..."/>
<input checked="" type="checkbox"/> Deploy Log Analytics agent for Windows VMs	Log Analytics workspace (Policy: Deploy Log Analytics agent for Windows VMs) : <input type="button" value="Set value(s)"/>
Optional: List of VM Images that have supported Windows OS to add to scope (Policy: Deploy Log Analytics agent for Windows VMs) : <input type="button" value="..."/>	

Buttons

Figure 3-71 Assign Blueprint

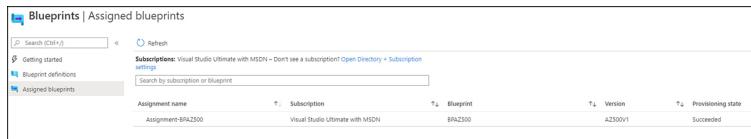
Among those options available in this page, the settings under the **Lock Assignment** section are very important because the selection will depend on the scenario's requirement. The available locks are

- **Don't Lock** This means that resources are not locked by this blueprint. Users, groups, and service principals with permissions can modify and delete deployed resources.
- **Do Not Delete** Although this type of lock is not supported by all resources, this lock allows resources to be modified but not deleted, even by subscription owners. Keep in mind that it might take up to 30 minutes for this blueprint lock to be enforced.
- **Read Only** As the name implies, the resources can't be modified in any way, nor can they be deleted, not even by the subscription

owner. This type of lock is not supported by all resources.

Resource locks deployed by Azure Blueprints are only applied to resources deployed by the blueprint assignment. This means that existing resources, such as those in existing resource groups are not affected since they don't have locks added to them. You can remove locking states by either changing the blueprint assignment's locking mode to **Don't Lock** or by deleting the blueprint assignment.

The **Artifacts Parameters** setting provides an option to type the parameters that were established during the blueprint creation. When you finish filling all those parameters you can click the **Assign** button. When you are finished making assignments, you can see the assignment under Assigned Blueprints in the left navigation pane, as shown in [Figure 3-71](#).

A screenshot of the Azure portal interface titled "Blueprints | Assigned blueprints". The page shows a list of assigned blueprints. There is one item listed: "Assignment-BPAZ300" which is associated with "Visual Studio Ultimate with MSDN" and "Blueprint BPAZ300". The status is "Succeeded".

Assignment name	Subscription	Blueprint	Version	Provisioning state
Assignment-BPAZ300	Visual Studio Ultimate with MSDN	BPAZ300	AZ000V1	Succeeded

Figure 3-72 Assigned Blueprints



Thought experiment

In this thought experiment, demonstrate your skills and knowledge of the topics covered in this chapter. You can find answers to this thought experiment in the next section.

Monitoring Security at Tailwind Traders

You are one of the Azure administrators for Tailwind Traders, an online general store that specializes in a variety of products for the home.

As a part of your duties for Tailwind Traders, you need to work with the Security Operations Center (SOC) to ensure that alerts generated by Azure Security Center are ingested in Azure Sentinel. The SOC Team also needs auditing information about VM creation, and this information needs to be streamed to Azure Sentinel.

Tailwind Traders has been using Azure Security Center Standard tier for a while, primarily to obtain alerts. The company now wants to use other capabilities in Security Center to reduce the attack surface of its IaaS VMs. One of the requirements is to ensure that management ports are closed by default and will only open when an explicit request is made for a specific period of time. Because of some internal auditing, Tailwind Traders database administrators also need to have a vulnerability assessment available for the company's Azure SQL database. With this information in mind, answer the following questions:

- 1.** Which connectors should be used in Azure Sentinel to enable this scenario?
- 2.** Which feature in Azure Security Center will help to reduce the attack surface based on Tailwind Traders' requirements?
- 3.** What needs to be done first before enabling SQL Vulnerability Assessment for Tailwind Traders' databases?

THOUGHT EXPERIMENT ANSWERS

This section contains the solution to the thought experiment.

- 1.** Azure Security Center and Azure Activity Log.
- 2.** Just-in-Time VM Access.

-  First, you need to enable Advanced Data Security (ADS) on your SQL databases.

CHAPTER SUMMARY

- Azure resources logs register operations that were executed at the data plane level, while activity logs at the subscription level register operations that were executed in the management plane.
- You can customize alerts in Azure Monitor for different data types, including metrics, log search queries, and activity logs events.
- Monitoring solutions leverages services in Azure to provide additional insight into the operation of an application or service.
- Azure Security Center Standard tier provides built-in vulnerability assessment using native integration with Qualys.
- To enable vulnerability assessment for SQL, you first need to enable the SQL Advanced Data Security (ADS) feature.
- To implement centralized policy management in Azure Security Center, you should assign the ASC Default initiative to the Management Group level.
- The regulatory compliance dashboard in Azure Security Center can be customized to add other standards that are not available out of the box.
- To ingest data from different data sources into Azure Sentinel, you can use service-to-service connectors or external connectors.
- Azure Blueprints enable you to define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements.

Chapter 4

Secure data and applications

The security of data stored in Azure, the security of SQL, and the security of your secrets, keys, and certificates are as important as the security of any other element of your cloud deployment. One of the most commonly reported cloud data breach types is the storage container full of important customer data that is left open to the world. You've also likely heard of application passwords and connection strings left exposed in source code repositories and SQL database data exfiltrated by clever attackers who leveraged SQL injection vulnerabilities that went undetected until breached data started showing up on the dark web. In this chapter, you will learn how to secure your organization's Azure Storage deployments, the steps that you can take to protect your organization's SQL Server instances, and how to configure and secure Azure Key Vault so that secrets such as connection strings—as well as keys and certificates—can only be accessed by authorized users and applications.

Skills in this chapter:

- [Skill 4.1: Configure security for storage](#)
- [Skill 4.2: Configure security for databases](#)
- [Skill 4.3: Configure and manage Key Vault](#)

SKILL 4.1: CONFIGURE SECURITY FOR STORAGE

Unsecured data storage containers are the source of many data breaches in the cloud. These breaches occur because storage containers that administrators believe

are only accessible to a select group of authorized people are, in fact, configured so that they are accessible to everyone in the world who knows the storage container's address. This objective deals with how to secure storage in Azure, from how to configure access control for storage accounts through how to manage storage account keys. You'll learn about shared access signatures, storage service encryption shared access policies, and using Azure AD to authenticate user access to storage resources in Azure.

Configure access control for storage accounts

Storage accounts are containers for Azure Storage data objects, such as blobs, files, queues, tables, and disks. Azure supports the following types of storage accounts:

- **General-Purpose V2 accounts** Stores blobs, files queues, and tables. Recommended for the majority of storage scenarios. General-Purpose V2 accounts replace General-Purpose V1 accounts, which you should not use for new deployments and should migrate away from if they are used in existing deployments.
- **BlockBlobStorage accounts** Storage accounts recommended for scenarios in which there are high transaction rates for block blobs and append blobs. Also recommended for scenarios that require smaller objects or consistently low storage latency.
- **FileStorage accounts** High-performance, files-only storage accounts. Recommended for high-performance applications.
- **BlobStorage accounts** Legacy storage account type that you should not use for new deployments and should migrate away from if they are used in existing deployments.

The recommended method of managing access control for storage accounts in the management plane is to use RBAC roles. RBAC roles for storage can be assigned at the following levels:

- **Individual container** Role assignments at this scope apply to all blobs in the container. Role assignments also apply to container properties and metadata when the container is accessed at the management plane.

- **Individual queue** Role assignments at this scope apply to all messages in the queue. Role assignments also apply to queue properties and metadata when the queue is accessed at the management plane.
- **Storage account** Role assignments at this scope apply to all containers, all blobs within those containers, all queues, and all messages.
- **Resource group** Role assignments at this scope apply to all storage accounts in the resource group as well as all items within those storage accounts.
- **Subscription** Role assignments at this scope apply to all storage account, in the subscription as well as all items within those storage accounts.
- **Management group** Role assignments at this scope apply to all storage accounts as well as all items within those storage accounts within all subscriptions in the management group.

When assigning an RBAC role, remember the rule of least privilege and assign the role with the narrowest possible scope. This means that if an individual user or application only requires access to a specific storage account and there are multiple storage accounts in a resource group, you should only assign the role at the storage account level. In addition to the rule of least privilege, remember to assign roles to groups rather than individual users. This way, role assignment becomes a matter of adding and removing user accounts from a specific group. Rather than assigning roles to individual users or applications, you should assign the role to a group and then add the user and application accounts to that group as a way of managing the role assignments.

Table 4-1 lists the RBAC roles that are appropriate for storage accounts:

Table 4-1 Storage account RBAC roles

Storage-related RBAC role	RBAC role description
Storage account Contributor	Allows management of storage accounts. Has access to the account key and can access data using Shared Key authorization.
Storage	Can list and regenerate storage account access

account Key Operator Service Role	keys.
Storage Blob Data Contributor	Can read, write, and delete Azure Storage containers and blobs.
Storage Blob Data Owner	Allows full access to Azure software blob containers and data.
Storage Blob Data Reader	Can view and list Azure Storage containers and blobs.
Storage Blob Delegator	Can generate a user delegation key. This key can be used to create a shared access signature for containers or blobs that are signed with Azure AD credentials.
Storage File SMB Share Contributor	Allows read, write, and delete access to files and directories on Azure fileshares.
Storage File Data SMB Share Elevated Contributor	In addition to read, write, and delete access to files and directories on Azure fileshares, modifies the Access Control Lists on files and directories.
Storage File Data SMB Share Reader	Has read only access to files and directories in Azure fileshares.
Storage Queue Data Contributor	Read, write, and delete Azure Storage queues, as well as queue messages.
Storage Queue Data Message Processor	Perform, peek, retrieve, and delete messages from Azure Storage queues.
Storage Queue Data Message Sender	Can add messages to an Azure Storage queue.

Storage Queue Data Reader	Can read and list the contents of an Azure Storage queue and queue messages.
---------------------------	--

To assign a role to a storage account in the Azure portal, perform the following steps:

1. In the Azure portal, open the Storage account for which you want to assign an RBAC role.
2. On the Storage account's page, select **Access Control (IAM)** from the menu, as shown in Figure 4-1.

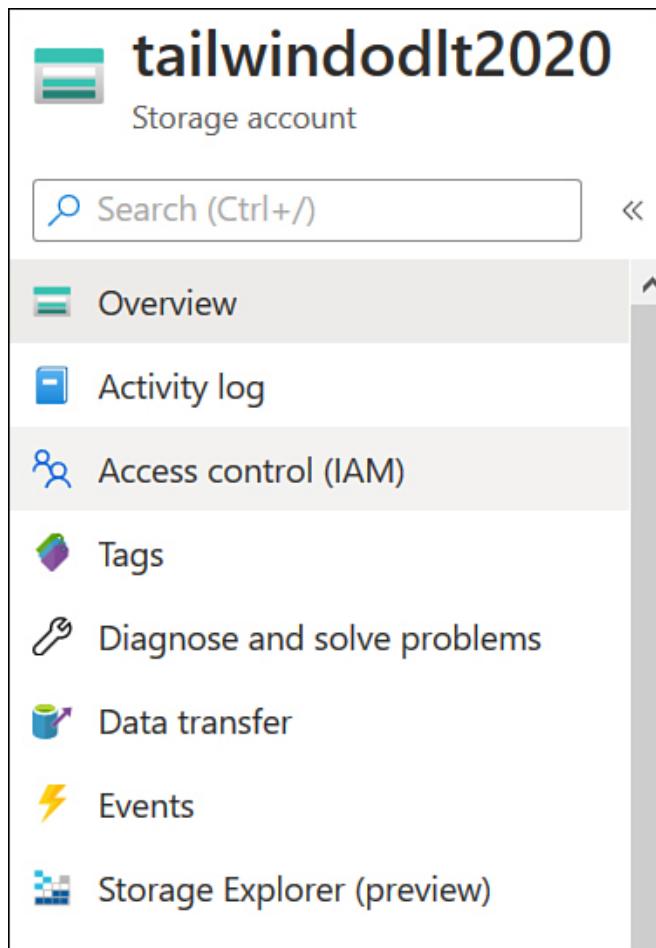


Figure 4-1 Access Control (IAM) node of a storage account

3. On the **Access Control (IAM)** blade, select **Role Assignments** and then click **Add > Role Assignment**, as shown in Figure 4-2. This will bring up the **Add Role Assignment** page.

The screenshot shows the 'tailwinddit2020 | Access control (IAM)' page. The left sidebar includes options like 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), and a search bar. The main area is titled 'Role assignments' with a sub-section 'Number of role assignments for this subscription'. It shows 2 items (2 Service Principals). The table lists:

Name	Type	Role	Scope
Contributor	App	Contributor	Subscription (Inherited)
Reader	App	Reader	Subscription (Inherited)

Figure 4-2 Role Assignments page

- On the **Add Role Assignment** page shown in Figure 4-3, select the security principal—preferably an Azure AD group—to which you want to assign the role and click **Save**.

The dialog has the title 'Add role assignment'. It contains three main sections: 'Role' (set to 'Storage Account Contributor'), 'Assign access to' (set to 'Azure AD user, group, or service principal'), and 'Select' (set to 'TW-Storage-Account-2020-Admins'). A preview section shows a thumbnail of a user icon labeled 'TW' next to the group name.

Figure 4-3 Storage account Contributor role assignment

More Info Rbac Roles for Blob and Queue Data

You can learn more about RBAC role access for blob and queue data at <https://docs.microsoft.com/en-us/azure/storage/common/storage-auth-aad-rbac-portal>.

Configure key management for storage accounts

Storage account access keys allow you to authorize access to storage account data. Each Azure Storage account has an associated pair of 512-bit storage account access keys.

If someone has access to an Azure Storage account key, they have access to the storage account associated with that key. The best practice is to use only the first key and to keep the second key in reserve. You then switch to using the second key when you perform key rotation. This allows you to generate a new primary key, which you will switch to when you perform key rotation in the future. The recommended location for storing storage account access keys is Azure Key Vault. You will learn more about Azure Key Vault later in this chapter.

Because there are only a single pair of access keys associated with a storage account, you should rotate and regenerate access keys on a periodic basis. Rotating storage account access keys ensures that if a storage account key leaks, the leak will be automatically remediated when existing storage account keys reach their end of life. For example, if you rotate keys every six weeks, the maximum amount of time a leaked key remains valid is six weeks. Even if you don't have reason to believe that a storage account key has leaked, the best practice is to rotate them periodically. Just because you don't have reason to believe that a storage account key hasn't leaked doesn't mean that it isn't accessible to someone who shouldn't have access to it.

View storage account access keys

Viewing a storage account access key requires either Service Administrator, Owner, Contributor, or Storage account Key Operator Service roles on the storage account the key is associated with. You can also access the key if you have been assigned an RBAC role that includes the

`Microsoft.Storage/storageAccounts/listkeys`
`/action` permission on a scope that includes the Storage account.

To view a storage account's storage account keys in the Azure portal, perform the following steps:

1. In the Azure portal, navigate to the storage account for which you are interested in learning the storage account access key details.
2. On the Storage account page, select **Access Keys** under **Settings**, as shown in Figure 4-4.

The screenshot shows the Azure Storage account settings page for a storage account named "tailwindodlt2020". The left sidebar lists various navigation options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Data transfer, Events, and Storage Explorer (preview). Below these, the "Settings" section is expanded, showing options for Access keys, Geo-replication, CORS, Configuration, and Encryption. The "Access keys" option is highlighted with a blue border, indicating it is selected.

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Data transfer
- Events
- Storage Explorer (preview)

- Access keys**
- Geo-replication
- CORS
- Configuration
- Encryption

Figure 4-4 Access Keys in the Storage Account keys menu

3. On the **Access Keys** page shown in Figure 4-5, you can view and copy the first and second keys.

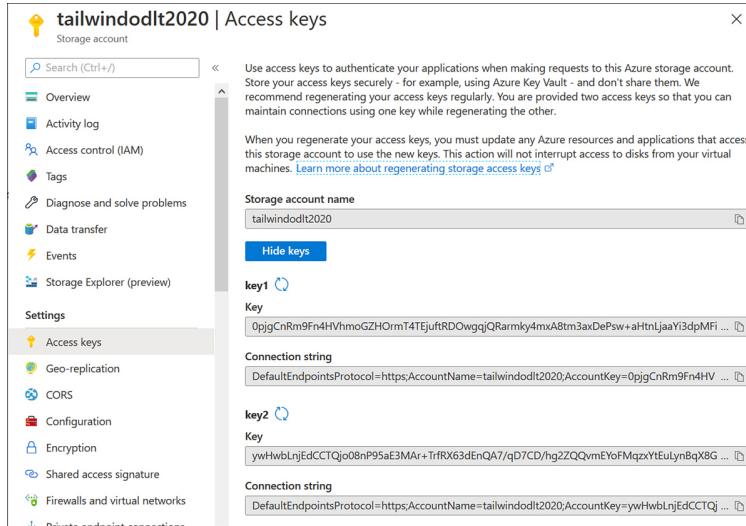


Figure 4-5 Storage account Access Keys

To view the storage account access keys using PowerShell, use the following PowerShell command:

[Click here to view code image](#)

```
$storageAccountKey = ` 
    (Get-AzStorageAccountKey ` 
        -ResourceGroupName <resource-group> ` 
        -Name <storage-account>).Value[0]
```

To view the storage account access keys via Azure CLI, use the following command:

[Click here to view code image](#)

```
az storage account keys list \
    --resource-group <resource-group> \
    --account-name <storage-account>
```

More Info Manage Storage Account Access Keys

You can learn more about managing storage account access keys at <https://docs.microsoft.com/en-us/azure/storage/common/storage-account-keys-manage>.

Manually rotating storage account access keys

Best practice is to rotate storage account access keys on a periodic basis. You should only use one storage account key at a time. Using only one key at a time will allow you to switch any application to the second storage account key of the pair prior to rotating the first. As discussed earlier, after some time has passed, you repeat the process, switching the application to the newly rotated storage account key before then regenerating the second key in the pair. To manually rotate your storage account access keys using the Azure portal, perform the following steps:

1. Ensure that you have updated the connection strings in any application code that reference the storage account access key you will be replacing.
2. Navigate to the **Access Keys** page for the storage account.
3. To regenerate the key, select the Regenerate icon shown in Figure 4-6. This will generate a new storage account access key and connection string. (The regenerate icon appears as a pair of curved arrows.)

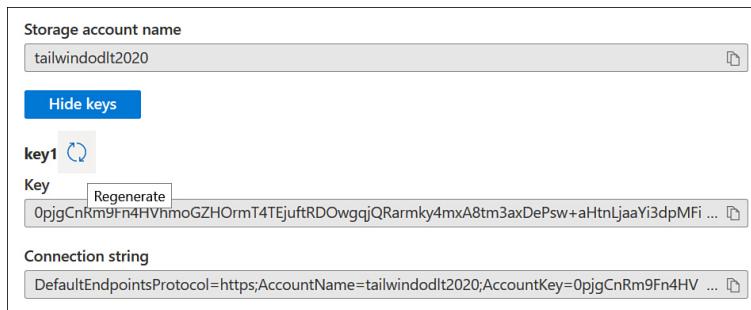


Figure 4-6 The Regenerate icon

To regenerate the storage account key using PowerShell, use the following command, substituting the resource group name and storage account name and either `key1` or `key2`, as appropriate.

[Click here to view code image](#)

```
New-AzStorageAccountKey -ResourceGroupName  
<resource-group>`
```

```
-Name <storage-account> `  
-KeyName key1
```

To regenerate the storage account key using Azure CLI, use the following command, substituting the resource group name and storage account name and specifying whether the key you want to regenerate is the primary or secondary key.

[Click here to view code image](#)

```
az storage account keys renew \  
--resource-group <resource-group> \  
--account-name <storage-account>  
--key primary
```

There are mechanisms that allow you to automate the rotation of storage account access keys. You will learn about these mechanisms later in this chapter.

Create and manage Shared Access Signatures (SAS)

Shared Access Signatures (SAS) allow you to provide secure, granular, and delegated access to storage accounts. Using an SAS, you can control what resources a client can access, the permissions the client has to those resources, and the length of time access will persist. An SAS is a signed Uniform Resource Identifier (URI) that provides the address of one or more storage resources and includes a token that determines how the resource may be accessed by the client.

Azure Storage supports the following types of SAS:

- **User delegation SAS** User delegation SAS can only be used with Blob Storage. User delegation SAS are secured by Azure AD and the permissions configured for the SAS.
- **Service SAS** Service SAS is secured with storage account keys. This SAS delegates access to one type of storage resources. Service SAS can be configured for Azure files, Blob storage, Queue storage, or Table storage.

- **Account SAS** Account SAS is secured with the storage account keys. These keys can be used to delegate access. In addition to all the operations that can be made available using User delegation SAS or Service SAS, Account SAS allows you to delegate access to operations that apply at the service level, such as Get/Set Service Properties. Account SAS also allows you to delegate access to read, write, and delete operations on blob containers, file shares, tables, and queues that are not possible with a Service SAS.

SAS comes in the following two forms:

- **Ad hoc SAS** Ad hoc SAS include the start time, expiry time, and resource permissions within the SAS URI. All SAS types can be ad hoc SAS.
- **Service SAS with stored access policy** Stored access policies are configured on resource containers, which includes blob containers, tables, queues, or fileshares. Service SAS with stored access policies allow the SAS to inherit the start time, expiry time, and permissions that have been configured for the stored access policy.

As is the case with storage account access keys, if an SAS is leaked, anyone who has access to the SAS has access to the storage resources to which the SAS has access.

Application developers should also remember that SAS periodically expire, and if the application is not configured to automatically obtain a new SAS, the application will lose access to the storage resources to which the SAS mediates.

Microsoft has a list of best practices for the use of SAS, which includes

- **Use user delegation SAS when possible** This type of SAS has the best security because it is secured through a user's Azure AD credentials. This means that account keys will not be stored with application code.
- **Be ready to revoke an SAS when necessary** If you determine that an SAS has been compromised, ensure that you can quickly revoke the SAS and replace it with one that is not compromised.
- **Configure stored access policies for service SAS** An advantage of stored access policies is that you can revoke permissions for a service SAS without having to regenerate storage account access keys.
- **Configure short expiration times for ad-hoc SAS** If an ad hoc SAS is compromised, the short expiration time will ensure that the compromised SAS isn't valid for a long time.

- **If necessary, ensure clients renew SAS** If clients regularly make requests to storage using SAS, configure the application so that the client can request SAS renewal before the SAS expires

More Info Shared Access Signatures

You can learn more about Shared Access Signatures at
<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>.

Create user delegation SAS

To create a user delegation SAS for a storage container using PowerShell, first create a storage context object by substituting the appropriate values into the following PowerShell code:

[Click here to view code image](#)

```
$ctx = New-AzStorageContext -StorageAccountName  
<storage-account> -UseConnectedAccount
```

Then create a user delegation SAS token by substituting the appropriate values in the following PowerShell code:

[Click here to view code image](#)

```
New-AzStorageContainerSASToken -Context $ctx `  
-Name <container> `  
-Permission racwdl `  
-ExpiryTime <date-time>
```

To create a user delegation SAS for a blob, substitute the appropriate values in the following PowerShell code:

[Click here to view code image](#)

```
New-AzStorageBlobSASToken -Context $ctx `  
-Container <container> `  
-Blob <blob> `  
-Permission racwd `  
-ExpiryTime <date-time>  
-FullUri
```

You can revoke a user delegation SAS using the `Revoke-AzStorageAccountUserDelegationKeys` command. For example, use the following PowerShell code, substituting the appropriate values where necessary:

[Click here to view code image](#)

```
Revoke-AzStorageAccountUserDelegationKeys -  
ResourceGroupName <resource-group> `  
-StorageAccountName <storage-account>
```

To create a user delegation SAS for a storage container using Azure CLI, run the following Azure CLI command, substituting the appropriate values where necessary:

[Click here to view code image](#)

```
az storage container generate-sas \  
--account-name <storage-account> \  
--name <container> \  
--permissions acdlrw \  
--expiry <date-time> \  
--auth-mode login \  
--as-user
```

To create a user delegation SAS for a blob using Azure CLI, run the following Azure CLI command, substituting the appropriate values where necessary:

[Click here to view code image](#)

```
az storage blob generate-sas \  
--account-name <storage-account> \  
--container-name <container> \  
--name <blob> \  
--permissions acdrw \  
--expiry <date-time> \  
--auth-mode login \  
--as-user  
--full-uri
```

To revoke a user delegation SAS using Azure CLI, run the following command, substituting the appropriate values where necessary:

[Click here to view code image](#)

```
az storage account revoke-delegation-keys \
--name <storage-account> \
--resource-group <resource-group>
```

It is important to note that because user delegation keys and Azure role assignments are cached by Azure Storage, the revocation process might not occur immediately.

More Info Create User Delegation SAS

You can learn more about creating a user delegation SAS at
<https://docs.microsoft.com/en-us/rest/api/storageservices/create-user-delegation-sas>.

Create an account SAS

The first step when creating an account SAS is the creation of an Account SAS URI. The Account SAS URI includes the URI of the storage resource to which the SAS provides access, as well as the SAS token. SAS tokens are special query strings that include the data used to authorize resource requests and to determine the service, resource, and access permissions. SAS tokens also include the period for which the signature will be valid.

Table 4-2 lists the required and optional parameters for the SAS token.

Table 4-2 SAS token parameters

SAS query parameter	Description
Ap-i-ve-rsi-on	Optional Allows you to specify the storage service version to use when executing the request.
Si	Required Specify the signed storage service version to

gn ed Ve rsi on (sv)	authorize requests. Must be configured to 2015-04-05 or later.
Si gn ed Se rvi ce s (ss)	<p>Required Allows you to specify the services accessible with the account SAS. Options include</p> <ul style="list-style-type: none"> • Blob • Queue • Table • File
Si gn ed Re so ur ce Ty pe s (sr t)	<p>Required Allows you to specify which resource types the SAS provides access to. Options include</p> <ul style="list-style-type: none"> • Service Access to service-level APIs. • Container Access to container-level APIs. • Object Access to object-level APIs.
Si gn ed Pe rm iss io n (s p)	<p>Required Permissions for the account SAS. Permissions include</p> <ul style="list-style-type: none"> • Read Valid for all resource types. • Write Valid for all resource types. • Delete Valid for container and object resource types, not including queue messages. • List Valid for service and container resource types.

	<ul style="list-style-type: none"> • Add Valid for queue messages, table entities, and append blobs. • Create Valid for blobs and files. • Update Valid for queue messages and table entities. • Process Only valid for queue messages.
Si gn ed St art (st)	Optional The time at which the SAS becomes valid.
Si gn ed Ex pir y (se)	Required The time at which the SAS becomes invalid.
Si gn ed IP (si p)	Optional Allows you to specify an allowed range of IP addresses.
Si gn ed Pr ot oc ol (s pr)	Optional Determines which protocols are can be used for requests made with the account SAS. Options are both HTTPS and HTTP or HTTPS only.
Si gn	Required Used to authorize the request made with the SAS. Signatures are hash-based message authentication

at ur e (si g)	codes calculated over the string that is signed and the storage account access key using the SHA256 algorithm. This signature is then encoded using Base64 encoding.
----------------------------	--

To construct the signature string, you need to encode the string as UTF-8 that you want to sign from the fields that you have included in the request and compute the signature using the HMAC-SHA256 algorithm.

More Info Create an Account SAS

You can learn more about creating an account SAS at <https://docs.microsoft.com/en-us/rest/api/storageservices/create-account-sas>.

Create a stored access policy for a blob or blob containers

Stored access policies allow you to specifically control service-level shared access signatures. You can configure stored access policies for blob containers, file shares, queues, and tables. Stored access policies consist of the start time, expiry time, and permissions for an SAS. Each of these parameters can be specified on the signature URI rather than in a stored access policy. You can also specify all these parameters on the stored access policy or use a combination of the two. It is important to note that it is not possible to specify the same parameter on both the SAS token and the stored access policy without problems occurring.

Azure allows you to set a maximum of five concurrent access policies on individual containers, tables, queues, or shares. To create or modify a stored access policy, you need to call the `Set ACL` operation for the resource you want to protect with the request body of the call that lists the terms of the access policy. The following is a template that you can use for the request body where you substitute your own start time, expiry time, abbreviated

permission list, and a unique signed identifier of your choosing:

[Click here to view code image](#)

```
<?xml version="1.0" encoding="utf-8"?>
<SignedIdentifiers>
  <SignedIdentifier>
    <Id>unique-64-char-value</Id>
    <AccessPolicy>
      <Start>start-time</Start>
      <Expiry>expiry-time</Expiry>
      <Permission>abbreviated-permission-
list</Permission>
    </AccessPolicy>
  </SignedIdentifier>
</SignedIdentifiers>
```

To change the parameters of an existing stored access policy, call the access control list operation for the resource type and specify new parameters while ensuring that the unique ID field remains the same. To remove all access policies from a storage resource, call the Set ACL operation with an empty request policy.

More Info Stored Access Policies

You can learn more about stored access policies at
<https://docs.microsoft.com/en-us/rest/api/storageservices/define-stored-access-policy>.

Configure Azure AD authentication for Azure Storage

Rather than rely upon storage account keys or shared access signatures, you can use Azure AD to authorize access to Blob and Queue Storage. Azure AD authenticates a security principal's identity and then returns an OAuth 2.0 token. The client includes this token in the request to the Blob or Queue Storage the security principal is accessing. You need to register an application with an Azure AD tenant before tokens can be issued in this manner.

The method that you use to assign specific rights to blob or queue storage is to configure RBAC permissions against the appropriate container, queue, or storage account. You determine what access is required by the user or application, create an Azure AD group, assign the group the appropriate RBAC permission, and then add the user account or service principal to the Azure AD group.

Azure includes the following built-in roles for authorizing access to blob and queue data:

- **Storage Blob Data Owner** Allows the security principal to set ownership and manage POSIX access control for Azure Data Lake Storage Gen2.
- **Storage Blob Data Contributor** Grants the security principal read/write/delete permissions to Blob Storage resources.
- **Storage Blob Data Reader** Allows the security principal to view items in Blob Storage.
- **Storage Blob Delegator** Allows the security principal to acquire the user delegation key, which in turn can be used to create a shared access signature for a container or blob. This shared access signature is signed with the security principal's Azure AD credentials.
- **Storage Queue Data Contributor** Grants the security principal read/write and delete permissions to Azure Storage queues.
- **Storage Queue Data Reader** Allows the security principal to view the messages in Azure Storage queues.
- **Storage Queue Data Message Processor** Allows the security principal to peek, retrieve, and delete messages in Azure Storage queues.
- **Storage Queue Data Message Sender** Allows the security principal to add messages in Azure Storage queues.

More Info Azure AD for Blobs and Queues

You can learn more about Azure AD authorization for blobs and queues at <https://docs.microsoft.com/en-us/azure/storage/common/storage-auth-aad>.

Configure Azure AD Domain Services authentication for Azure Files

When you enable AD DS authentication for Azure Files, your Active Directory Domain Services (AD DS) domain-joined computers can mount Azure File Shares using AD DS user credentials. Access occurs over an encrypted Server Message Block (SMB) protocol connection. You can secure Azure Files using identity-based authentication over Server Message Block (SMB) where either Azure AD DS or an on-premises Active Directory Domain Services Domain (AD DS) Ifunctions as the identity provider. Azure AD Domain Services authentication for Azure Files currently supports the following scenarios:

- If you are using AD DS as your identity provider, you must use Azure AD Connect to synchronize identities to Azure AD.
- If you are using AD DS as your identity provider, you can access the fileshare using a computer that is a member of an AD DS domain. You cannot access the fileshare using a computer that is joined to the Azure AD DS domain.
- If you are using Azure AD DS as an identity provider, you will need to access the file share using a computer that is a member of the Azure AD DS domain.
- When enabled, this form of authentication supports Azure File shares that are integrated with Azure File Sync
- This form of authentication supports single sign-on.
- This form of authentication only supports access from accounts in the AD DS forest in which the storage account is registered unless a specially configured forest trust is present.

Your first step when enabling AD authentication for Azure file shares is to create a storage account that is in a proximate region to the users who will access the files stored in the fileshare on that storage account. You should do this simply because accessing a storage account that is closer to you will provide a much better user experience than trying to open and save files to a fileshare located on the other side of the world. At the start of the process, you won't need to create any fileshares from the storage account. Prior to creating the fileshares, you'll need to enable Active Directory authentication at the storage account level rather than at the individual fileshares level.

Enabling AD DS authentication

The first step when enabling AD DS authentication is to create an identity to represent the storage account in your on-premises Active Directory instance. To do this, first create a new Kerberos key for the storage account using the following Azure PowerShell commands from Cloud Shell:

[Click here to view code image](#)

```
$ResourceGroupName = "<resource-group-name-here>"  
$StorageAccountName = "<storage-account-name-here>"  
New-AzStorageAccountKey -ResourceGroupName  
$ResourceGroupName -Name $StorageAccountName  
-KeyName kerb1  
Get-AzStorageAccountKey -ResourceGroupName  
$ResourceGroupName -Name $StorageAccountName  
-ListKerbKey | where-object{$_._Keyname -contains  
"kerb1"}
```

Once the key has been generated, create a service account in your on-premises domain and configure the account with the following service principal name (SPN):

"cifs/your-storage-account-name-here.file.core.windows.net" using the setspn.exe command. Set the account password to the Kerberos key and configure the account's password to never expire and make a note of the account security identifier (SID). You can use the Get-AdUser PowerShell cmdlet to determine the SID of a user account.

The next step is to use Azure PowerShell to enable Active Directory authentication. You can do this with the following command, substituting the appropriate values:

[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>"
```

You also have the option of using the `AzFilesHybrid` PowerShell module to perform steps similar to these. Using the `AzFilesHybrid` PowerShell module involves downloading the most recent version of the module from Microsoft's website and installing it on a computer that is domain joined and performing the following steps:

1. Change the execution policy to allow the `AzFilesHybrid` PowerShell module to be imported:

[Click here to view code image](#)

```
Set-ExecutionPolicy -ExecutionPolicy  
Unrestricted -Scope CurrentUser
```

2. Switch to the directory where `AzFilesHybrid` has been decompressed and copy the files into your path so that the files can be called directly:

```
.\CopyToPSPPath.ps1
```

3. Import the module into the current PowerShell session:

[Click here to view code image](#)

```
Import-Module -Name AzFilesHybrid
```

4. Initiate a session to your Azure subscription using an Azure AD credential that has either storage account-owner or contributor access to the storage account you created to host the Azure fileshare instance:

Connect-AzAccount

5. Populate the PowerShell session with the appropriate parameter values and then select the appropriate subscription if your account is associated with multiple subscriptions:

[Click here to view code image](#)

```
$SubscriptionId = "<your-subscription-id-  
here>"  
$ResourceGroupName = "<resource-group-  
name-here>"  
$StorageAccountName = "<storage-account-  
name-here>"  
Select-AzSubscription -SubscriptionId  
$SubscriptionId
```

6. The next step involves registering the target storage account with your on-premises AD environment. You should choose an appropriate OU. Use the `Get-ADOrganizationalUnit` cmdlet to determine the name and `DistinguishedName` of the OU that you want to host the registered account:

[Click here to view code image](#)

```
Join-AzStorageAccountForAuth `  
    -ResourceGroupName  
    $ResourceGroupName `  
        -StorageAccountName  
    $StorageAccountName `  
        -DomainAccountType "  
    <ComputerAccount|ServiceLogonAccount>" `  
        -  
        OrganizationalUnitDistinguishedName "<ou-  
distinguishedname-here>" #
```

If you don't provide the OU name as an input parameter, the AD identity that represents the storage account is created under the root directory.

The `Debug-AzStorageAccountAuth` cmdlet allows you to conduct a set of basic checks on your AD configuration with the logged-in AD user once you have performed account registration:

[Click here to view code image](#)

```
Debug-AzStorageAccountAuth -StorageAccountName  
$StorageAccountName -ResourceGroupName  
$ResourceGroupName -Verbose
```

In the event that you are unable to configure the on-premises service account so that its password does not expire, you'll need to use the `Update-AzStorageAccountADObjectPassword` cmdlet to update the Azure Storage account each time your on-premises service account password changes. This cmdlet is a part of the `AzFilesHybrid` module and must be run on a computer in the on-premises AD DS-joined environment with an account that has permissions within AD DS as well as owner permissions to the storage account. The following command—with appropriate variable substitutions—acquires the second storage account key and updates the password of the service account registered in AD DS:

[Click here to view code image](#)

```
# Update the password of the AD DS account  
registered for the storage account  
# You may use either kerb1 or kerb2  
Update-AzStorageAccountADObjectPassword `  
    -RotateToKerbKey kerb2 `  
    -ResourceGroupName "<your-resource-group-  
name-here>" `  
    -StorageAccountName "<your-storage-  
account-name-here>"
```

Configuring share-level permissions

You configure share-level permission by assigning RBAC roles at the Azure fileshare. The following three roles are available for assigning fileshare permissions:

- **Storage File Data SMB Share Reader** This role provides read access to Azure fileshares over SMB to users who have this role.
- **Storage File Data SMB Share Contributor** This role allows users who hold it read, write, and delete access to the Azure Storage fileshares over SMB.

- **Storage File Data SMB Share Elevated Contributor** This role allows read, write, and delete access, as well as the ability to modify Windows Access Control Lists (ACLs) of Azure Storage File Shares over SMB.

When multiple roles are assigned, permissions are cumulative. The exception to this rule is when a deny permission applies; in this case, the deny permission overrides any allow permissions. While it is possible to assign RBAC roles and therefore, configure share-level permissions at the storage account level, you should instead assign RBAC roles at the individual fileshare level. Full administrative control of fileshares, which includes the ability to take ownership of files, currently requires the storage account key. You cannot take ownership of a file using Azure AD credentials.

Configuring file and folder permissions

Once you have assigned share-level permissions to an Azure File Share using RBAC, you should then configure file and folder permissions on the contents of the share. When reading the Azure documentation, most Windows Server administrators will recognize that NTFS permissions are referred to as Windows ACLs.

You can configure file and folder permissions using the `Set-ACL` PowerShell cmdlet, using the `icacls.exe` command or using Windows File Explorer if you have mounted the shared folder on a computer running a Windows Client or Windows Server operating system.

More Info AD Authentication for Azure Files

You can learn more about AD Authentication for Azure Files at
<https://docs.microsoft.com/en-us/azure/storage/files/storage-files-identity-auth-active-directory-domain-service-enable>.

Azure AD DS authentication

Earlier in the chapter, you learned about using on-premises AD DS authentication to secure Azure File

Shares. Also, you can use Azure AD Domain Services to configure authentication for SMB connections to Azure File Shares. Azure AD Domain Services is an Azure service that works with Azure AD to provide the functionality of domain controllers on an Azure subnet. When you enable Azure AD DS, you can domain join a Windows client or server VM that is hosted on an Azure subnet without having to deploy VMs that function as domain controllers. You can't use on-premises Active Directory authentication and Azure AD DS authentication on the same storage account or fileshares.

Once you have enabled Azure AD DS on a subscription, you can enable identity-based access through AD DS when creating the storage account by selecting the **Azure Active Directory Domain Services (Azure AD DS)** identity option. You can also enable this option on the **Configuration** page of the storage account, as shown in Figure 4-7.

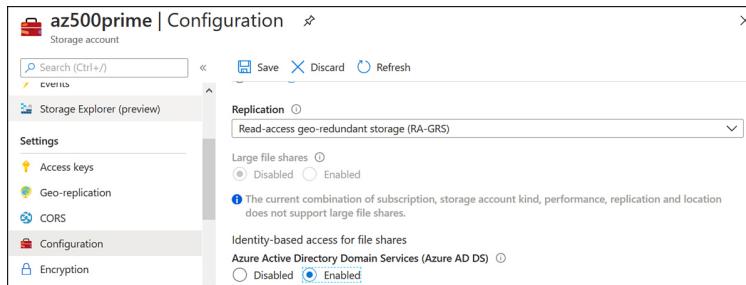


Figure 4-7 Enable Azure AD DS authentication

You can also use the `Set-AzStorageAccount` PowerShell cmdlet with the `EnableAzureActiveDirectoryDomainServicesForFile` parameter to enable Azure AD DS authentication for an Azure fileshare. For example, to enable Azure AD DS authentication for the Azure fileshare named `tailwind-files` stored in the resource group `FilesRG`, run this PowerShell command:

[Click here to view code image](#)

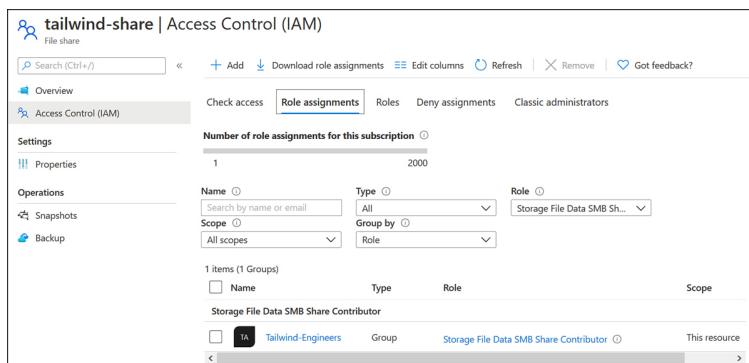
```
Set-AzStorageAccount -ResourceGroupName "FilesRG"
`-Name "tailwind-files"
`-EnableAzureActiveDirectoryDomainServicesForFile
$true
```

You can use the `az storage account update` Azure CLI command with the `--enable-files-adds` option to enable Azure AD DS authentication for an Azure fileshare. For example, to enable Azure AD DS authentication for the Azure fileshare named `tailwind-files` stored in the resource group `FilesRG`, run the Azure CLI command:

[Click here to view code image](#)

```
az storage account update -n tailwind-files -g
FilesRG --enable-files-adds $true
```

Once Azure AD DS authentication has been enabled on the storage account, you can use the **Access Control (IAM)** page of the storage account's properties to assign one of the Storage File Share RBAC roles discussed earlier in this chapter as a share-level permission. Figure 4-8 shows that the Tailwind-Engineers Azure AD group has assigned the Storage File Data SMB Share Contributor role to the `tailwind-share` Azure File share.



The screenshot shows the 'Access Control (IAM)' blade for the 'tailwind-share' storage account. The 'Role assignments' tab is selected. A search bar at the top left is empty. Below it, there are filters for 'Name', 'Type', 'Role', and 'Scope'. The 'Name' filter dropdown shows 'Search by name or email'. The 'Type' dropdown shows 'All'. The 'Role' dropdown shows 'Storage File Data SMB Sh...'. The 'Scope' dropdown shows 'All scopes'. Below these filters, a summary states 'Number of role assignments for this subscription' with a value of '1'. Underneath, a table lists the single assignment: 'Storage File Data SMB Share Contributor' for the 'Tailwind-Engineers' group, which is a 'Group' type with a scope of 'This resource'.

Figure 4-8 Fileshare Role Assignments

The process for configuring NTFS permissions on files and folders is the same as it is when you enable authentication for on-premises AD DS accounts. You first mount the fileshare on a Windows client or server computer, and then you use tools such as Windows File Explorer, PowerShell, or the `icacls.exe` utility to configure the permissions.

More Info Azure AD DS Authentication

You can learn more about Azure AD DS authentication for Azure Files at
<https://docs.microsoft.com/en-us/azure/storage/files/storage-files-identity-auth-active-directory-domain-service-enable>.

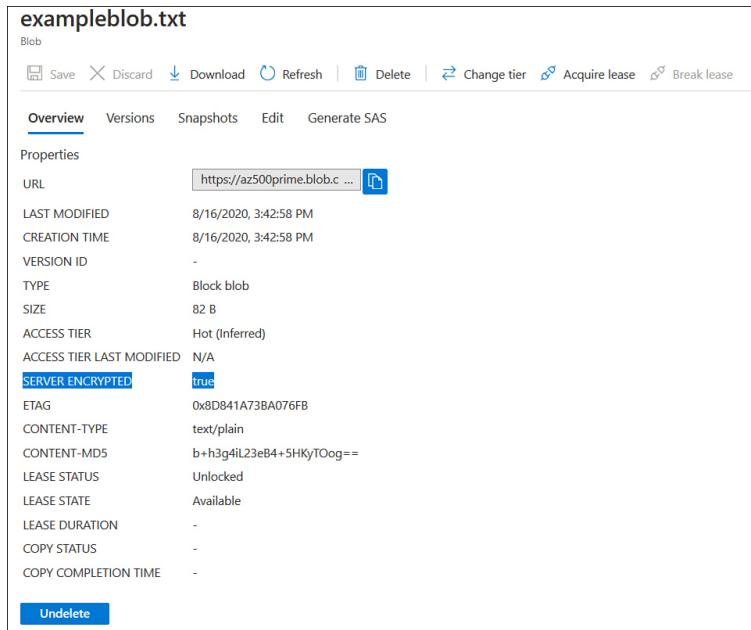
Configure Storage Service Encryption

Azure Storage encryption is enabled by default for all storage accounts regardless of performance or access tiers. This means you don't have to modify code or applications for Azure Storage Encryption to be enabled. Data stored in Azure is transparently encrypted and decrypted using 256-bit AES encryption. You cannot disable Azure Storage encryption, and it isn't necessary to alter code or applications to take advantage of Azure Storage encryption.

Any block blobs, append blobs, or page blobs written to Azure Storage since October 20, 2017, are subject to Azure Storage encryption. Microsoft has undertaken a process where all blobs created prior to this date are being retroactively encrypted. If you are concerned that a blob is not encrypted, you can view that blob's encryption status using the following technique:

1. In the Azure portal, navigate to the storage account you want to check.
2. In the **Containers** section of the Storage account's page, select **Containers** under **Blob Storage** and then locate the container that hosts the blob you are interested in checking. Open that container.
3. In the container you opened, select the blob you want to check.

4. On the **Overview** page, verify that the **Server Encrypted** setting is set to **True**, as shown in Figure 4-9.



The screenshot shows the Azure Storage Blob Overview page for a blob named 'exampleblob.txt'. The 'Properties' section is displayed, including various metadata fields. The 'SERVER ENCRYPTED' field is explicitly highlighted with a blue background, showing the value 'true'. Other visible fields include URL, LAST MODIFIED, CREATION TIME, TYPE (Block blob), SIZE (82 B), ACCESS TIER (Hot (Inferred)), and LEASE STATUS (Unlocked). A large blue 'Undelete' button is located at the bottom left of the page.

Figure 4-9 Verify blob encryption status

You can check the encryption status of a blob using the following PowerShell code, substituting the values in the example code for the values of the blob that you want to check:

[Click here to view code image](#)

```
$account = Get-AzStorageAccount -  
ResourceGroupName <resource-group> `  
-Name <storage-account>  
$blob = Get-AzStorageBlob -Context  
$account.Context `  
-Container <container> `  
-Blob <blob>  
$blob.ICloudBlob.Properties.IsServerEncrypted
```

To check the encryption status of the blob using Azure CLI, use the following command substituting the values in the example code for the values of the blob that you want to check:

[Click here to view code image](#)

```
az storage blob show \
--account-name <storage-account> \
--container-name <container> \
--name <blob> \
--query "properties.serverEncrypted"
```

If you have a blob in Azure that was created prior to October 20, 2017, and which is not encrypted, you can simply rewrite the blob, which will force encryption to occur. One method of doing this is to download the blob to your local file system using AzCopy and then copying the blob back to Azure Storage.

More Info Storage Service Encryption

You can learn more about Storage Service Encryption at <https://docs.microsoft.com/en-us/azure/storage/common/storage-service-encryption>.

Encryption Key Management

By default, Azure Storage accounts encrypt data stored using an encryption key managed by Microsoft. In the event that having Microsoft managing Azure Storage account encryption keys is considered undesirable, you can manage encryption using your own keys, as shown in Figure 4-10.

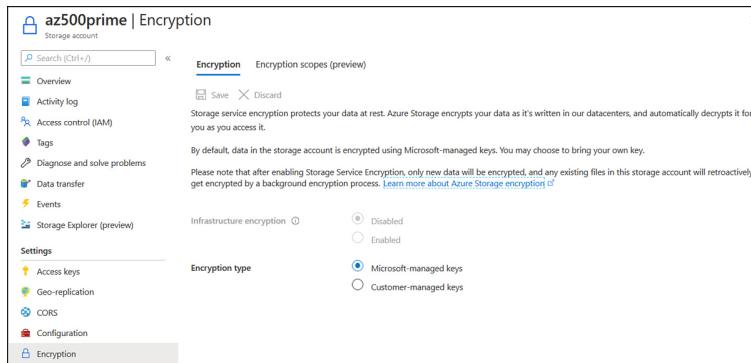


Figure 4-10 Configure encryption type

When you choose the option of managing encryption with keys that you provide; you have the following options:

- **Use a customer-managed key with Azure Key Vault** In this scenario, you upload your encryption key to an Azure Key Vault or use Azure Key Vault APIs to generate keys. The storage account and the Key Vault need to be in the same Azure region and associated with the same Azure AD tenancy. The storage account and Key Vault do not need to be in the same subscription.
- **Use a customer-provided key on Blob Storage operations** In this scenario, encryption keys are provided on a per-request basis. Customer-provided keys can be stored in Azure Key Vault or in an alternate key store.

Infrastructure encryption

As you learned earlier in the chapter, Azure Storage automatically encrypts all data in an Azure Storage account using 256-bit AES encryption. When you enable infrastructure encryption, the data in the storage account will be encrypted twice. Data is first encrypted using one encryption algorithm and one key at the service level and then is encrypted at the infrastructure level using a separate encryption algorithm and encryption key. This double encryption protects data if one of the encryption algorithms or keys becomes compromised. While service level encryption allows you to use either a Microsoft-managed or customer-managed keys, infrastructure-level encryption only uses a Microsoft-managed key. Infrastructure encryption must be enabled during storage account creation. It is not possible to convert an existing storage account to support infrastructure encryption if it was not created with that option enabled.

More Info Storage Account with Infrastructure Encryption

You can learn more about infrastructure encryption for storage accounts at <https://docs.microsoft.com/en-us/azure/storage/common/infrastructure-encryption-enable>.

Encryption scopes

Azure Storage accounts use a single encryption key for all encryption operations across the storage account. Encryption scopes allow you to configure separate encryption keys at the container and blob levels. This

allows for scenarios such as storing customer data from different customers in the same storage account while having each customer's data protected by a different encryption key.

To create a new encryption scope, perform the following steps:

1. In the Azure portal, open the storage account for which you want to configure encryption scopes.
2. On the storage account's page, select **Encryption**, as shown in Figure 4-11 and then select **Encryption Scopes**.

The screenshot shows the Azure Storage account settings for 'az500prime'. The 'Encryption' tab is selected, and the 'Encryption scopes (preview)' sub-tab is active. The main area displays a message 'Showing 0 scopes' and 'No results.' Below this is a table with columns 'Name' and 'Status'. On the left sidebar, under 'Settings', the 'Encryption' option is highlighted. Other settings listed include Access keys, Geo-replication, CORS, Configuration, and another 'Encryption' link.

Figure 4-11 Encryption Scopes

3. On the **Encryption Scope** page, click **Add**.
4. On the **Create Encryption Scope** page, provide an encryption scope name and then specify whether the encryption scope will use Microsoft-managed keys or customer-managed keys, as shown in Figure 4-12.

Create encryption scope

Encryption scope name *

 ✓

Encryption type

Microsoft-managed keys

Customer-managed keys

Create

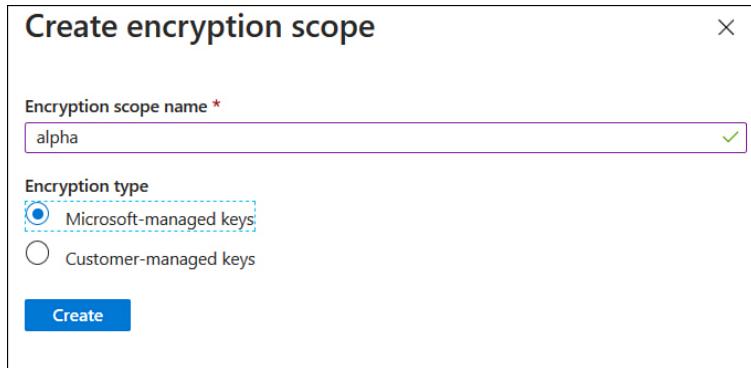


Figure 4-12 Create Encryption Scope

Once you have encryption scopes present for a storage account, you can specify which encryption scope will be used for individual blobs when you create the blob or specify a default encryption scope when you create a container, as shown in Figure 4-13.

New container

Name *

 ✓

Public access level ⓘ

Private (no anonymous access)

Advanced

Encryption scope

alpha

Use this encryption scope for all blobs in the container

Create **Discard**

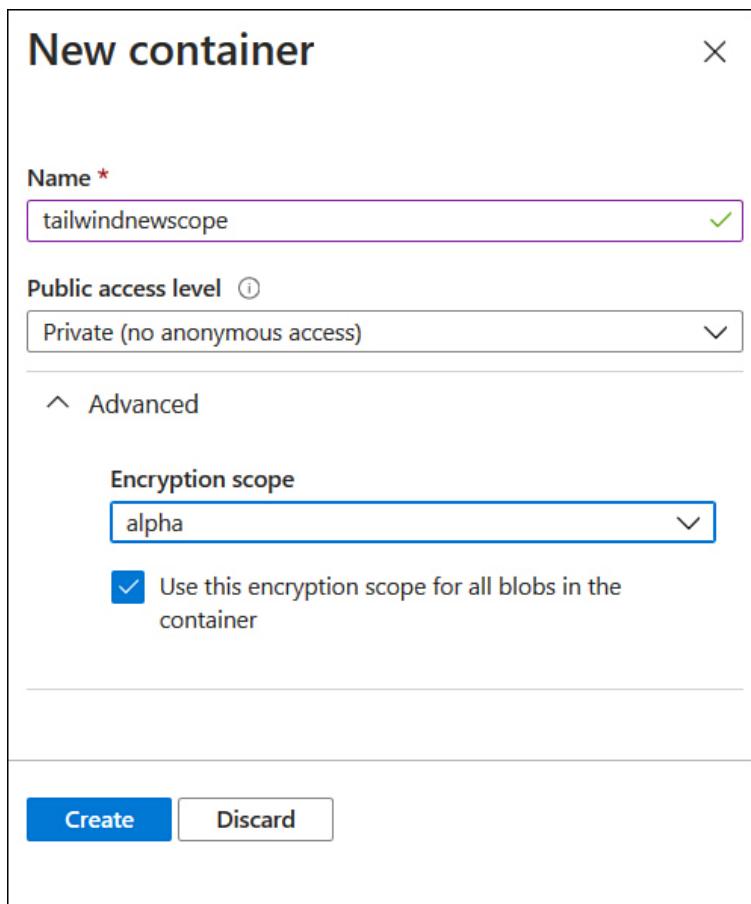


Figure 4-13 New Container Encryption Scope

You can modify the encryption key for an encryption scope by performing the following steps:

1. In the Azure portal, open the storage account for which you want to configure encryption scopes.
2. On the storage account's page, select **Encryption > Encryption Scopes**.
3. Select the **More** button next to the encryption scope for which you want to update the encryption key.
4. On the **Edit Encryption Scope** page shown in Figure 4-14, change the **Encryption Type** and then click **Save**.

The screenshot shows the 'Edit encryption scope - alpha' dialog box. It has a 'Status' section with 'Enabled' selected. Under 'Encryption type', 'Customer-managed keys' is selected. There are dropdown menus for 'Key vault', 'Key', and 'Key Version'. A blue 'Save' button is at the bottom.

Figure 4-14 Edit Encryption Scope

More Info Storage Account Encryption Scopes

You can learn more about storage account encryption scopes at <https://docs.microsoft.com/en-us/azure/storage/blobs/encryption-scope-manage>.

Advanced Threat Protection for Azure Storage

Advanced Threat Protection (ATP) for Azure Storage allows you to detect unusual and malicious attempts to interact with Azure Storage accounts. When you enable ATP for Azure Storage, security alerts will trigger when

Azure detects anomalous storage account activity. These detections are based on existing recognized patterns of malicious activity identified by Microsoft security researchers. These alerts are integrated with Azure Security Center and can also be forwarded by email to administrators of the subscription. The alert information will detail the nature of the suspicious activity as well as provide recommendations on how to further investigate and remediate these issues. Specifically, an Azure Storage ATP alert will inform you of

- The nature of the anomaly
- Storage account name
- Event time
- Storage type
- Probable causes
- Investigation steps
- Remediation steps

ATP for Azure Storage is available for Blob Storage, Azure files, and Azure Data Lake Storage Gen2. General-Purpose V2, block blob, and Blob Storage accounts support this service.

More Info Azure Storage Advanced Threat Protection

You can learn more about Azure Storage Advanced Threat Protection at
<https://docs.microsoft.com/en-us/azure/storage/common/storage-advanced-threat-protection?tabs=azure-security-center>.



Exam Tip

Remember that an account or user delegation SAS will always be an ad-hoc SAS. You can't use stored access policies for the account or user delegation SAS types.

SKILL 4.2: CONFIGURE SECURITY FOR DATABASES

This objective deals with the steps that you can take to secure Azure SQL database instances. To master this objective, you'll need to understand how to configure database authentication, what the options are for database auditing, what the benefits of Azure SQL Database Advanced Threat Protection are, how to configure database encryption, and how to enable Azure SQL Database Always Encrypted on specific database table columns.

Enable database authentication

When you create an Azure SQL database server instance, you create an administrator log in and a password associated with that log in. This administrative account granted full administrative permissions on all databases hosted off the Azure SQL instance as a server-level principal. This login has all the possible permissions on the Azure SQL instance and cannot be limited.

A separate user account called `dbo` is created for the administrator login for each user database. The `dbo` user has all database permissions and is mapped to the `db_owner` database role. You can determine the identity of the administrator account for an Azure SQL database on the **Properties** page of the database in the Azure portal, as shown in [Figure 4-15](#).

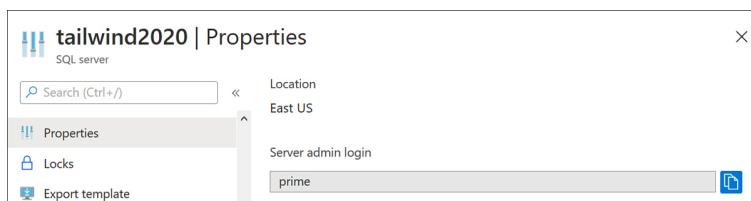


Figure 4-15 Server Admin Login

The admin log-in identifier cannot be changed once the database is created. You can reset the password of this account by selecting the Azure SQL server in the Azure portal and selecting **Reset Password** from the **Overview** page, as shown in Figure 4-16.



Figure 4-16 Reset Password

When adding administrative users, the following options are available:

- You can create an Azure Active Directory Administrator account. If you enable Azure Active Directory authentication, you can configure a user or group account in Azure AD with administrative permissions. You can do this by selecting the **Active Directory Admin** section under the **Azure SQL Instances** setting and then configuring an admin account by clicking the **Set Admin** button (see Figure 4-17).



Figure 4-17 Configuring Active Directory Admin for Azure SQL Server

- Create an additional SQL login in the master database, create a user account associated with this login in the master database, and then add this user account to the `dbmanager` role, the `loginmanager` role, or both roles in the master database using the `ALTER ROLE` statement.

To create additional accounts for nonadministrative users, create SQL logins in the master database and then create user accounts in each database to which the user requires access and associate that user account with the SQL login.

More Info Log Ins, User Accounts, Roles, and Permissions

You can learn more about topic at <https://docs.microsoft.com/en-us/azure/sql/database/logins-create-manage>.

Enable database auditing

Auditing allows you to track database events, such as tables being added or dropped. Audit logs for Azure SQL databases can be stored in an Azure Storage account, in a Log Analytics workspace, or in Event Hubs. Auditing for Azure SQL can be defined at both the server level and the database level. The differences are as follows:

- If you configure a server policy, it will apply to all existing and any newly created databases on the Azure SQL server instance.
- When server auditing is enabled at the instance level, it will always apply to the database.
- Enabling auditing on individual Azure SQL database will not override any server auditing settings, with both audits existing in parallel.
- Microsoft recommends against enabling both server auditing and database blob auditing unless you want to use a different storage account, retention period, or Log Analytics Workspace for a specific database or if you want to audit a separate set of event types of categories for a specific database.

To enable auditing for an SQL instance, perform the following steps:

1. In the Azure portal, open the Azure SQL instance on which you want to configure auditing.
2. Under the **Security** node, select **Auditing**, as shown in Figure 4-18.

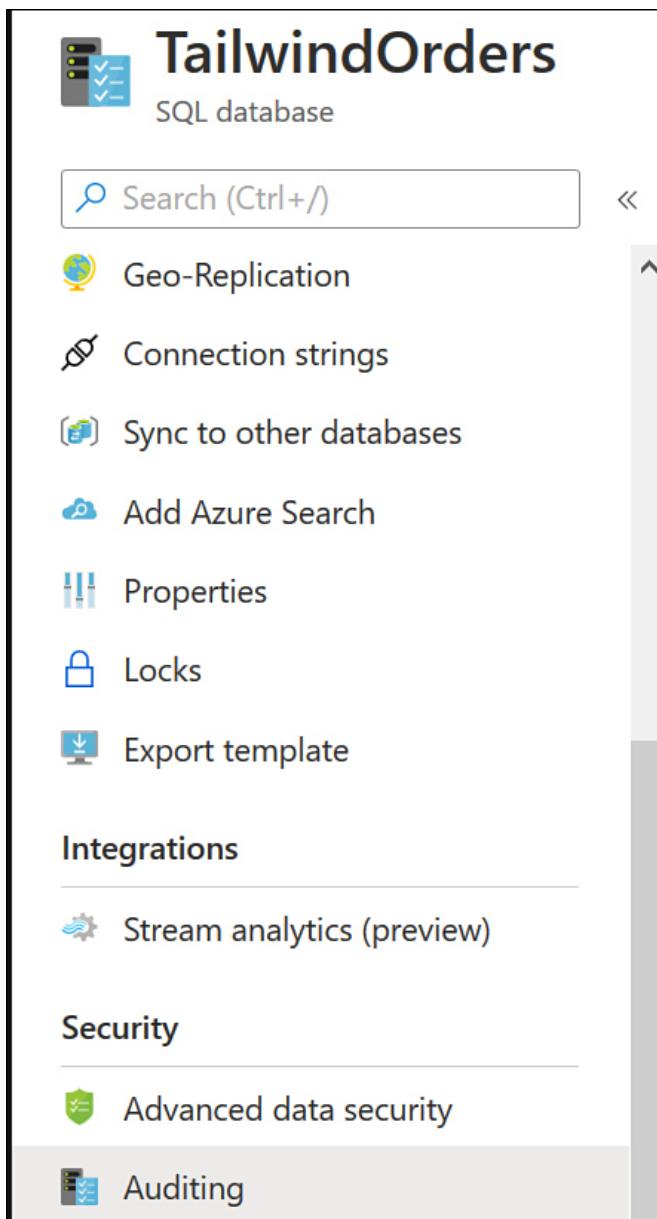


Figure 4-18 Auditing in an Azure SQL Server's properties page

3. Set the **Auditing** slider to **On**, as shown in Figure 4-19. Specify the audit log destination. You can choose between **Storage**, **Log Analytics**, or **Event Hub** and click **Save**.

The screenshot shows the 'Auditing' settings for the 'TailwindOrders' database. At the top, there's a breadcrumb navigation: Home > TailwindOrders (tailwind2020/TailwindOrders) | Auditing >. Below it is a section titled 'Auditing' with a subtitle 'Default settings for all databases on server'. There are three buttons: 'Save', 'Discard', and 'Feedback'. A toggle switch is set to 'ON'. To its right is a link 'Learn more - Getting Started Guide'. Below the switch, there's a section for 'Audit log destination (choose at least one)'. Underneath, there's a checked checkbox for 'Storage' with a sub-section 'Storage details' showing 'tailwindstorage'. Other options like 'Log Analytics (Preview)' and 'Event Hub (Preview)' are also listed.

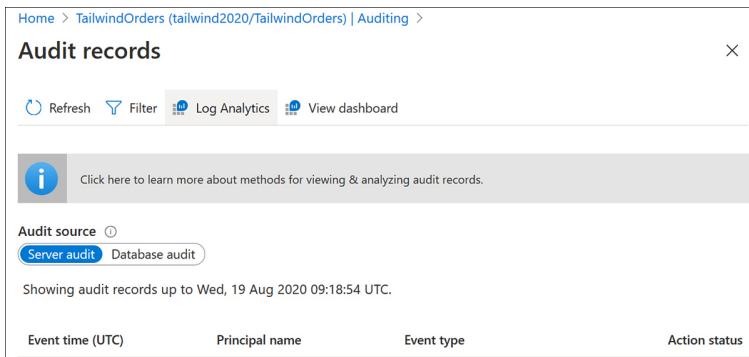
Figure 4-19 Azure SQL auditing settings

You can configure audit logs to be written to Azure Storage accounts, Event Hubs, and Log Analytics workspaces, which can be consumed by Azure Monitor logs. You can choose to have data written to multiple locations should you so choose. When auditing to a storage destination, the retention period is unlimited. You can modify retention settings so that audit logs are kept for a shorter amount of time. **Figure 4-20** shows the **Retention (Days)** setting configured to 14 days.

The screenshot shows the 'Storage settings' configuration page. At the top, there's a breadcrumb navigation: Home > TailwindOrders (tailwind2020/TailwindOrders) | Auditing > Auditing > Storage settings. The main section is titled 'Storage settings' with a close button 'X'. It contains several configuration items: 'Subscription' (set to 'Azure Pass - Sponsorship'), 'Storage account' (set to 'tailwindstorage'), 'Retention (Days)' (set to 14), and 'Storage access key' (with tabs for 'Primary' and 'Secondary'). At the bottom is a blue 'OK' button.

Figure 4-20 Auditing storage retention

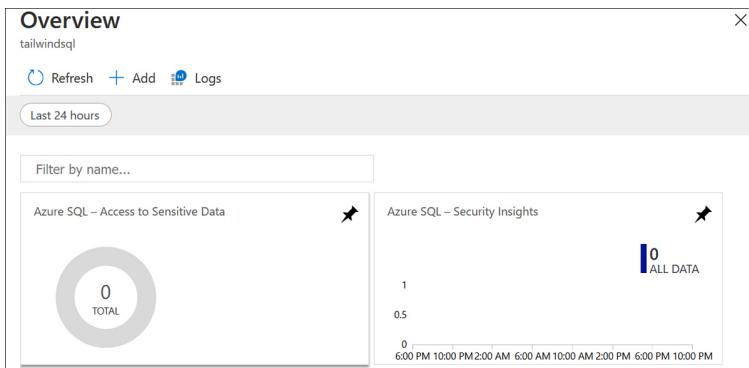
You can view audit logs by clicking on the **View Audit Logs** item from the **Auditing** page of the Azure SQL server's instance. From this page, you can view audit information from the server or database level, as shown in Figure 4-21.



The screenshot shows the 'Audit records' page in the Azure portal. At the top, there are navigation links: Home > TailwindOrders (tailwind2020/TailwindOrders) | Auditing >. Below this is a header with 'Audit records' and a close button. A toolbar contains 'Refresh', 'Filter', 'Log Analytics', and 'View dashboard'. A help section says 'Click here to learn more about methods for viewing & analyzing audit records.' Below the toolbar, 'Audit source' dropdowns show 'Server audit' and 'Database audit' (selected). A message indicates 'Showing audit records up to Wed, 19 Aug 2020 09:18:54 UTC.' The main area has four columns: 'Event time (UTC)', 'Principal name', 'Event type', and 'Action status'. The first row shows a single audit record.

Figure 4-21 Audit records

You also have the option of clicking **Log Analytics** to view the logs in the Log Analytics workspace. If you click **View Dashboard**, you'll be able to view an auditing dashboard that will include access to sensitive data and security insight information, as shown in Figure 4-22.



The screenshot shows the 'Overview' dashboard for the 'tailwindsql' resource. It includes a 'Logs' section with a 'Last 24 hours' button and a 'Logs' link. A 'Filter by name...' input field is present. Two cards are displayed: 'Azure SQL – Access to Sensitive Data' (0 TOTAL) and 'Azure SQL – Security Insights' (0 ALL DATA). The 'Azure SQL – Security Insights' card includes a chart showing data over time from 6:00 PM to 10:00 PM.

Figure 4-22 Auditing dashboard

More Info Auditing for Azure SQL Database
You can learn more about auditing for Azure SQL Database at <https://docs.microsoft.com/en-us/azure/azure-sql/database/auditing-overview>.

Configure Azure SQL Database Advanced Threat Protection

Azure SQL Database Advanced Threat Protection allows you to detect unusual activity that indicates that a third party might be trying to attack your organization's Azure SQL databases. When you enable Azure SQL Database Advanced Threat Protection, you will be notified when unusual database activity occurs, when there are potential database vulnerabilities given the current configuration, and when SQL injection attacks occur.

Azure SQL Database Advanced Threat Protection integrates with Azure Security Center, so you will also be provided with recommendations on how to further investigate and remediate suspicious activity and threats.

To configure Azure SQL Database Advanced Threat Protection, perform the following steps:

1. In the Azure portal, open the Azure SQL Server instance for which you want to configure ATP.
2. Under the **Security** node, click **Advanced Data Security**, as shown in Figure 4-23.

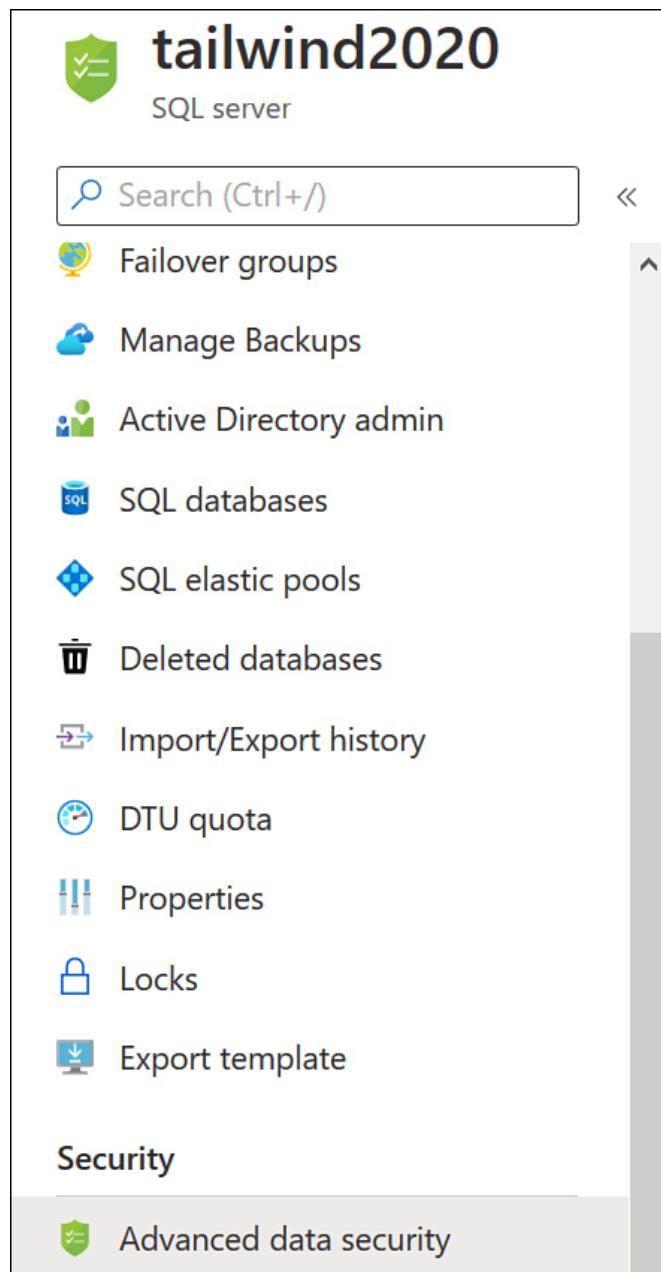


Figure 4-23 Advanced Data Security section

3. On the **Advanced Data Security** page shown in Figure 4-24, configure the following settings:
 1. **Advanced Data Security** This functionality has a per-month cost, which includes Data Discovery, Classification, Vulnerability Assessment, and Advanced Threat Protection. These services allow you to detect data that might be at risk, such as personal data stored within the database, as well as vulnerabilities that might not be detected by other means but which become apparent through analysis of database activity.

2. **Subscription** This setting determines which subscription the vulnerability assessment settings will be billed against.
3. **Storage account** This is where data from assessments will be logged.
4. **Periodic recurring scans** This setting determines whether periodic vulnerability assessment scans are run against the Azure SQL instance. You can specify the email address to which scan reports will be sent.
5. **Advanced Threat Protection Settings** You can configure where advanced threat protection information will be forwarded.

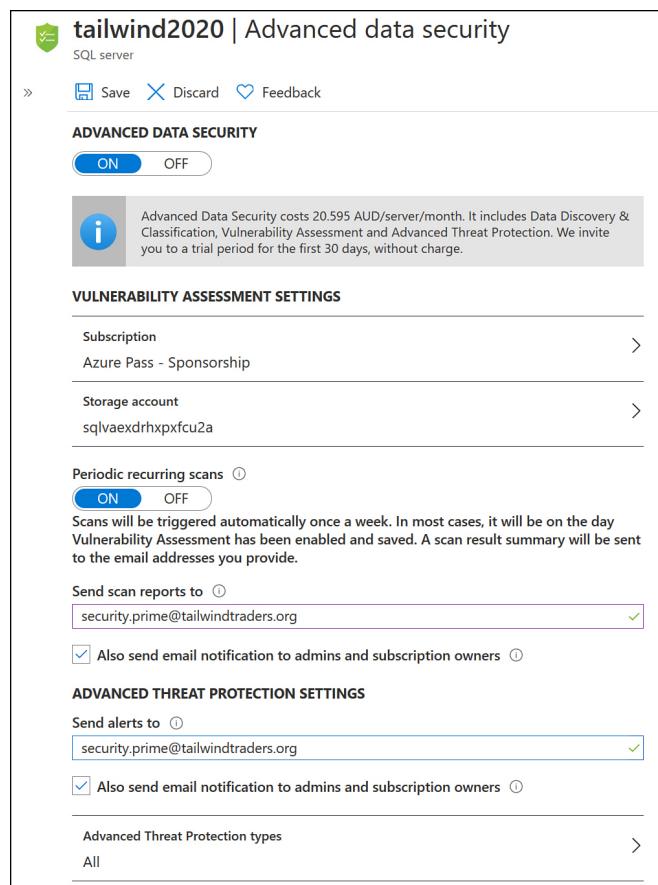


Figure 4-24 Advanced Data Security options

4. On the **Advanced Threat Protection Types** page shown in Figure 4-25, choose which of the following advanced protection alerts you want to be notified about.
 1. **SQL Injection** SQL injection has occurred against a monitored SQL instance.
 2. **SQL Injection Vulnerability** An application vulnerability to SQL injection was detected.

3. **Data Exfiltration** Activity resembling data exfiltration was detected.
4. **Unsafe Action** A potentially unsafe action was detected.
5. **Brute Force** A brute force attack was detected.
6. **Anomalous Client Login** A login with suspicious characteristics was detected.

Home > All resources > tailwind2020 | Advanced data security >

Advanced Threat Protection types

[Learn more - Advanced Threat Protection alerts](#)

- All
- SQL injection ⓘ
- SQL injection vulnerability ⓘ
- Data exfiltration ⓘ
- Unsafe action ⓘ
- Brute Force ⓘ
- Anomalous client login ⓘ

Figure 4-25 Advanced Threat Protection options

More Info Advanced Threat Protection for Azure SQL Database

You can learn more about Azure SQL Database Advanced Threat Protection at <https://docs.microsoft.com/en-us/azure/azure-sql/database/threat-detection-overview>.

Implement database encryption

Transparent data encryption (TDE) allows you to protect Azure SQL databases by encrypting data at rest. When you enable TDS, the databases, associated backups, and transaction log files are automatically encrypted and decrypted, as necessary. TDE is enabled by default for all new Azure SQL Databases. TDE is configured at the server level and is inherited by all databases hosted on the Azure SQL Server instance.

Azure SQL TDE has a database encryption key (DEK) protected by a built-in server certificate that is unique to each Azure SQL instance and leverages the AES 256 encryption algorithm. Microsoft automatically rotates these security certificates.

Customer-managed TDE, also known as “Bring Your Own Key” (BYOK), is supported in Azure SQL. When you configure BYOK, the TDE protection key is stored within Azure Key Vault. When you configure BYOK, you configure an Azure Key Vault with permissions so that the Azure SQL instance can interact with the Key Vault to retrieve the key. If the Key Vault is removed or the Azure SQL instance loses permissions to the Key Vault in a BYOK scenario, the database will be inaccessible.

You can verify that TDE is enabled for an Azure SQL instance by selecting the **Transparent Data Encryption** section of a database server instance’s properties page in the Azure portal, as shown in Figure 4-26.

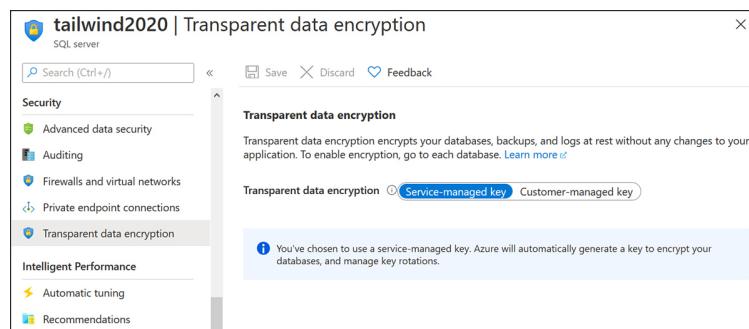


Figure 4-26 TDE Service-Managed Key

If you want to switch to a customer-managed key for an Azure SQL instance, you should first create and configure an Azure Key Vault in the same region as the Azure SQL instance. You can then use the portal to create a key in the Key Vault and configure the Azure SQL instance with the appropriate permissions. To switch a database to a customer-managed key, perform the following steps:

1. On the **Transparent Data Encryption** page of the Azure SQL database instance, select **Customer Managed Key**.
2. The **Key Selection Method** offers two choices: You can choose **Enter A Key Identifier** or you can choose **Select A Key** and then click the **Change Key** link, as shown in Figure 4-27.



Figure 4-27 Configure Customer-Managed Key

3. On the **Select Key From Azure Key Vault** page, select the subscription and the **Key Vault** that will host the key.
4. If no suitable key is present in the Key Vault, you can click **Create New**. This will allow you to create a key, as shown in Figure 4-28.

Create a key

Options

Generate ▼

Name * (i)
AzureSQLBYOK ✓

Key Type (i)
RSA EC

RSA Key Size
2048 3072 4096

Set activation date? (i)

Set expiration date? (i)

Enabled?
Yes No

Create

Figure 4-28 Create a key for BYOK

5. On the **Select Key From Azure Key Vault** page, select the version of the key, as shown in Figure 4-29. If you've just created the key, only the most recent version will be available.

Select key from Azure Key Vault

The key 'AzureSQLBYOK' has been successfully created.

Subscription *	Azure Pass - Sponsorship
Key vault *	KeyVaultODLTMagnus Create new
Key	AzureSQLBYOK Create new
Version ⓘ	2fbfb9f48c974d62afa45f227aff6b2b Create new

Figure 4-29 Selecting a key for BYOK

6. Click **Save** to configure Azure SQL to use your customer key.

More Info Azure SQL Database Encryption

You can learn more about Azure SQL Database encryption at
<https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/sql-server-encryption?view=azuresqldb-current>.

Implement Azure SQL Database Always Encrypted

Always Encrypted is a technology available for Azure SQL that allows you to protect specific types of sensitive data that has a known recognizable pattern, such as passport numbers, tax file identification numbers, and credit card numbers. When Always Encrypted is enabled, clients interacting with the database server will encrypt the sensitive data inside the client applications and will not forward the encryption keys used to decrypt that data to the database server that will store that data. This ensures that administrators who manage Azure SQL servers cannot view sensitive data protected by Always Encrypted.

Deterministic or Randomized Encryption

Always Encrypted supports two forms of encryption: deterministic encryption and randomized encryption:

- **Deterministic encryption** When you use deterministic encryption, the same encrypted value will always be generated for the same plain text value, though this value will be unique to each database. Implementing deterministic encryption will allow you

to perform point lookups, equality joins, grouping and indexing on encrypted columns. It may, however, allow unauthorized users to guess information about encrypted values by looking for patterns in encrypted columns. This is especially true if there are a small set of possible values. Deterministic encryption requires that the column collation is configured with a binary2 sort order for character columns.

- **Randomized encryption** When you configure randomized encryption, data is encrypted in a less predictable manner. While randomized encryption is more secure than deterministic encryption, enabling randomized encryption prevents searching, grouping, indexing, and performing joins on encrypted columns.

In general, you should plan to use deterministic encryption if columns will be used for searchers or where you will be grouping parameters. An example of this is where you need to search for a specific passport number. The client will be able to perform the hash of the query value and then locate values within the database that match that encrypted hash. You should use randomized encryption if your database has information that isn't grouped with other records and which isn't used to join tables, such as medical notes.

Configuring Always Encrypted

Configuring Always Encrypted is an activity that requires use of client-side tools. You can't use Transact SQL statements to configure Always Encrypted and instead must configure Always Encrypted using SQL Server Management Studio or PowerShell. Configuring Always Encrypted requires performing the following tasks:

- Provisioning column master keys, column encryption keys, and encrypted column encryption keys with corresponding column master keys
- Creating key metadata in the database
- Creating new tables with encrypted columns
- Encrypting existing data in selected database columns

Always Encrypted is not supported for columns that have the following characteristics:

- Columns with `xml`, `timestamp/rowversion`, `image`, `ntext`, `text`, `sql_variant`, `hierarchyid`, `geography`, `geometry`, alias types or user-defined types
- `FILESTREAM` columns
- Columns with the `IDENTITY` property
- Columns with `ROWGUIDCOL` property
- String columns with non-`bin2` collections.
- Columns that are keys for clustered and nonclustered indexes (if you are using randomized encryption)
- Columns that are keys for full-text indexes (if you are using randomized encryption)
- Computed columns
- Columns referenced by computed columns
- Sparse column set.
- Columns referenced by statistics (if you are using randomized encryption)
- Columns using alias types
- Partitioning columns
- Columns with default constraints
- Columns referenced by unique constraints (if you are using randomized encryption)
- Primary key columns (if you are using randomized encryption)
- Referencing columns in foreign key constraints
- Columns referenced by check constraints
- Columns tracked using change data capture
- Primary key columns on tables that have change tracking enabled
- Columns masked using Dynamic Data Masking
- Columns in Stretch Database Tables

To configure Always Encrypted on an Azure SQL database using SQL Server Management Studio, perform the following steps:

1. Connect to the database that hosts the tables with columns you want to encrypt using Object Explorer in SQL Server Management Studio. If the database does not already exist, you can create the database and then create the tables that you will configure to use Always Encrypted.
2. Right-click the database, select **Tasks > Encrypt Columns**. This will open the **Always Encrypted Wizard**. Click **Next**.
3. On the **Column Selection** page, expand the databases tables, and then select the columns that you want to encrypt.

4. For each column selected, you will need to set the **Encryption Type** attribute to **Deterministic** or **Randomized**.
5. For each column selected, you will need to choose an **Encryption Key**. If you do not already have an encryption key, you can have one automatically generated.
6. On the **Master Key Configuration** page, choose a location to store the key. You will then need to select a master key source.
7. On the **Validation** page, select whether you want to run the script immediately or use a PowerShell script later.
8. On the **Summary** page, review the selected option and click **Finish**.

More Info Azure SQL Database Always Encrypted

You can learn more about Azure SQL Database Always Encrypted at
<https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine?view=sql-server-ver15>.



Exam Tip

Remember the difference between deterministic and randomized encryption.

SKILL 4.3: CONFIGURE AND MANAGE KEY VAULT

This objective deals with configuring and managing Azure Key Vault, which can be thought of as a cloud hardware security module (HSM). You can use Azure Key Vault to securely store encryption keys and secrets, including certificates, database connection strings, and virtual machine passwords. In this section, you'll learn how to ensure that the items stored in Azure Key Vault are only accessible to authorized applications and users. To master this objective, you'll need to understand how to manage access to Key Vault, including how to configure permissions to secrets, certificates, and keys. You'll also need to understand how to configure RBAC

for managing Key Vault. You'll also need to understand how to manage the items within Key Vault, including how to rotate keys and how to perform backup and recovery on secure Key Vault items.

Manage access to Key Vault

Azure Key Vault allows you to store information that should not be made public, such as secrets, certificates, and keys. Because a Key Vault can store sensitive information, you naturally want to limit who has access to it rather than allowing access to the entire world. You manage Key Vault access at the management plane and at the data plane. The management plane contains the tools you use to manage Key Vault, such as the Azure portal, Azure CLI, and Cloud Shell. When you control access at the management plane, you can configure who can access the contents of the Key Vault at the data plane. From the Key Vault perspective, the data plane involves the items stored within Key Vault, and access permissions allow the ability to add, delete and modify certificates, secrets, and keys. Access to the Key Vault at both the management plane and the data planes should be as restricted as possible. If a user or application doesn't need access to the Key Vault, they shouldn't have access to the Key Vault. Microsoft recommends that you use separate Key Vaults for development, preproduction, and production environments.

Each Key Vault you create is associated with the Azure AD tenancy that is linked to the subscription that hosts the Key Vault. All attempts to manage or retrieve Key Vault content require Azure AD authentication. An advantage of requiring Azure AD authentication is that it allows you to determine which security principal is attempting access. Access to Key Vault cannot be granted based on having access to a secret or key and requires some form of Azure AD identity.

[More Info](#) Key Vault Security

You can learn more about Key Vault Security at
<https://docs.microsoft.com/en-us/azure/key-vault/general/overview-security>.

Key Vault firewalls and virtual networks

The Networking page of a Key Vault's **Networking** page, shown in [Figure 4-30](#), allows you to configure the network locations from which a specific Key Vault can be accessed. You can configure the Key Vault to be accessible from all networks or to specific virtual networks and sets of IPv4 address ranges.

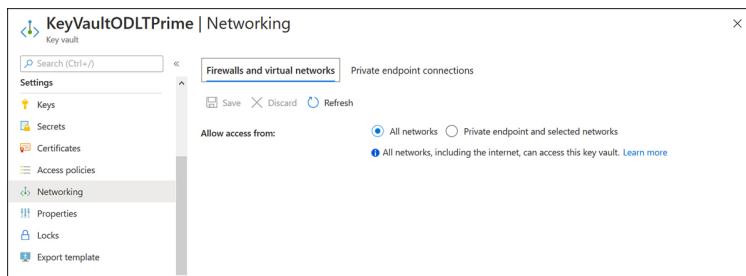


Figure 4-30 Firewalls And Virtual Networks

When configuring network access rules for Azure Key Vault, keep the following in mind:

- Each Key Vault can be configured with a maximum of 127 virtual network rules and 127 IPv4 rules.
- /31 and /32 CIDR subnet masks are not supported. Instead of individual IP addresses, rules should be allowed when allowing access from these subnets.
- IP network rules can only be configured for public IP address ranges. You should use virtual network rules for private IP address ranges.
- IPv6 addresses are not presently supported by Azure Key Vault firewall rules.

You can configure Key Vault firewalls and virtual networks in the Azure portal by performing the following steps:

1. In the Azure portal, open the Key Vault that you want to configure.
2. Under **Settings**, select **Networking**. On the **Networking** page, select **Firewalls And Virtual Networks**.

3. By default, the Key Vault will be accessible from all networks. Select the **Private Endpoint And Selected Networks** option. When you enable this option, trusted Microsoft services can bypass the firewall. You can disable access from trusted Microsoft services if you choose.
4. To add an existing virtual network or a new virtual network, click the **Add Existing Virtual Networks** or **Add New Virtual Networks** items, as shown in Figure 4-31.

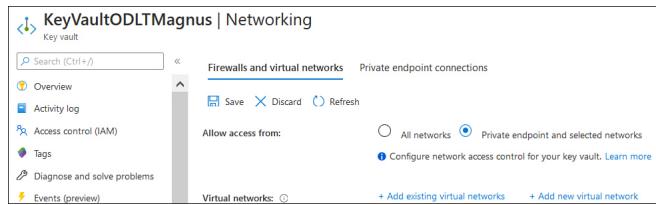


Figure 4-31 Private Endpoint And Selected Networks

5. When you add a virtual network, you must select the subscription, virtual network, and subnets that you want to grant access to the Key Vault, as shown in Figure 4-32. If a service endpoint isn't present on the virtual network subnet, you can enable one.

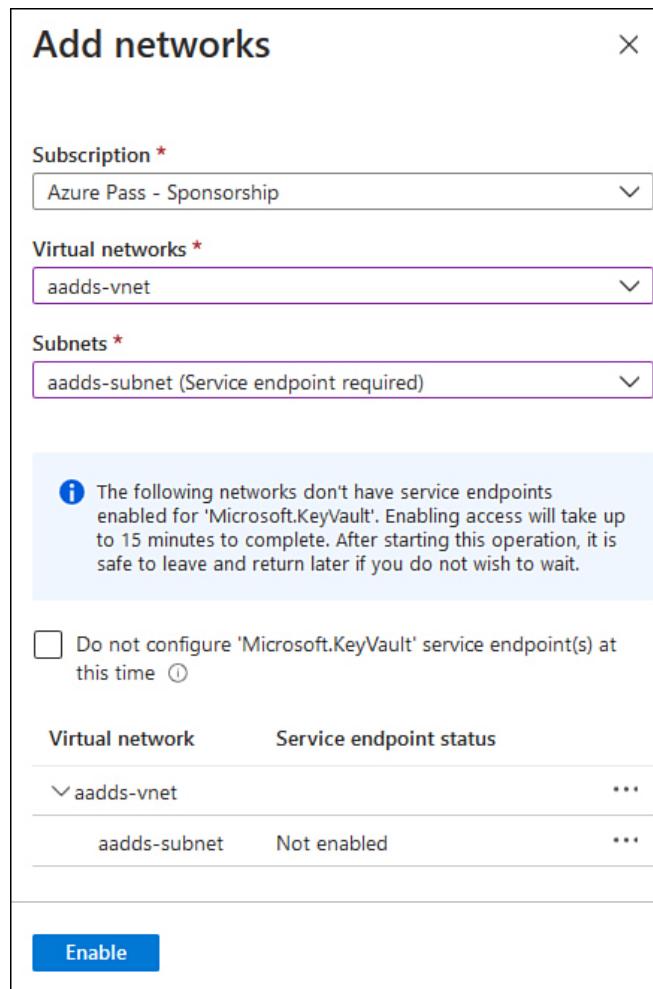


Figure 4-32 Add Networks

- To add an IPv4 address range, enter the IPv4 address or CIDR range, as shown in Figure 4-33.

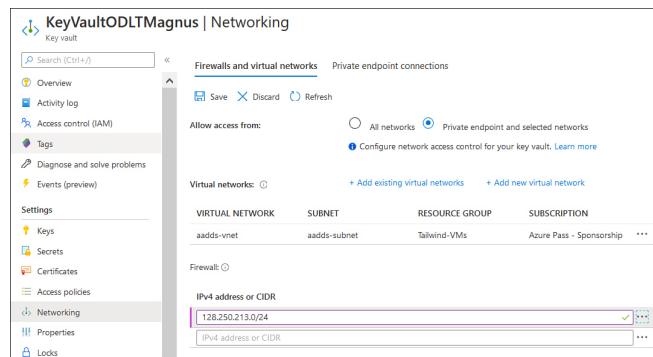


Figure 4-33 Key Vault Firewall.

- Click **Save** to save the **Firewall And Virtual Networks** configuration.

You can use the **Private Endpoint Connections** tab to add private endpoint access to a specific Key Vault. An Azure Private Endpoint is a network interface that allows a private and secure connection to a service using an Azure Private Link. Azure Private Link allows access to Azure PaaS Services, such as Azure Key Vault over a private connection on the Microsoft network backbone. No traffic that traverses a private link passes across the public Internet.

More Info Key Vault Firewalls and Virtual Networks

You can learn more about Key Vault firewalls and virtual networks at <https://docs.microsoft.com/en-us/azure/key-vault/general/network-security>.

Manage permissions to secrets, certificates, and keys

You use Key Vault access control policies to manage permissions to secrets, certificates, and keys at the data plane level. Each Key Vault access control policy includes entries specifying what access the designated security principal has to keys, secrets, and certificates. Each Key Vault supports a maximum of 1,024 access policy entries.

An access policy entry grants a distinct set of permissions to a security principal. A security principal can be a user, service principal, managed identity, or group. Microsoft recommends assigning permissions to groups and then adding and removing users, service principals, and managed identities to and from those groups as a way of granting or revoking permissions.

You can configure the permissions for the keys, secrets, and certificates outlined in Table 4-3.

Table 4-3 Key vault permissions

Certificate permissions	Key permissions	Secrets permissions
-------------------------	-----------------	---------------------

Get View the current certificate version in the Key Vault.	Decrypt Perform a decryption operation with the key.	Get Read a secret.
List List current certificates and certificate versions in the Key Vault.	Encrypt Perform an encryption operation with the key.	List List secrets or secret versions.
Delete Delete a certificate from the Key Vault.		Set Create a secret.
Create Create a Key Vault certificate.	UnwrapKey Use the key for key decryption.	Delete Delete a secret.
Import Import certificate material into a Key Vault certificate.	WrapKey Use the key for key encryption.	Backup Back up secret in a Key Vault.
Update Update a certificate in Key Vault.	Verify Use the key to verify a signature.	Restore Restore a backed-up secret to a Key Vault.
Managecontacts Manage Key Vault certificate contacts.	Sign Use the key for signing operation.	Recover Recover a deleted secret.
Getissuers View the certificate's issuing authority	Get Read the public parts of a key.	Purge Permanently delete a deleted secret.
Listissuers List a certificate's issuing authority information.	List List all keys in the vault.	
Setissuers Update a Key Vault certificate authority or issuers.	Update Modify the key's attributes/meta data.	
Deleteissuers Remove information about a Key Vault's certificate authorities or issuers.	Create Create a key in a Key Vault.	
Manageissuers Manage a Key Vault's list of certificate authorities/issuers.	Import Import an existing key	

Recover Recover a certificate that has been deleted from a Key Vault.	into a Key Vault.	
Backup Back up a certificate stored in Key Vault.	Delete Remove a key from a Key Vault.	
Restore Restore a backed-up Key Vault certificate.	Backup Export a key in protected form.	
Purge Permanently delete a deleted certificate.	Restore Import a previously backed up key.	
	Recover Recover a deleted key.	
	Purge Permanently delete a deleted key.	

Key Vault access policies don't allow you to configure granular access to specific keys, secrets, or certificates. You can only assign a set of permissions at the keys, secrets, or certificates levels. If you need to allow a specific security principal access to only some and not all keys, secrets, or certificates. Instead, you should store those keys, secrets, or certificates in separate Key Vaults. For example, if there are three secrets that you need to protect using Key Vault, and one user should only have access to two of those secrets, you'll need to store the third of those secrets in a separate Key Vault from the first two.

You use the `Set-AzKeyVaultAccessPolicy` Azure PowerShell to configure a Key Vault policy using Azure PowerShell. When using this cmdlet, the important parameters are the vault name, the resource group name,

the security principal identifier, which can be `UserPrincipalName`, `ObjectID`, `ServicePrincipalName` and then the parameters that define permissions to keys, secrets, and certificates. The `Set-AzKeyVaultAccessPolicy` cmdlet has the following format:

[Click here to view code image](#)

```
Set-AzKeyVaultAccessPolicy -VaultName <your-key-vault-name> -PermissionsToKeys <permissions-to-keys> -PermissionsToSecrets <permissions-to-secrets> -PermissionsToCertificates <permissions-to-certificates> -ObjectId <Id>
```

If you prefer Azure CLI, you can use the `az keyvault set-policy` command to configure access policies to Key Vault items. The `az keyvault set-policy` command has the following format:

[Click here to view code image](#)

```
az keyvault set-policy -n <your-unique-keyvault-name> --spn <ApplicationID-of-your-service-principal> --secret-permissions <secret-permissions> --key-permissions <key-permissions> --certificate-permissions <certificate-permissions>
```

More Info Manage Permissions to Key Vault Items

You can learn more about this topic at <https://docs.microsoft.com/en-us/azure/key-vault/general/group-permissions-for-apps>.

Configure RBAC usage in Azure Key Vault

RBAC allows you to secure Azure Key Vault at the management plane. In mid-2020, Microsoft introduced a new set of RBAC roles that provide a simplified way of assigning permissions to the contents of Key Vaults. Going forward, you should only configure access policies when you need to configure complex permissions that

are not covered by the new RBAC roles. You assign Key Vault RBAC roles on the Access Control (IAM) page of a Key Vault's properties, as shown in Figure 4-34. While you can also assign Key Vault RBAC roles at the resource group, subscription, and management group level, security best practice is to assign roles with the narrowest-possible scope.

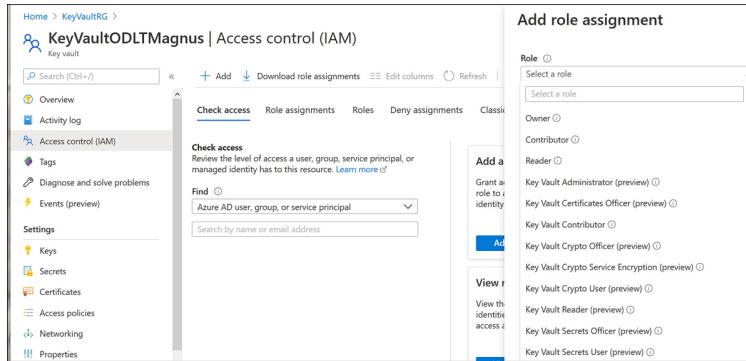


Figure 4-34 Add Role Assignment

The RBAC roles for Azure Key Vault are as follows:

- **Key Vault Administrator** Can perform any action on secrets, certificates, and keys in a Key Vault, except managing permissions
- **Key Vault Certificates Officer** Can perform any actions on Key Vault certificates, except managing permissions
- **Key Vault Contributor** Allows for the management of Key Vault but does not allow access to the items within a Key Vault
- **Key Vault Crypto Officer** Can perform any actions on Key Vault keys, except managing permissions
- **Key Vault Crypto Service Encryption** Has read access to key metadata and can perform wrap and unwrap operations
- **Key Vault Crypto User** Can perform cryptographic operations on keys and certificates
- **Key Vault Reader** Can read Key Vault item metadata but not Key Vault item contents
- **Key Vault Secrets Officer** Can perform all actions on Key Vault secrets except managing permissions
- **Key Vault Secrets User** Can read the contents of secrets

More Info Important Configure RBAC in KEY Vault

You can learn more about this topic at <http://docs.microsoft.com/en-us/azure/key-vault/general/secure-your-key-vault>.

Manage certificates

Azure Key Vault supports the following management actions for x509 certificates:

- Allows for the creation of an x509 certificate or for the import of an x509 certificate
- Supports Certificate Authority-generated certificates and self-signed certificates
- Allows a Key Vault certificate owner to store that certificate securely without requiring access to the private key
- Allows a certificate owner to configure policies that allow Key Vaults to manage certificate life cycles
- Allows certificate owners to provide contact information so that they can be notified about life cycle events, including certificate expiration and renewal
- Can be configured to support automatic certificate renewal with specific Key Vault partner x509 certificate authorities

Certificate policies provide information to the Key Vault on how to create and manage the life cycle of a certificate stored within the Key Vault. This includes information on whether the certificate's private key is exportable.

When you create a certificate in a Key Vault for the first time, a policy must be supplied. Once this policy is established, you won't need the policy for subsequent certificate creation operations. Certificate policies contain the following elements:

- **X509 certificate properties** Includes subject name, subject alternate names, and other properties used during the creation of an x509 certificate.
- **Key properties** Specifies the key type, key length, whether the key is exportable, and how the key should be treated in renewal fields. These properties provide instruction on how a Key Vault generates a certificate key.
- **Secret properties** Specifies secret properties, including the type of content used to generate the secret value, when retrieving a certificate as a Key Vault secret.
- **Lifetime actions** Specifies lifetime settings for the Azure Key Vault certificate. This includes the number of days before expiry and an action option, which either emails specified contacts or triggers autorenewal of the certificate.
- **Issuer** Includes information about the x509 certificate issuer.
- **Policy attributes** Lists attributes associated with the policy.

Azure Key Vault presently can work with two certificate-issuance providers for TLS/SSL certificates: DigiCert and GlobalSign. When you onboard a certificate authority provider, you gain the ability to create TLS/SSL certificates that include the certificate authority provider as the apex of the certificate trust list. This ensures that certificates created through the Azure Key Vault will be trusted by third parties who trust the certificate authority provider.

Certificate contacts information includes the addresses where notifications are sent when specific certificate life cycle events occur. Certificate contact information is shared across all certificates generated by a Key Vault. If you have configured a certificate's policy so that auto-renewal occurs, notifications will be sent

- Prior to certificate renewal.
- After successful certificate auto-renewal.
- If an error occurs during auto-renewal.
- If manual renewal is configured, you are provided with a warning that you should renew the certificate.

More Info Storing X509 Certificates in Key Vault

You can learn more about storing x509 certificates in Key Vault at
<https://docs.microsoft.com/en-us/azure/key-vault/certificates/about-certificates>.

Creating and importing certificates.

You can add certificates to Key Vault by importing them or generating them using the Key Vault. When generating certificates, you can have the certificate self-signed or have it be generated as part of a trust chain from a trusted CA provider.

To create a self-signed certificate using the Azure portal, perform the following steps:

1. In the Azure portal, open the **Key Vault** properties page and click **Certificates**, as shown in Figure 4-35.

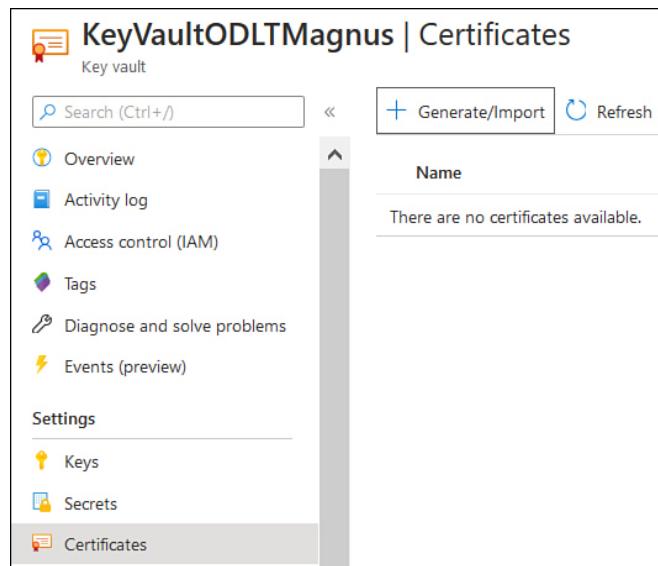


Figure 4-35 Certificates section of Key Vault

2. Select **Generate/Import**. On the **Create A Certificate** page shown in Figure 4-36, set the **Method Of Certificate Creation** as **Generate**. You can also set this to **Import An Existing Certificate**, which you will learn about later in this chapter. Ensure that **Type Of Certificate Authority (CA)** is set to **Self-Signed Certificate**. Provide a **Certificate Name**, a **Subject**, and any **DNS Names**, and then click **Create**.

The screenshot shows the 'Create a certificate' form. The 'Method of Certificate Creation' dropdown is set to 'Generate'. The 'Certificate Name *' field contains 'TailwindWebsite'. The 'Type of Certificate Authority (CA)' dropdown is set to 'Self-signed certificate'. The 'Subject *' field contains 'CN=tailwindtraders.org'. Under 'DNS Names', there is a single entry '1 DNS name'. The 'Validity Period (in months)' input field has '12' entered. The 'Content Type' dropdown shows 'PKCS #12' is selected over 'PEM'. The 'Lifetime Action Type' dropdown is set to 'Automatically renew at a given percentage lifetime'. The 'Percentage Lifetime' slider is set to 80. In the 'Advanced Policy Configuration' section, it says 'Not configured'. At the bottom, there is a blue 'Create' button.

Figure 4-36 Create A Certificate

You can use Azure Key Vault to create TLS/SSL certificates that leverage a trust chain from a trusted CA provider after you have performed the following steps to create an issuer object:

1. Performed the onboarding process with your chosen Certificate Authority (CA) provider. At present, DigiCert and GlobalSign are partnered with Microsoft to support TLS/SSL certificate generation. Certificates generated in this manner will be trusted by third-party clients.
2. The chosen CA provider will provide credentials that can be used by Key Vault to enroll, renew, and implement TLS/SSL certificates. You can enter these credentials on the **Create A Certificate Authority** page in the Azure portal, as shown in Figure 4-37. You get to this page by selecting **Certificate Authorities** on the **Certificates** page of Key Vault and then clicking **Add**.

The screenshot shows the 'Create a certificate authority' dialog box. It has fields for Name, Provider (set to DigiCert), Account ID, Account Password, and Organization ID. The Account ID and Account Password fields are highlighted in red, indicating validation errors: 'The value must be between 1 and 200 characters long.' A 'Create' button at the bottom is blue.

Name	DigiCert	Provider	Account ID	Account Password	Organization ID	Create
<input type="text"/>	<input type="text"/>	DigiCert	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Create"/>

Figure 4-37 Create A Certificate Authority

3. Add the certificate issuer resource to the Key Vault.

4. Configure **Certificate Contacts** for notifications. This step isn't required, but it is recommended. You can do this on the **Certificate Contacts** page, available through the **Certificates** page, as shown in Figure 4-38.

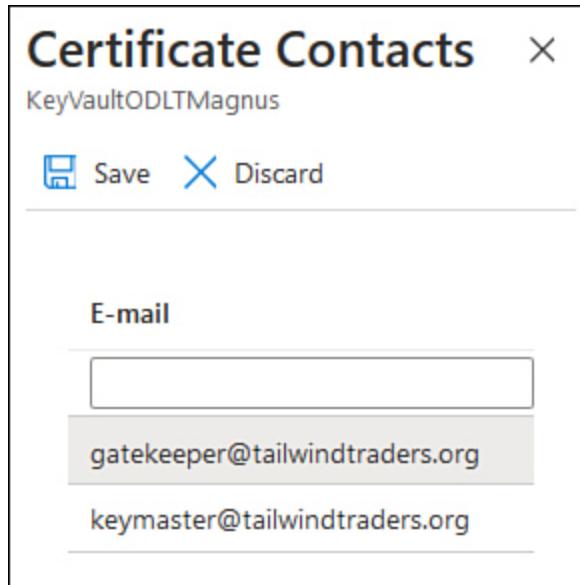


Figure 4-38 Certificate Contacts

Once you have configured the relationship with the issuing CA, you will be able to create TLS/SSL certificates using the portal or by creating a request using JSON code similar to the following. (This requires the `CertificateIssuer` resource created earlier, and this example assumes a partnership with DigiCert.)

[Click here to view code image](#)

```
{
  "policy": {
    "x509_props": {
      "subject": "CN=TailwindCertSubject1"
    },
    "issuer": {
      "name": "mydigicert",
      "cty": "OV-SSL"
    }
  }
}
```

The POST method to send this request URI is similar to the following, with your Key Vault's address substituted

where appropriate:

`https://mykeyvault.vault.azure.net/certificates/mycert1/create?api-version={api-version}.`

To create a Key Vault certificate manually instead of relying on the partner certificate authority provider, use the same method as outlined earlier, but don't include the issuer field. As an alternative, you can create a self-signed certificate by setting the issuer name to "Self" in the certificate policy, as shown here:

```
"issuer": {  
    "name": "Self"  
}
```

You can import an X509 certificate into Key Vault that has been issued by another provider, as long as you have the certificate in PEM or PFX format and you have the certificate's private key. You can perform an import through the Azure portal, as shown in [Figure 4-39](#), by using the `az certificate import` Azure CLI command or by using the `Import-AzKeyVaultCertificate` PowerShell cmdlet.

The screenshot shows the 'Create a certificate' page in the Azure portal. At the top, there is a breadcrumb navigation: Home > KeyVaultRG > KeyVaultODLTMagnus | Certificates >. The main title is 'Create a certificate'. Below the title, there is a section titled 'Method of Certificate Creation' with a dropdown menu set to 'Import'. The next section is 'Certificate Name *' with a help icon (info icon) and an empty input field. The following section is 'Upload Certificate File *' with a 'Select a file' button and a file upload input field. The final section is 'Password' with an empty input field.

Figure 4-39 Import a certificate

You can use the PowerShell cmdlets in Table 4-4 to manage Azure Key Vault certificates.

Table 4-4 PowerShell cmdlets for managing Azure Key Vault certifications

PowerShell cmdlet	Description
Add-AzKeyVaultCertificate	Adds a certificate to Azure Key Vault
Add-AzKeyVaultCertificateContact	Adds a contact for certificate notifications
Backup-AzKeyVaultCertificate	Backs up a certificate already present in an Azure Key Vault
Get-AzKeyVaultCertificate	View a Key Vault certificate
Get-AzKeyVaultCertificateContact	View the contacts registered with the Key Vault for notifications
Get-AzKeyVaultCertificateIssuer	View the certificate issuers configured for a Key Vault
Get-AzKeyVaultCertificateOperation	View the status of any operations in the Key Vault
Get-AzKeyVaultCertificatePolicy	View the policy for certificates in a Key Vault
New-AzVaultCertificateAdministratorDetail	Create an in-memory certificate administrator details object

New-AzKeyVaultCertificateOrganizationDetail	Creates an in-memory organization details object
New-AzKeyVaultCertificatePolicy	Creates an in-memory certificate policy object
Remove-AzKeyVaultCertificate	Removes a certificate from a Key Vault
Remove-AzKeyVaultCertificateContact	Removes a contact registered for Key Vault notifications
Remove-AzKeyVaultCertificateIssuer	Removes a configured issuer certificate authority from a Key Vault
Remove-AzKeyVaultCertificateOperation	Removes an operation that is running in a Key Vault
Restore-AzKeyVaultCertificate	Restores a certificate from backup
Set-AzKeyVaultCertificateIssuer	Configures an issuer certificate authority for a Key Vault
Set-AzKeyVaultCertificatePolicy	Creates or modifies a certificate policy in a Key Vault
Stop-AzKeyVaultCertificateOperation	Cancels a pending operation in a Key Vault
Undo-AzKeyVaultCertificateRemoval	Recover a deleted certificate and places it in an active state

Update-AzKeyVaultCertificate	Modifies editable attributes of a certificate
------------------------------	---

If you prefer to use Azure CLI to manage certificates in Azure Key Vault, you can use the commands shown in [Table 4-5](#).

Table 4-5 Azure CLI commands for managing Azure Key Vault certifications

Command	Description
Az keyvault certificate backup	Back up an x509 certificate in an Azure Key Vault
Az keyvault certificate contact	Manages informational contacts for certificates in an Azure Key Vault
Az keyvault certificate contact add	Adds informational contacts for certificates in an Azure Key Vault
Az keyvault certificate contact delete	Deletes informational contacts for certificates in an Azure Key Vault
Az keyvault certificate contact list	Lists informational contacts for certificates in an Azure Key Vault
Az keyvault certificate create	Creates a certificate in an Azure Key Vault
Az keyvault certificate delete	Deletes a certificate from an Azure Key Vault
Az keyvault certificate download	Downloads the public part of a certificate from an Azure Key Vault

Az keyvault certificate get-default-policy	Enables you to view the properties of the default Key Vault certificate policy
Az keyvault certificate import	Imports a certificate into a Key Vault
Az keyvault certificate issuer	Manages issuer certificate authorities
Az keyvault certificate issuer admin	Manages administrators for issuer certificate authorities
Az keyvault certificate issuer admin add	Enables you to add an administrator for an issuer certificate authority
Az keyvault certificate issuer admin delete	Removes a configured administrator for a specific issuer certificate authority
Az keyvault certificate issuer admin list	Lists the administrators configured for a specific issuer certificate authority
Az keyvault certificate issuer create	Configures an issuer certificate authority for an Azure Key Vault
Az keyvault certificate issuer delete	Deletes an issuer certificate authority from an Azure Key Vault
Az keyvault certificate issuer list	Lists the issuer certificate authorities for a specific Azure Key Vault
Az keyvault certificate issuer show	Enables you to view information about a specific issuer certificate authority
Az keyvault	Updates information about issuer

<code>certificate issuer update</code>	certificate authority
<code>Az keyvault certificate list</code>	Lists certificates in an Azure Key Vault
<code>Az keyvault certificate list-deleted</code>	Enables you to view a list of deleted certificates that can be recovered
<code>Az keyvault certificate list-versions</code>	Enables you to view the versions of a certificate
<code>Az keyvault certificate pending</code>	Manages certificate-creation operations
<code>Az keyvault certificate pending delete</code>	Terminates the pending creation of a certificate
<code>Az keyvault certificate pending merge</code>	Merges a certificate or a certificate chain with a key pair that is present in the Key Vault
<code>Az keyvault certificate pending show</code>	Enables you to views the status of a certificate's creation operation
<code>Az keyvault certificate purge</code>	Permanently deletes a deleted certificate
<code>Az keyvault certificate recover</code>	Recovers a deleted certificate
<code>Az keyvault certificate restore</code>	Restores a backed-up certificate to a Key Vault
<code>Az keyvault certificate set</code>	Updates a certificate's attributes

attributes	
Az keyvault certificate show	Enables you to view certificate information
Az keyvault certificate show-deleted	Enables you to view information on a deleted certificate

More Info Getting Started with Key Vault Certificates

You can learn more getting started with Key Vault certificates at <https://docs.microsoft.com/en-us/azure/key-vault/certificates/certificate-scenarios>.

Manage secrets

Secrets, in the context of Azure KeyVault, allow you to securely store items such as passwords and database connection strings. Key Vault automatically encrypts all stored secrets. This encryption is transparent. The Key Vault will encrypt a secret when you add it, and it decrypts the secret when an authorized user accesses the secret from the vault. Each Key Vault encryption key is unique to an Azure Key Vault.

Key Vault secrets are stored with an identifier and the secret itself. When you want to retrieve the secret, you specify the identifier in the request to the Key Vault. You can add a secret to a Key Vault using the `az keyvault secret set` command. For example, to add a secret to the Key Vault named TailwindKV where the secret identifier name is Alpha and the value of the secret is Omega, you would run this command:

```
az keyvault secret set \
    --name Alpha \
    --value Omega \
    --vault-name TailwindKV
```

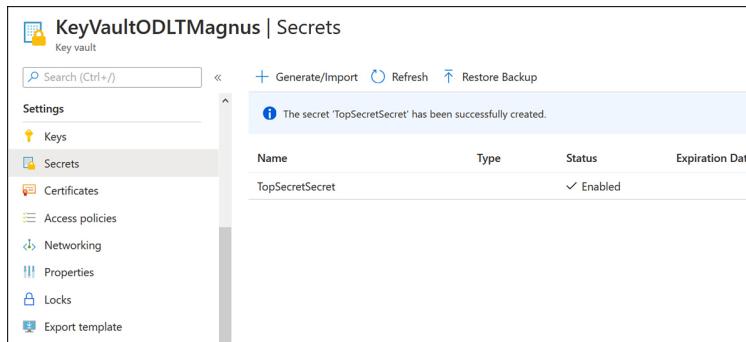
You can view a secret using the `azure keyvault secret show` Azure CLI command, and you can delete a secret using the `azure keyvault secret delete` Azure CLI command. To add the same secret to the same Azure Key Vault used in the earlier example using PowerShell, run the command:

[Click here to view code image](#)

```
$secretvalue = ConvertTo-SecureString 'Omega' -  
AsPlainText -Force  
$secret = Set-AzKeyVaultSecret -VaultName  
'TailwindKV' -Name 'Omega' -SecretValue  
$secretvalue
```

You can view an Azure Key Vault Secret with the `Get-AzureKeyVaultSecret` cmdlet. You can modify an existing Azure Key Vault secret with the `Update-AzureKeyVaultSecret` Azure PowerShell cmdlet, and you can delete an Azure Key Vault secret with the `Remove-AzureKeyVaultSecret` cmdlet.

You can manage secrets using the Azure portal from the **Secrets** section of a Key Vault's properties page, as shown in [Figure 4-40](#).



The screenshot shows the 'Secrets' blade of the Azure Key Vault 'KeyVaultODLTMagnus'. The left sidebar lists 'Settings', 'Keys', 'Secrets' (which is selected), 'Certificates', 'Access policies', 'Networking', 'Properties', 'Locks', and 'Export template'. The main area has a search bar and buttons for 'Generate/Import', 'Refresh', and 'Restore Backup'. A message states: 'The secret 'TopSecretSecret' has been successfully created.' A table displays the secret details:

Name	Type	Status	Expiration Date
TopSecretSecret		✓ Enabled	

Figure 4-40 Key Vault secrets

Beyond the secret ID and the secret itself, you can configure the following attributes for Azure Key Vault secrets.

- **Expiration time (exp)** Allows you to specify a specific time after which the secret should not be retrieved from the Key Vault. Use of this attribute does not block the use of the secret, just as the expiration date on food doesn't stop you from eating it after that date has passed. The expiration time attribute simply provides the secret keeper with a method of recommending that a secret is beyond its use-by date.
- **Not before (nbf)** Similar to the expiration time attribute, the `not before` attribute allows the secret keeper to specify the time at which a secret becomes valid. For example, you could store a secret in a Key Vault and set the `not before` attribute to 2030, which would inform anyone retrieving the secret that the secret information itself won't be useful until 2030.
- **Enabled** Allows you to specify whether secret data is retrievable. This attribute is used in conjunction with the `exp` and `nbf` attributes. Any operation that involves the `Enabled` attribute that doesn't include the `exp` or `nbf` attributes will be disallowed.

You can use the Azure PowerShell cmdlets in Table 4-6 to manage secrets in Azure Key Vault.

Table 4-6 PowerShell cmdlets for managing Key Vault secrets

PowerShell cmdlet	Description
<code>Backup-AzKeyVaultSecret</code>	Securely backs up a Key Vault secret
<code>Get-AzKeyVaultSecret</code>	View the secrets in a Key Vault
<code>Remove-AzKeyVaultSecret</code>	Deletes a Key Vault secret
<code>Restore-AzKeyVaultSecret</code>	Restores a Key Vault secret from a backup
<code>Set-AzKeyVaultSecret</code>	Creates or modifies a secret in a Key Vault
<code>Undo-AzKeyVaultSecretRemoval</code>	Recovers a deleted secret that has not been permanently removed

Update-AzKeyVaultSecret	Updates the attributes of a secret in a Key Vault
-------------------------	---

You can use the Azure CLI commands in [Table 4-7](#) to manage Key Vault Secrets.

Table 4-7 Azure CLI commands for managing Key Vault secrets

Azure CLI command	Description
Az keyvault secret backup	Backs up a specific secret in a secure manner
Az keyvault secret delete	Deletes a specific secret from the Key Vault
Az keyvault secret download	Downloads a secret from the Key Vault
Az keyvault secret list	Lists secrets in a specific Key Vault
Az keyvault secret list-deleted	Lists secrets that have been deleted but not purged from the Key Vault
Az keyvault secret list-versions	Lists all versions of secrets stored in the Key Vault
Az keyvault secret purge	Permanently removes a specific secret so that it cannot be recovered from the Key Vault
Az keyvault secret recover	Recovers a deleted secret to the latest version
Az keyvault	Restores a backed-up secret

<code>secret restore</code>	
<code>Az keyvault secret set</code>	Creates or updates a secret in Key Vault
<code>Az keyvault secret set-attributes</code>	Modifies the attributes associated with a specific Key Vault secret
<code>Az keyvault secret show</code>	Retrieves a specific secret from an Azure Key Vault
<code>Az keyvault secret show-deleted</code>	Views a specific deleted, but not purged, secret

More Info Key Vault Secrets

You can learn more about Key Vault secrets at
<https://docs.microsoft.com/en-us/azure/key-vault/secrets>.

Configure key rotation

Key rotation is the process of updating an existing key or secret with a new key or secret. You should do this on a regular basis in case the existing key or secret has accidentally or deliberately become compromised. How often you do this depends on the needs of your organization, with some organizations rotating keys every 28 days and others rotating them every six months.

Manage Keys

Cryptographic keys stored in an Azure Key Vault are stored as JSON Web Key (JWK) objects. Azure Key Vault supports RSA and Elliptic Curve (EC) keys only. Azure Key Vault supports two types of protection for keys, software protection and hardware secure module (HSM) protection. These differences manifest in the following manner:

- **Software-protected keys** The key is processed in software by Azure Key Vault. The key is protected using encryption at rest, with the system key stored in an Azure HSM. RSA or EC keys can be imported into an Azure Key Vault configured for software protection. You can also configure Azure Key Vault to create a key that uses these algorithms.
- **HSM-protected keys** The key is stored in a specially allocated HSM. Clients can import RSA or EC keys from a software-protected source or from a compatible HSM device. You can also use the Azure management plane to request that Key Vault generate a key using these algorithms. When you use HSM-protected keys, the `key_hsm` attribute is appended to the JWK.

Azure Key Vault allows the following operations to be performed on key objects:

- **Create** This operation allows a security principal to create a key. The key value will be generated by Key Vault and stored in the vault. Key Vault supports the creation of asymmetric keys.
- **Import** Allows the security principal to import an existing key into Key Vault. Key Vault supports the importation of asymmetric keys.
- **Update** Allows a security principal to modify key attributes (metadata) associated with a key that is stored within Key Vault.
- **Delete** Allows a security principal to remove a key from Key Vault.
- **List** Allows a security principal to list all keys in a Key Vault.
- **List versions** Allows a security principal to view all versions of a specific key in a Key Vault.
- **Get** Allows a security principal to view the public elements of a specific key stored in a Key Vault.
- **Backup** Exports a key from the Key Vault in a protected form.
- **Restore** Imports a previously exported Key Vault key.

You can use keys that are stored within an Azure Key Vault to perform the following cryptographic operations:

- Sign and Verify
- Key Encryption/Wrapping
- Encrypt and Decrypt

You can manage Key Vault keys using Azure portal by navigating to the Key Vault and selecting **Keys** under **Settings**, as shown in [Figure 4-41](#).

The screenshot shows the 'Keys' page of an Azure Key Vault named 'KeyVaultODLTMagnus'. The left sidebar under 'Settings' has 'Keys' selected. The main content area includes a search bar, a 'Generate/Import' button, a 'Refresh' button, and a 'Restore Backup' button. A table lists keys, but it shows 'There are no keys available.'

Figure 4-41 Keys page

To create a Key using Azure Key Vault in the Azure portal, perform the following steps:

1. In the Azure portal, open the Key Vault that you want to create the key in and navigate to **Keys** in the **Settings** section.
2. On the **Keys** page, click **Generate/Import**. This will open the **Create A Key** page.
3. On the **Create A Key** page shown in Figure 4-42, make sure that the **Options** drop-down menu is set to **Generate**. Provide a name for the key, specify the key properties, specify whether the key has an activation or expiration date, and specify whether the key is enabled. Azure Key Vault will generate the key when you click **Create**.

Home > KeyVaultRG > KeyVaultODLTMagnus | Keys >

Create a key

Options
Generate

Name * ⓘ
ExampleKey

Key Type ⓘ
RSA EC

RSA Key Size
2048 3072 4096

Set activation date? ⓘ

Set expiration date? ⓘ

Enabled?
Yes No

Create

Figure 4-42 Creating a key

You can use the Azure PowerShell cmdlets in [Table 4-8](#) to manage Azure Key Vault keys.

Table 4-8 PowerShell cmdlets for managing Azure Key Vault keys

PowerShell cmdlet	Description
Add-AzKeyVaultKey	Creates or imports a key in an Azure Key Vault
Backup-AzKeyVaultKey	Backs up a key stored in an Azure Key Vault
Get-AzKeyVaultKey	Views keys stored in an Azure Key Vault

Remove-AzKeyVaultKey	Deletes a key stored in an Azure Key Vault
Restore-AzKeyVaultKey	Recovers a key to Azure Key vault from a backup
Undo-AzKeyVaultKeyRemoval	Undeletes a deleted Azure Key Vault key
Update-AzKeyVaultKey	Allows you to update the attributes of a key stored in an Azure Key vault

You can use the Azure CLI commands in [Table 4-9](#) to manage Azure Key Vault keys.

Table 4-9 Azure CLI commands to manage Azure Key Vault keys

Command	Description
Az keyvault key backup	Backs up an Azure Key Vault key
Az keyvault key create	Creates a new Azure Key Vault key
Az keyvault key decrypt	Uses an Azure Key Vault key to decrypt data
Az keyvault key delete	Deletes an Azure Key Vault key
Az keyvault key download	Downloads the public part of a stored key
Az keyvault key encrypt	Encrypts data using a key stored in Azure Key Vault

Az keyvault key import	Imports a private key
Az keyvault key list	Lists the Azure Key Vault keys in a specific vault
Az keyvault key list-deleted	Lists Azure Key Vault keys that have been deleted but can be recovered
Az keyvault key list-versions	Lists Azure Key Vault key versions
Az keyvault key purge	Permanently deletes an Azure Key Vault key from the Key Vault
Az keyvault key recover	Recovers a deleted key
Az keyvault key restore	Restores a key from a backup
Az keyvault key set-attributes	Allows you to configure the attributes of an Azure Key Vault key
Az keyvault key show	View the public portion of an Azure Key Vault key
Az keyvault key show-deleted	View the public portion of a deleted Azure Key Vault key

More Info Key Vault Keys

You can learn more about Key Vault keys at
<https://docs.microsoft.com/en-us/azure/key-vault/keys>.

Rotate Secrets

Earlier in this chapter, you learned about the concept of key rotation that followed this process:

1. The access keys to a storage account were rotated through a process by which the applications that used the first key were switched to the second key.
2. The first key was retired and replaced.
3. Eventually, the applications were migrated back to use the first key.
4. Once the applications were migrated back to the first key, the second key was replaced, and the process could start again.

While Microsoft recommends the use of identity rather than secrets for authentication, there are workloads that run in Azure that cannot leverage identity-based authentication and must instead rely upon keys and secrets for authentication.

When you publish a secret into an Azure Key Vault, you can specify an expiration date for that secret, as shown in [Figure 4-43](#). You can use the publication of a “near expiry” event to Azure Event Grid as the trigger for a functions app that would generate a new version of the secret and that then updates the relevant workload to use the newly generated secret, allowing the existing secret to be discarded.

Create a secret

Upload options

Manual

Name * ⓘ

ExampleSecret ✓

Value * ⓘ

***** ✓

Content type (optional)

Set activation date? ⓘ

Set expiration date? ⓘ

Expiration Date

01/01/2023 6:00:00 PM

(UTC+10:00) Canberra, Melbourne, Sydney ✓

Enabled? Yes No

Create

Figure 4-43 Creating a secret

More Info Rotate Secrets

You can learn more about automating secret rotation at
<https://docs.microsoft.com/en-us/azure/key-vault/secrets/tutorial-rotation>.

Backup and restore of Key Vault items

The items stored in Key Vault are by their nature valuable and something to which you don't want to lose access. As Key Vault items are valuable, you should ensure that these items are backed up and can be recovered if something goes wrong. "Something goes wrong" can include items being accidentally deleted or corrupted, or it can mean an administrative error that causes you to lose access to the Key Vault itself. For example, you could lose access to the Key Vault if a malicious actor gains control of your subscription or if a distracted administrator incorrectly reconfigures RBAC permissions or the Key Vault's Access policy. Unlike on-premises hardware security modules that store secrets, Azure Key Vaults will fail over to a paired Azure region without requiring intervention should something disastrous happen to the datacenter that hosts the primary instance of the Key Vault.

When you back up a Key Vault Item, the item will be available for download as an encrypted blob. Recovery involves recovering this encrypted blob to the same or another Key Vault within the same subscription. It is important to note that this encrypted blob can only be decrypted inside a Key Vault within the same Azure subscription and Azure geography as the Key Vault the item was first backed up from. For example, if you backed up a secret stored in a Key Vault that was hosted in Australia in subscription A, you wouldn't be able to restore that secret to a Key Vault in an Azure geography outside Australia or in a Key Vault associated with any subscription other than subscription A.

At the time of writing, Azure Key Vault does not allow for the entirety of a Key Vault in a single back up operation. Microsoft cautions that you should perform Key Vault back-up operations manually rather than automatically. This is because automatic operations using the currently available tools are likely to result in errors. It's also possible, using automatic operations, to exceed the Key

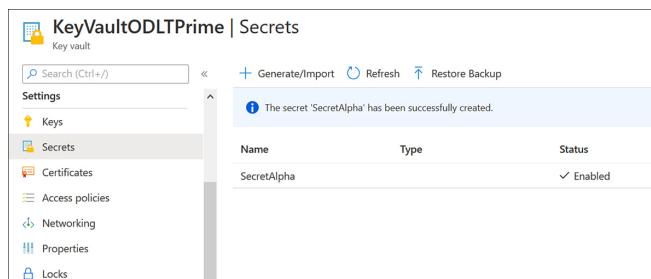
Vault's service limits in terms of requests per second. If this occurs, the Key Vault will be throttled, causing any back up operation to fail. Using scripts or automated actions to back up Key Vault items is not supported by Microsoft or the Azure Key Vault development team.

To back up objects in an Azure Key Vault, the following conditions must be met:

- Contributor-level or higher permissions on the Key Vault
- A primary Key Vault that contains items that you want to back up
- A secondary Key Vault where the secrets will be restored

To back up an item in the Azure portal, perform the following steps:

1. In the Azure portal, open the Key Vault. On the **Settings** page, select the item type that you want to back up and then select the item you want to back up. In Figure 4-44, the **Secrets** section is selected.

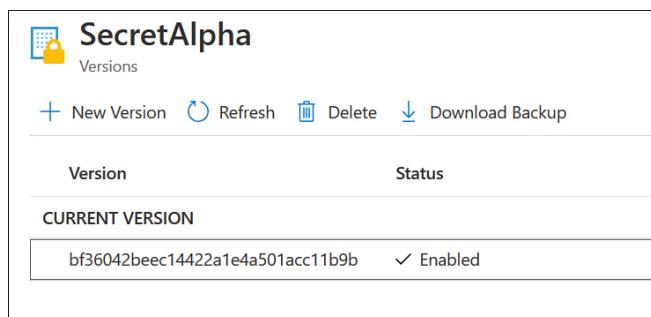


The screenshot shows the 'KeyVaultODLTPrime | Secrets' page in the Azure portal. The left sidebar has 'Settings' selected, with 'Keys', 'Certificates', 'Access policies', 'Networking', 'Properties', and 'Locks' also listed. The main area shows a message: 'The secret 'SecretAlpha' has been successfully created.' Below this is a table with columns 'Name', 'Type', and 'Status'. One row is shown: 'SecretAlpha' with 'Enabled' status.

Name	Type	Status
SecretAlpha		✓ Enabled

Figure 4-44 Secrets in Key Vault

2. Select the item that you want to back up on the item's page, shown in Figure 4-45, and select **Download Backup**.



The screenshot shows the 'SecretAlpha' item page. The top navigation bar includes 'New Version', 'Refresh', 'Delete', and 'Download Backup'. The main table has columns 'Version' and 'Status'. A single row is shown under 'CURRENT VERSION': 'bf36042beec14422a1e4a501acc11b9b' with 'Enabled' status.

Version	Status
bf36042beec14422a1e4a501acc11b9b	✓ Enabled

Figure 4-45 Download backup

3. Select **Download** to download the encrypted blob.

To restore an item using the Azure portal, perform the following steps:

1. In the Azure portal, open the Key Vault to which you want to restore the item. On the **Settings** page, select the item type that you want to restore.
2. Click **Restore Backup** (see Figure 4-46).

The screenshot shows the 'KeyVaultODLTSecondus | Secrets' page in the Azure portal. The left sidebar has 'Overview' selected. The main area shows a table with one row: 'There are no secrets available.' At the top right, there are buttons for 'Generate/Import', 'Refresh', and 'Restore Backup'. The 'Restore Backup' button is highlighted with a blue border.

Figure 4-46 Restore backup

3. On the **File Upload** page, select the encrypted blob that you want to restore to the Key Vault and then select **Open**. The encrypted blob will be uploaded to the Key Vault. An item will be restored as long as the Key Vault is in the same subscription and geographic region as the Key Vault that hosted the originally backed up item.

You can use the Azure CLI commands in Table 4-10 to back up Key Vault items.

Table 4-10 Azure CLI commands for backing up Key Vault items

Azure CLI command	Description
<code>Az keyvault certificate backup</code>	Use this command to back up specific certificates stored in an Azure Key Vault.
<code>Az keyvault key backup</code>	Use this command to back up specific keys stored in an Azure Key Vault.

Az keyvault secret backup	Use this command to back up specific secrets stored in an Azure Key Vault.
---------------------------	--

You can use the Azure CLI commands shown in [Table 4-11](#) to restore Key Vault items.

Table 4-11 Azure CLI commands for restoring Key Vault items

Azure CLI command	Description
Az keyvault certificate restore	Use this command to restore a specific certificate to an Azure Key Vault.
Az keyvault key restore	Use this command to restore a specific key to an Azure Key Vault.
Az keyvault secret restore	Use this command to restore a specific secret to an Azure Key Vault.

You can use the Azure PowerShell commands shown in [Table 4-12](#) to back up Key Vault items.

Table 4-12 Azure PowerShell commands to back up Key Vault items

Azure PowerShell command	Description
Backup-AzureKeyVaultCertificate	Use this cmdlet to back up specific certificates stored in an Azure Key Vault.
Backup-AzureKeyVaultKey	Use this cmdlet to back up an Azure Key Vault Key.

Backup-AzureKeyVaultSecret	Use this cmdlet to back up a specific secret that is stored in an Azure Key Vault.
----------------------------	--

You can use the Azure PowerShell commands in [Table 4-13](#) to restore Key Vault items.

Table 4-13 Azure PowerShell commands to restore Key Vault items

Azure PowerShell command	Description
Restore-AzureKeyVaultCertificate	Use this cmdlet to restore specific certificates stored in an Azure Key Vault.
Restore-AzureKeyVaultKey	Use this cmdlet to restore an Azure Key Vault Key.
Restore-AzureKeyVaultSecret	Use this cmdlet to restore a specific secret that is stored in an Azure Key Vault.

More Info Key Vault Item Backup and Recovery

You can learn more about backup and recovery of Key Vault at <https://docs.microsoft.com/en-us/azure/key-vault/general/backup>.



Exam Tip

Remember that you can only restore Key Vault items if the Key Vault you are using in the restore operation is in the same subscription and geographic region as the Key Vault where the original backup was taken.



Thought Experiment

In this thought experiment, demonstrate your skills and knowledge of the topics covered in this chapter. You can find answers to this thought experiment in the next section.

Securing data at Tailwind Traders

Tailwind Traders has migrated some of their operations to Azure and are now attempting to improve the security of the data stored in their Azure subscription. With this information in mind, Tailwind Traders has the following challenges they need to address:

- Members of the product research team need to be able to add and remove data in Blob Storage across several storage accounts. They should not be assigned any unnecessary permissions.
- To comply with local government regulation, Tailwind Traders needs to manage the keys used for transparent data encryption on their Azure SQL instance. They will be configuring BYOK.
- Members of the sales team at Tailwind Traders need to be able to regularly perform cryptographic operations with keys and certificates stored in an Azure Key Vault but should not be assigned any unnecessary permissions.

With this information, answer the following questions:

- 1.** Which RBAC role should you assign to the product research team?
- 2.** Where should tailwind traders store its TDE key?
- 3.** Which RBAC role should the Sales team be assigned to the Key Vault?

THOUGHT EXPERIMENT ANSWERS

This section contains the solution to the thought experiment. Each answer explains why the answer choice is correct.

- 1.** The product research team should be assigned Storage Blob Data Contributor role because this provides the minimum necessary permissions to add and remove data from Blob Storage.
- 2.** Tailwind traders should store the TDE key in an Azure Key Vault because this is the only location in which you can store a key in a BYOK scenario.
- 3.** The sales team should be assigned the Key Vault Crypto User RBAC role because this allows them to perform cryptographic operations on keys and certificates.

CHAPTER SUMMARY

- There are two storage account access keys that can be used to provide access to a storage account. You should only use one at a time so that you can perform key rotation on a regular basis:
 - Shared Access Signatures (SAS) allow you to provide secure granular delegated access to storage accounts.
 - Stored access policies allow you to specifically control service-level shared access signatures.
- Rather than rely upon storage account keys or shared access signatures, you can use Azure AD to authorize access to Blob and Queue Storage. Azure AD authenticates a security principal's identity and then returns an OAuth 2.0 token.
- When you enable AD DS authentication for Azure Files, your Active Directory Domain Services (AD DS) domain joined computers can mount Azure File Shares using AD DS user credentials.
- You configure share-level permission by assigning RBAC roles at the Azure File Share-level. Once you have assigned share-level permissions to an Azure File Share using RBAC, you should then configure file and folder permissions on the contents of the share.
- Azure Storage encryption is enabled by default for all storage accounts regardless of performance tier or access tier. This means you don't have to modify code or applications for Azure Storage Encryption to be enabled.
- Encryption scopes allow you to configure separate encryption keys at the container and blob level.

- Advanced threat protection for Azure Storage allows you to detect unusual and malicious attempts to interact with Azure Storage accounts.
- When you create an Azure SQL database server instance, you create an administrator login and a password associated with that login. This administrative account granted full administrative permissions on all databases hosted off the Azure SQL instance as a server-level principal.
- Auditing allows you to track database events, such as tables being added or dropped. Audit logs for Azure SQL databases can be stored in an Azure Storage account, a Log Analytics workspace, or Event Hubs.
- Azure SQL Database Advanced Threat Protection allows you to detect unusual activity that might indicate that a third party might be trying to attack your organization's Azure SQL databases.
- Transparent data encryption (TDE) allows you to protect Azure SQL databases by encrypting data at rest. When you enable TDS, the databases, associated backups, and transaction log files are automatically encrypted and decrypted, as necessary.
- Always Encrypted is a technology available for Azure SQL that allows you to protect specific types of sensitive data that has a known recognizable pattern, such as passport numbers, tax file identification numbers, and credit card numbers.
- Azure Key Vault allows you to store information that should not be made public, such as secrets, certificates, and keys.
- You use Key Vault access control policies to manage permissions to secrets, certificates, and keys at the data plane level. Each Key Vault access control policy includes entries specifying what access the designated security principal has to keys, secrets, and certificates.

Index

A

- access control, 38-63, 74-85
 - access reviews, 40-42
 - activating/configuring PIM, 43-45
 - administering MFA users, 54-60
 - account lockout settings, 57
 - blocking/unblocking users, 58
 - fraud alert settings, 58
 - OATH tokens, 59
 - phone call settings, 59
 - reporting utilization, 60
 - application access, 64-73
 - API management policies, 73
 - assigning, 66-70
 - permission consent, 71-73
 - permission scopes, 70-71
 - registering applications, 64-66
 - for Azure Key Vault, 282-285
 - best practices, 81
 - conditional access policies, 46-54
 - creating, 47-49
 - implementing MFA, 49-54
 - types of, 46-47
 - configuring identity protection, 60-63
 - custom roles, 81-84

identifying roles, [81](#)
interpreting permissions, [84](#)
monitoring privileged access, [38-40](#)
principle of least privilege, [81](#)
RBAC roles
 assigning, [245-247](#)
 levels of, [244](#)
 list of, [245](#)
resource group permissions, [79-80](#)
subscription and resource permissions, [74-79](#)
viewing user resource permissions, [84-85](#)
for VMs (virtual machines), [155](#)

accessing

- Azure Activity Log, [182](#)
- Azure AD administrative console, [6](#)

access keys for storage accounts, [247](#)

- rotating keys, [247-250](#)
- viewing keys, [248-249](#)

access reviews, [40-42](#)

account lockout settings for MFA, [57](#)

account SAS, [251-254](#)

ACR (Azure Container Registry)

- security configuration, [167-168](#)
- vulnerability management, [164-165](#)

action groups for Azure Monitor alerts, [185-186](#)

Active Directory Federation Services (AD FS) in Azure

- AD Connect, [28](#)

activity logs in Azure Monitor, [180](#)

- accessing, [182](#)

Add-AzKeyVaultCertificate cmdlet, [293](#)

Add-AzKeyVaultCertificateContact cmdlet, [293](#)

Add-AzKeyVaultKey cmdlet, [300](#)
Add-AzRouteConfig cmdlet, [97](#)
Add-AzureADDirectoryRoleMember cmdlet, [79](#)
Add-AzureADGroupMember cmdlet, [8](#)
Add-AzureADGroupOwner cmdlet, [8](#)
Add-AzVirtualNetworkPeering cmdlet, [99](#)
adding
 certificates to Azure Key Vault, [289-293](#)
 compliance standards to Regulatory Compliance dashboard, [210-211](#)
 group members, [10](#)
ADE (Azure Disk Encryption), [168-169](#)
ad hoc SAS, [251](#)
administrative console (Azure AD), accessing, [6](#)
ADS (Advanced Data Security), [199](#)
Advanced Threat Protection (ATP) for Azure Storage, [267-268](#)
AKS (Azure Kubernetes Service)
 authentication, [159-161](#)
 isolation configuration, [166-167](#)
 security configuration, [161-164](#)
alerts
 in Azure Monitor
 creating/customizing, [183-189](#)
 viewing/changing, [188](#)
 in Azure Sentinel, creating/customizing, [217-224](#)
Always Encrypted, [279-281](#)
analytics in Azure Sentinel, [213](#)
API management policies, [73](#)
application access, [64-73](#)
 API management policies, [73](#)

assigning, [66-70](#)
permission consent, [71-73](#)
permission scopes, [70-71](#)
registering applications, [64-66](#)

Application Administrator role, [75](#)

Application Developer role, [75](#)

application gateways

- Azure Front Door, [126-133](#)
 - capabilities, [126](#)
 - configuring, [127-133](#)
 - topology, [127](#)
- WAF (Web Application Firewall) configuration, [133-135](#)

application objects, [2](#)

application permissions, [71](#)

application rules, creating, [120-122](#)

applications

- assigning roles, [3-6](#)
- registering, [2, 64-66](#)

application security groups (ASGs), [114-117](#)

app passwords, [32](#)

ArcDelete ACR role, [167](#)

ArcImageSigner ACR role, [167](#)

ArcPull ACR role, [167](#)

ArcPush ACR role, [167](#)

ASGs (application security groups), [114-117](#)

assigning

- application access, [66-70](#)
- permissions to service principals, [3-6](#)

RBAC roles, [245-247](#)

roles to applications, [3-6](#)

users to roles, [78-79](#)

ATP (Advanced Threat Protection) for Azure Storage,
[267-268](#)

auditing databases, [270-273](#)

audit logs, viewing, [271-273](#)

authentication, [30-36](#)

- in Azure App Service, configuring, [174-176](#)
- for Azure Files, [256-261](#)
 - enabling, [257-261](#)
 - file and folder permissions, [260](#)
 - share-level permissions, [259](#)
- certificate-based, [33](#)
- for containers, [159-161](#)
- for databases, [268-269](#)
- MFA (multifactor authentication), [49, 54](#)
 - administering users, [54-60](#)
 - enabling, [50-54](#)
- passwordless, [33-36](#)
- for storage accounts, [255-256](#)
- types of, [31-32](#)
- for VPN gateways, [104-106](#)

Authentication Administrator role, [75](#)

authorization in Azure App Service, configuring, [174-176](#)

Azure Active Directory (Azure AD)

- access control, [38-63, 74-85](#)
 - access reviews, [40-42](#)
 - activating/configuring PIM, [43-45](#)
 - administering MFA users, [54-60](#)
 - best practices, [81](#)
 - conditional access policies, [46-54](#)
 - configuring identity protection, [60-63](#)

custom roles, [81-84](#)
identifying roles, [81](#)
interpreting permissions, [84](#)
monitoring privileged access, [38-40](#)
principle of least privilege, [81](#)
resource group permissions, [79-80](#)
subscription and resource permissions, [74-79](#)
viewing user resource permissions, [84-85](#)

administrative console, accessing, [6](#)
application access, [64-73](#)
 API management policies, [73](#)
 assigning, [66-70](#)
 permission consent, [71-73](#)
 permission scopes, [70-71](#)
 registering applications, [64-66](#)
applications, registering, [2](#)
authentication methods, [30-36](#)
 certificate-based, [33](#)
 passwordless, [33-36](#)
 for storage accounts, [255-256](#)
 types of, [31-32](#)
container authentication, [159-161](#)
identities
 configuring identity protection, [60-63](#)
 groups, [6-12](#)
 service principals, [2-6](#)
 types of, [1](#)
 users, [13-15](#)
password writeback, [15-30](#)
 enabling self-service password reset, [28-30](#)
installing/configuring Azure AD Connect, [15-28](#)

transferring subscriptions, [36-37](#)

Azure Active Directory Connect, [15-28](#)

- connectivity requirements, [16](#)
- deployment account requirements, [17](#)
- installing, [17-25](#)
- sign-in options, [27-28](#)
- SQL Server requirements, [16-17](#)
- system requirements, [15-16](#)
- UPN suffixes and nonroutable domains, [25-27](#)

Azure Active Directory Domain Services (Azure AD DS),
authentication for Azure Files, [256-261](#)

- enabling, [257-261](#)
- file and folder permissions, [260](#)
- share-level permissions, [259](#)

Azure Active Directory logs in Azure Monitor, [181](#)

Azure Activity Log, accessing, [182](#)

Azure App Service

- firewalls, [143-144](#)
- security configuration, [170-176](#)
 - authentication, [174-176](#)
 - software updates, [176](#)
 - SSL/TLS certificates, [172-174](#)

Azure Application Gateway

- as load balancer, [126](#)
- WAF (Web Application Firewall) configuration, [133-135](#)

Azure Automation Update Management, [156-159](#)

Azure Bastion, [135-137](#)

Azure Blueprint security settings, configuring, [236-240](#)

Azure Container Registry (ACR)

- security configuration, [167-168](#)

vulnerability management, [164-165](#)

Azure DDoS, [147-151](#)

Azure Disk Encryption (ADE), [168-169](#)

Azure Files authentication, [256-261](#)

- enabling, [257-261](#)
- file and folder permissions, [260](#)
- share-level permissions, [259](#)

Azure Firewall

- application rules, [120-122](#)
- configuring, [119-120](#)
- logging, [123-125](#)
- network rules, [122-123](#)
- topology, [117-118](#)

Azure Front Door, [126-133](#)

- capabilities, [126](#)
- configuring, [127-133](#)
- topology, [127](#)

WAF (Web Application Firewall) integration, [133](#)

Azure Key Vault

- access control, [282-285](#)
- with ADE (Azure Disk Encryption), [168](#)
- backup and restore, [303-307](#)
- certificate management, [288-296](#)
- firewalls, [142-143](#)
- key rotation, [298-303](#)
- network access, [282-285](#)
- permissions management, [285-287](#)
- RBAC usage, [287-288](#)
- secrets management, [296-298](#)
- secrets rotation, [302-303](#)
- storage account encryption keys, [264](#)

Azure Kubernetes Service (AKS)

authentication, [159-161](#)

isolation configuration, [166-167](#)

security configuration, [161-164](#)

Azure Logic Apps playbooks, configuring, [224-228](#)

Azure Monitor, [179-196](#)

activity logs, [180](#)

alerts

creating/customizing, [183-189](#)

viewing/changing, [188](#)

Azure Active Directory logs, [181](#)

enabling, [179](#)

layers in, [180-181](#)

log collecting

IaaS VM logs, [192-194](#)

searching events in Log Analytics workspace, [195-196](#)

Security and Audit solution, [194-195](#)

metrics in, [181-184](#)

operational overview, [180-183](#)

resource (diagnostic) logs, [180](#)

configuring settings, [189-192](#)

resources in, [181](#)

Azure Policy

centralized policy management in Azure Security Center, [206-209](#)

security settings, configuring, [232-236](#)

Azure Resources layer (Azure Monitor), [180](#)

Azure Security Center, [196-211](#)

for AKS (Azure Kubernetes Service), [163-164](#)

Azure App Service security recommendations in, [171-172](#)

centralized policy management, [206-209](#)

JIT (Just In Time) VM access, [201-205](#)

Regulatory Compliance dashboard, [209-211](#)

viewing endpoint protection, [151-154](#)

VM threat detection, [155-156](#)

vulnerability assessment, [196-200](#)

vulnerability management, [164-165](#)

Azure Sentinel, [212-232](#)

alerts, creating/customizing, [217-224](#)

components of, [212-213](#)

data connectors, configuring, [213-217](#)

playbooks, configuring, [224-228](#)

results, evaluating, [228-232](#)

Azure SQL Database Advanced Threat Protection, [273-276](#)

Azure SQL databases. *See* [databases](#)

Azure Storage. *See* [storage accounts](#)

Azure Subscription layer (Azure Monitor), [180](#)

Azure Tenant layer (Azure Monitor), [181](#)

B

backing up Azure Key Vault items, [303-307](#)

Backup-AzKeyVaultCertificate cmdlet, [293](#)

Backup-AzKeyVaultKey cmdlet, [300](#)

Backup-AzKeyVaultSecret cmdlet, [297](#)

Backup-AzureKeyVaultCertificate cmdlet, [306](#)

Backup-AzureKeyVaultKey cmdlet, [306](#)

Backup-AzureKeyVaultSecret cmdlet, [306](#)

best practices

access control, [81](#)
for SAS (Shared Access Signatures), [251-252](#)

Billing Administrator role, [75](#)

blobs

- authentication, [255-256](#)
- encryption, viewing status, [262-263](#)
- stored access policies, [255](#)

BlobStorage accounts, [244](#)

BlockBlobStorage accounts, [244](#)

blocking MFA users, [58](#)

blueprints, [236-240](#)

BYOK (Bring Your Own Key), [276](#)

C

cases in Azure Sentinel, [212](#)

CDS (Common Data Service), [176](#)

centralized policy management in Azure Security Center, [206-209](#)

certificate authorities for Azure Key Vault, [289-292](#)

certificate policies, elements of, [288-289](#)

certificate-based authentication, [33](#)

certificates

- in Azure Key Vault
 - adding, [289-293](#)
 - backup and restore, [303-307](#)
 - importing, [289-293](#)
 - managing, [288-296](#)
 - permissions, [286](#)
- contacts information, [289](#)
- SSL/TLS, configuring, [172-174](#)

changing Azure Monitor alerts, [188](#)
Cloud Application Administrator role, [75](#)
Cloud Device Administrator role, [75](#)
Common Data Service (CDS), [176](#)
Community page in Azure Sentinel, [213](#)
Compliance Administrator role, [75](#)
compliance policies in Azure Security Center, [209-211](#)
compute security
 for ACR (Azure Container Registry), [167-168](#)
 authentication for containers, [159-161](#)
 for Azure App Service, [170-176](#)
 container security, [161-164](#)
 disk encryption, [168-169](#)
 endpoint security, [151-156](#)
 isolation, [166-167](#)
 system updates for VMs, [156-159](#)
 vulnerability management, [164-165](#)
Conditional Access Administrator role, [75](#)
conditional access policies, [46-54](#)
 creating, [47-49](#)
 implementing MFA, [49-54](#)
 types of, [46-47](#)
Connect-AzAccount cmdlet, [95](#)
connectivity requirements for Azure AD Connect, [16](#)
connectors. *See* data connectors
containers
 authentication, [159-161](#)
 isolation configuration, [166-167](#)
 security configuration, [161-164](#)
Contributor ACR role, [167](#)
Contributor role, [77](#)

Customer Lockbox access approver role, 75
custom roles, 81-84
custom routes, creating, 97

D

dashboards in Azure Sentinel, 212
databases
 auditing, 270-273
 authentication, 268-269
 Azure SQL Database Advanced Threat Protection,
 273-276
 encryption
 Always Encrypted, 279-281
 TDE (transparent data encryption), 276-279
 firewalls for, 140-142
data connectors in Azure Sentinel, 213-217
data plane for Key Vault access control, 282
data plane logs, 192
DDoS (distributed denial of service) protection, 147-151
Debug-AzStorageAccountAuth cmdlet, 259
delegated permissions, 71
deleting
 group members, 10
 nested groups, 12
 users, 14
deployment account requirements for Azure AD
 Connect, 17
Destination Network Address Translation (DNAT), 118
detection mode (WAF on Application Gateway), 134
deterministic encryption, 279
Device Administrators role, 75

diagnostic logs in Azure Monitor, [180](#)
configuring settings, [189-192](#)

Directory Readers role, [75](#)

Directory Synchronization Accounts role, [75](#)

Directory Writers role, [75](#)

distributed denial of service (DDoS) protection, [147-151](#)

DNAT (Destination Network Address Translation), [118](#)

dynamic group membership, [7](#)

Dynamics 365 Administrator/CRM Administrator role, [75](#)

E

email addresses for authentication, [32](#)

email scope (application access), [71](#)

enabling

- AD DS authentication, [257-259](#)
- Azure AD DS authentication, [260-261](#)
- Azure Monitor, [179](#)
- database auditing, [270-273](#)
- database authentication, [268-269](#)
- firewall logging, [124-125](#)
- MFA (multifactor authentication), [50-54](#)
- passwordless authentication, [34-35](#)
- self-service password reset, [28-30](#)
- sign-in risk policies, [61-63](#)
- user-risk policies, [61-63](#)

encryption

of databases

- Always Encrypted, [279-281](#)
- TDE (transparent data encryption), [276-279](#)

ExpressRoute, [106-107](#)
of storage accounts, [262-267](#)
 infrastructure encryption, [264](#)
 key management, [263-264](#)
 scopes, [264-267](#)
 viewing status, [262-263](#)
types of, [279-280](#)
for VMs (virtual machines), [156](#)

encryption at rest, [168-169](#)

endpoint security within VMs, [151-156](#)

evaluating results in Azure Sentinel, [228-232](#)

events, searching in Log Analytics workspace (Azure Monitor), [195-196](#)

Exchange Administrator role, [76](#)

ExpressRoute, [92](#), [104-107](#)

external connectors in Azure Sentinel, [214](#)

F

FIDO2 Security keys, [34](#)

file and folder permissions, [260](#)

FileStorage accounts, [244](#)

firewalls

- Azure Firewall
 - application rules, [120-122](#)
 - configuring, [119-120](#)
 - logging, [123-125](#)
 - network rules, [122-123](#)
 - topology, [117-118](#)
- for Azure Key Vault, [283-285](#)
- resource firewalls, [138-144](#)

in Azure App Service, [143-144](#)
in Azure Key Vault, [142-143](#)
in Azure SQL databases, [140-142](#)
in Azure Storage, [138-140](#)

WAF (Web Application Firewall)
Azure Front Door integration, [133](#)
configuring on Azure Application Gateway, [133-135](#)
inbound HTTP/S protection, [118, 122](#)

fraud alert settings for MFA, [58](#)

Front Door. *See* [Azure Front Door](#)

G

General-Purpose V2 accounts, [244](#)
Get-ADOrganizationalUnit cmdlet, [258](#)
Get-AdUser cmdlet, [257](#)
Get-AzAdServicePrincipal cmdlet, [3](#)
Get-AzKeyVaultCertificate cmdlet, [293](#)
Get-AzKeyVaultCertificateContact cmdlet, [293](#)
Get-AzKeyVaultCertificateIssuer cmdlet, [293](#)
Get-AzKeyVaultCertificateOperation cmdlet, [293](#)
Get-AzKeyVaultCertificatePolicy cmdlet, [293](#)
Get-AzKeyVaultKey cmdlet, [300](#)
Get-AzKeyVaultSecret cmdlet, [297](#)
Get-AzRouteTable cmdlet, [97](#)
Get-AzureADDirectoryRole cmdlet, [78](#)
Get-AzureADDirectoryRoleMember cmdlet, [78](#)
Get-AzureADGroup cmdlet, [8](#)
Get-AzureKeyVaultSecret cmdlet, [296](#)
Get-AzVirtualNetworkGatewayConnectionSharedKey
cmdlet, [105](#)

Get-AzVmDiskEncryptionStatus cmdlet, [169](#)
Global Administrator/Company Administrator role, [76](#)
groups, [6-12](#)
 adding/removing members, [10](#)
 assigning application access, [67-70](#)
 assigning roles to, [244](#)
 creating, [8-10](#)
 dynamic membership, [7](#)
 naming, [9](#)
 nested, [10-12](#)
 types of, [6-7](#)
Guest Inviter role, [76](#)

H-I

HSM (hardware secure module) key protection, [299](#)
hunting in Azure Sentinel, [212](#), [231-232](#)
IaaS (Infrastructure as a Service) VM security logs,
 collecting with Azure Monitor, [192-194](#)
identities
 configuring identity protection, [60-63](#)
 groups, [6-12](#)
 adding/removing members, [10](#)
 creating, [8-10](#)
 dynamic membership, [7](#)
 naming, [9](#)
 nested, [10-12](#)
 types of, [6-7](#)
service principals, [2-6](#)
 assigning permissions, [3-6](#)
 components of, [3](#)

creating, 3
viewing list of, 3
types of, 1
users, [13-15](#)
 creating, [13-14](#)
 deleting, [14](#)
 recovering, [14](#)
identity providers for Azure App Service, [176](#)
Import-AzKeyVaultCertificate cmdlet, [293](#)
importing certificates to Azure Key Vault, [289-293](#)
inbound rules for NSGs (network security groups), [110](#)
incidents in Azure Sentinel, [230-231](#)
Information Protection Administrator role, [76](#)
Infrastructure as a Service (IaaS) VM security logs,
 collecting with Azure Monitor, [192-194](#)
infrastructure encryption, [264](#)
installing Azure AD Connect, [17-25](#)
Intune Administrator role, [76](#)
IPSec encryption, [107](#)
isolation configuration, [166-167](#)

J-K

JIT (Just In Time) VM access, [201-205](#)
key management for storage accounts, [247](#). *See also*
[Azure Key Vault](#)
 encryption, [263-264](#)
 rotating keys, [247-250](#)
 viewing keys, [248-249](#)
Key Vault. *See* [Azure Key Vault](#)

Key Vault Administrator role, [288](#)
Key Vault Certificates Officer role, [288](#)

Key Vault Contributor role, [288](#)
Key Vault Crypto Officer role, [288](#)
Key Vault Crypto Service Encryption role, [288](#)
Key Vault Crypto User role, [288](#)
Key Vault Reader role, [288](#)
Key Vault Secrets Officer role, [288](#)
Key Vault Secrets User role, [288](#)
keys in Azure Key Vault
 backup and restore, [303-307](#)
 permissions, [286](#)
 rotating, [298-303](#)
KQL (Kusto Query Language), [125](#)
Kubernetes. *See AKS (Azure Kubernetes Service)*

L

layers in Azure Monitor, [180-181](#)
least privilege, principle of, [81, 155, 166](#)
License Administrator role, [76](#)
license requirements, PIM (Privileged Identity Management), [45](#)
load balancers, Azure Application Gateway as, [126](#)
locks in Azure Blueprint, [240](#)
Log Analytics workspace (Azure Monitor), searching events, [195-196](#)
Log Analytics workspace (Azure Sentinel), [228-229](#)
log collecting with Azure Monitor
 IaaS VM logs, [192-194](#)
 searching events in Log Analytics workspace, [195-196](#)
 Security and Audit solution, [194-195](#)
log retention in Azure Monitor, configuring, [189-192](#)
logging in Azure Firewall, [123-125](#)

logical isolation, [166](#)

Logic Apps. *See* [Azure Logic Apps](#)

M

MACsec, [106-107](#)

management plane for Key Vault access control, [282](#)

Message Center Reader role, [76](#)

metrics in Azure Monitor, [181-183](#)

 creating alerts from, [184](#)

MFA (multifactor authentication), [49-60](#)

 administering users, [54-60](#)

 account lockout settings, [57](#)

 blocking/unblocking users, [58](#)

 fraud alert settings, [58](#)

 OATH tokens, [59](#)

 phone call settings, [59](#)

 reporting utilization, [60](#)

 enabling, [50-54](#)

 for VPN gateways, [105](#)

Microsoft Authenticator app, [32-34](#)

Microsoft incident creation rules in Azure Sentinel, [217](#),
[223-224](#)

Microsoft Threat Intelligence, [119](#)

mobile phone numbers for authentication, [32](#)

Monitor. *See* [Azure Monitor](#)

monitoring privileged access, [38-40](#)

multifactor authentication. *See* [MFA \(multifactor authentication\)](#)

multi-site VPNs, [104](#)

N

naming groups, 9

NAT (network address translation), 100-103

NAT Gateway

- billing, 101
- creating, 101-103
- topology, 100-101

nested groups, 10-12

network access for Azure Key Vault, 282-285

network components, 89-103

- NAT (network address translation), 100-103
- peering, 97-100
- routing, 95-97
- subnets, 91
- virtual network gateways, 91

VNets (virtual networks), configuring, 90-95

network rules, creating, 122-123

network security

- ASGs (application security groups), 114-117
- Azure Bastion, 135-137
- Azure Firewall, 117-125
- DDoS (distributed denial of service) protection, 147-151
- NSGs (network security groups), 91, 109-114, 201
- resource firewalls, 138-144
- service endpoints, 145-147
- VPN gateways, 104-108
 - authentication, 104-106
 - ExpressRoute encryption, 106-107
 - point-to-site (P2S), 107-108
 - site-to-site (S2S), 108

types of, [104](#)
WAF (Web Application Firewall), [133-135](#)
network security groups (NSGs), [91](#), [109-114](#), [201](#)
New-AzADServicePrincipal cmdlet, [3](#)
New-AzFirewallApplicationRule cmdlet, [122](#)
New-AzFirewall cmdlet, [120](#)
New-AzFirewallNetworkRule cmdlet, [123](#)
New-AzKeyVaultCertificateOrganizationDetail cmdlet,
[294](#)
New-AzKeyVaultCertificatePolicy cmdlet, [294](#)
New-AzNatGateway cmdlet, [104](#)
New-AzNetworkSecurityGroup cmdlet, [112](#)
New-AzNetworkSecurityRuleConfig cmdlet, [114](#)
New-AzRoleAssignment cmdlet, [5](#)
New-AzRouteTable cmdlet, [97](#)
New-AzureADGroup cmdlet, [8](#)
New-AzVaultCertificateAdministratorDetail cmdlet, [294](#)
New-AzVirtualNetwork cmdlet, [95](#)
New-AzVM cmdlet, [95](#)
nonroutable domains, UPN suffixes and, [25-27](#)
notebooks in Azure Sentinel, [213](#)
NSGs (network security groups), [91](#), [109-114](#), [201](#)

O

OATH tokens, [32](#)
 for MFA users, [59](#)
OAuth, [32](#)
Office 365 groups, [6-7](#)
offline access scope (application access), [71](#)
open scope (application access), [71](#)

operating systems supported on VMs, [197](#)
outbound rules for NSGs (network security groups), [111](#)
Owner ACR role, [167](#)
Owner role, [77](#)

P

P2S (point-to-site) VPNs, [104](#), [107-108](#)
pass-through authentication in Azure AD Connect, [27-28](#)
Password Administrator/Helpdesk Administrator role,
[76](#)
password authentication, [31](#)
passwordless authentication, [33-36](#)
password synchronization in Azure AD Connect, [27](#)
password writeback, [15-30](#)
 Azure AD Connect, [15-28](#)
 connectivity requirements, [16](#)
 deployment account requirements, [17](#)
 installing, [17-25](#)
 sign-in options, [27-28](#)
 SQL Server requirements, [16-17](#)
 system requirements, [15-16](#)
 UPN suffixes and nonroutable domains, [25-27](#)
 enabling self-service password reset, [28-30](#)
peering virtual networks, [97-100](#)
permission consent for application access, [71-73](#)
permission scopes for application access, [70-71](#)
permissions, [74-85](#)
 assigning to service principals, [3-6](#)
 for Azure Key Vault, [285-287](#)
 custom roles, [81-84](#)

file and folder, [260](#)
identifying roles, [81](#)
interpreting, [84](#)
principle of least privilege, [81](#)
resource group permissions, [79-80](#)
share-level, [259](#)
subscription and resource permissions, [74-79](#)
viewing user resource permissions, [84-85](#)

phone call settings for MFA, [59](#)

physical isolation, [167](#)

PIM (Privileged Identity Management)

- access reviews, [40-42](#)
- activating/configuring, [43-45](#)
- license requirements, [45](#)
- viewing resource audit history, [38-40](#)

playbooks in Azure Sentinel, [213](#)

- configuring, [224-228](#)

point-to-site (P2S) VPNs, [104, 107-108](#)

policies

- blueprints versus, [236](#)
- centralized policy management in Azure Security Center, [206-209](#)
- policy definitions, [206](#)
- policy effect, [206](#)
- policy enforcement, configuring
 - in Azure Blueprint, [236-240](#)
 - in Azure Policy, [232-236](#)

Power BI Administrator role, [76](#)

prevention mode (WAF on Application Gateway), [135](#)

pricing tiers, ACR (Azure Container Registry), [167](#)

principle of least privilege, [81, 155, 166](#)

private endpoint connections for Azure Key Vault, [284](#)

privileged access, monitoring, [38-40](#)

Privileged Identity Management (PIM)

access reviews, [40-42](#)

activating/configuring, [43-45](#)

license requirements, [45](#)

viewing resource audit history, [38-40](#)

Privileged Role Administrator role, [76](#)

profile scope (application access), [71](#)

protocols for P2S (point-to-site) VPNs, [108](#)

Q-R

Qualys extension, [196-198](#)

queue storage authentication, [255-256](#)

RADIUS, [105-106](#)

randomized encryption, [279](#)

RBAC (role-based access control)

with Azure Key Vault, [287-288](#)

configuring, [77](#)

container authentication, [159-161](#)

custom roles, [81-84](#)

identifying roles, [81](#)

interpreting permissions, [84](#)

principle of least privilege, [81](#)

resource group permissions, [79-80](#)

roles

assigning, [245-247](#)

for blob and queue storage, [256](#)

levels of, [244](#)

list of, [245](#)

subscription and resource permissions, [74-79](#)
viewing user resource permissions, [84-85](#)

Reader ACR role, [167](#)

Reader role, [77](#)
recovering users, [14](#)
registering applications, [2, 64-66](#)

Regulatory Compliance dashboard (Azure Security Center), [209-211](#)

Remove-AzKeyVaultCertificate cmdlet, [294](#)
Remove-AzKeyVaultCertificateContact cmdlet, [294](#)
Remove-AzKeyVaultCertificateIssuer cmdlet, [294](#)
Remove-AzKeyVaultCertificateOperation cmdlet, [294](#)
Remove-AzKeyVaultKey cmdlet, [300](#)
Remove-AzKeyVaultSecret cmdlet, [297](#)
Remove-AzureADDirectoryRoleMember cmdlet, [79](#)
Remove-AzureADGroup cmdlet, [8](#)
Remove-AzureADGroupMember cmdlet, [8](#)
Remove-AzureADGroupOwner cmdlet, [8](#)
Remove-AzureKeyVaultSecret cmdlet, [296](#)

removing
group members, [10](#)
nested groups, [12](#)
users, [14](#)

reports, MFA utilization, [60](#)

Reports Reader role, [76](#)

requirements
Azure AD Connect
connectivity requirements, [16](#)
deployment account requirements, [17](#)
SQL Server requirements, [16-17](#)
system requirements, [15-16](#)

certificate-based authentication, 33

PIM (Privileged Identity Management), license requirements, 45

resource audit history, viewing, 38-40

resource firewalls, 138-144

- in Azure App Service, 143-144
- in Azure Key Vault, 142-143
- in Azure SQL databases, 140-142
- in Azure Storage, 138-140

resource group permissions, 79-80

resource logs in Azure Monitor, 180

- configuring settings, 189-192

resource permissions, 74-79

- viewing, 84-85

resources in Azure Monitor, 181

Restore-AzKeyVaultCertificate cmdlet, 294

Restore-AzKeyVaultKey cmdlet, 300

Restore-AzKeyVaultSecret cmdlet, 297

Restore-AzureKeyVaultCertificate cmdlet, 306

Restore-AzureKeyVaultKey cmdlet, 306

Restore-AzureKeyVaultSecret cmdlet, 306

restoring Azure Key Vault items, 303-307

results, evaluating in Azure Sentinel, 228-232

revoking user delegation SAS, 252-253

role-based access control. *See RBAC (role-based access control)*

roles

- assigning
 - to applications, 3-6
 - users to, 78-79
- custom, 81-84

defined, 74
identifying, 81
list of, 75-76
RBAC
 assigning, 245-247
 for blob and queue storage, 256
 levels of, 244
 list of, 245
 viewing assignments, 77-78
rotating
 keys in Azure Key Vault, 298-303
 secrets in Azure Key Vault, 302-303
 storage account access keys, 247-250
routing, 95-97
rule of least privilege, 244
rules, creating
 application rules, 120-122
 network rules, 122-123

S

S2S (site-to-site) VPNs, 104, 108
SAS (Shared Access Signatures), 251-254
 account SAS, 253-254
 best practices, 251-252
 tokens, 253-254
 types of, 251
 user delegation SAS, 252-253
scheduled query rules in Azure Sentinel, 217-223
scope
 for permissions, 74

for storage account encryption, [264-267](#)

searching events in Log Analytics workspace (Azure Monitor), [195-196](#)

secrets in Azure Key Vault

- backup and restore, [303-307](#)
- managing, [296-298](#)
- permissions, [286](#)
- rotating, [302-303](#)

security

- Azure Front Door, [126-133](#)
- compute security
 - for ACR (Azure Container Registry), [167-168](#)
 - authentication for containers, [159-161](#)
 - for Azure App Service, [170-176](#)
 - container security, [161-164](#)
 - disk encryption, [168-169](#)
 - endpoint security, [151-156](#)
 - isolation, [166-167](#)
 - system updates for VMs, [156-159](#)
 - vulnerability management, [164-165](#)
- network security
 - ASGs (application security groups), [114-117](#)
 - Azure Bastion, [135-137](#)
 - Azure Firewall, [117-125](#)
 - Azure Front Door, [126-133](#)
 - DDoS (distributed denial of service) protection, [147-151](#)
 - NSGs (network security groups), [91, 109-114, 201](#)
 - resource firewalls, [138-144](#)
 - service endpoints, [145-147](#)
 - VPN gateways, [104-108](#)

WAF (Web Application Firewall), [133-135](#)

Security Administrator role, [76](#)

Security and Audit solution (Azure Monitor), [194-195](#)

Security Center. *See* [Azure Security Center](#)

security groups, [6-7](#)

Security Information and Event Management (SIEM),
[212](#)

security key sign-in, [34](#)

Security Orchestration, Automation, and Response
(SOAR), [212](#)

security principals, [74, 285](#)

security questions, [31-32](#)

Security Reader role, [76](#)

security services configuration. *See* [Azure Monitor](#)

security settings, configuring

- with Azure Blueprint, [236-240](#)
- with Azure Policy, [232-236](#)

self-service password reset (SSPR), [15](#)

- enabling, [28-30](#)

service endpoints, [145-147](#)

service principal objects, [2](#)

service principals, [2-6](#)

- assigning permissions, [3-6](#)
- components of, [3](#)
- creating, [3](#)
- viewing list of, [3](#)

service SAS, [251](#)

Service Support Administrator role, [76](#)

Set-ACL cmdlet, [260](#)

Set-AzDiagnosticSetting cmdlet, [125](#)

Set-AzKeyVaultAccessPolicy cmdlet, [286](#)

Set-AzKeyVaultCertificateIssuer cmdlet, [294](#)
Set-AzKeyVaultCertificatePolicy cmdlet, [294](#)
Set-AzKeyVaultSecret cmdlet, [296](#), [298](#)
Set-AzRouteTable cmdlet, [97](#)
Set-AzStorageAccount cmdlet, [261](#)
Set-AzureADGroup cmdlet, [8](#)
Set-AzVirtualNetwork cmdlet, [97](#)
Set-AzVirtualNetworkGatewayConnectionSharedKey
cmdlet, [105](#)
Set-AzVirtualNetworkSubnetConfig cmdlet, [97](#)
Set-AzVmDiskEncryptionExtensions cmdlet, [169](#)
Shared Access Signatures (SAS), [251](#)-[254](#)

- account SAS, [253](#)-[254](#)
- best practices, [251](#)-[252](#)
- tokens, [253](#)-[254](#)
- types of, [251](#)
- user delegation SAS, [252](#)-[253](#)

shared responsibility model, [89](#)
share-level permissions, [259](#)
SharePoint Administrator role, [76](#)
SIEM (Security Information and Event Management),
[212](#)

- sign-in options in Azure AD Connect, [27](#)-[28](#)
- sign-in risk policies, [61](#)-[63](#)
- single sign-on, [15](#)
- site-to-site (S2S) VPNs, [104](#), [108](#)

Skype for Business/Lync Administrator role, [76](#)
SOAR (Security Orchestration, Automation, and
Response), [212](#)
software-protected keys, [299](#)
software updates in Azure App Service, [176](#)

SQL databases. *See* [databases](#)

SQL Server requirements, Azure AD Connect, [16-17](#)

SQL Servers, vulnerability assessment, [199-200](#)

SSL/TLS certificates, configuring, [172-174](#)

SSPR (self-service password reset), [15](#)

enabling, [28-30](#)

Stop-AzKeyVaultCertificateOperation cmdlet, [294](#)

Storage account Contributor role, [245](#)

Storage account Key Operator Service Role, [245](#)

storage accounts

ATP (Advanced Threat Protection) for Azure Storage,
[267-268](#)

authentication with Azure AD, [255-256](#)

Azure Files authentication, [256-261](#)

encryption, [262-267](#)

infrastructure encryption, [264](#)

key management, [263-264](#)

scopes, [264-267](#)

viewing status, [262-263](#)

firewalls, [138-140](#)

key management, [247](#)

rotating keys, [247-250](#)

viewing keys, [248-249](#)

RBAC roles

assigning, [245-247](#)

levels of, [244](#)

list of, [245](#)

SAS (Shared Access Signatures), [251-254](#)

account SAS, [253-254](#)

best practices, [251-252](#)

types of, [251](#)

user delegation SAS, [252-253](#)
stored access policies, [255](#)
types of, [244](#)

Storage Blob Data Contributor role, [245, 256](#)
Storage Blob Data Owner role, [245, 256](#)
Storage Blob Data Reader role, [245, 256](#)
Storage Blob Delegator role, [245, 256](#)
Storage File Data SMB Share Contributor role, [259](#)
Storage File Data SMB Share Elevated Contributor role,
[245, 259](#)
Storage File Data SMB Share Reader role, [245, 259](#)
Storage File SMB Share Contributor role, [245](#)
Storage Queue Data Contributor role, [245, 256](#)
Storage Queue Data Message Processor role, [245, 256](#)
Storage Queue Data Message Sender role, [245, 256](#)
Storage Queue Data Reader role, [245, 256](#)
stored access policies
 for blob containers, [255](#)
 with service SAS, [251](#)

subnets, [91](#)
subscription permissions, [74-79](#)
subscriptions (Azure), transferring, [36-37](#)
system requirements, Azure AD Connect, [15-16](#)
system updates for VMs, [156-159](#)

T

TDE (transparent data encryption), [276-279](#)
Teams Administrator role, [76](#)
Teams Communications Administrator role, [76](#)
Teams Communications Support Engineer role, [76](#)

Teams Communications Support Specialist role, [76](#)
templates for scheduled query rules in Azure Sentinel, [222-223](#)
tenants (Azure), transferring subscriptions, [36-37](#)
threat detection for VMs (virtual machines), [155-156](#)
threat hunting in Azure Sentinel, [231-232](#)
threat protection for SQL, [199](#)
traffic interruptions, [91](#)
transferring subscriptions (Azure), [36-37](#)
transparent data encryption (TDE), [276-279](#)
troubleshooting JIT (Just In Time) VM access, [205](#)

U

unblocking MFA users, [58](#)
Undo-AzKeyVaultCertificateRemoval cmdlet, [294](#)
Undo-AzKeyVaultKeyRemoval cmdlet, [300](#)
Undo-AzKeyVaultSecretRemoval cmdlet, [298](#)
Update-AzKeyVaultCertificate cmdlet, [294](#)
Update-AzKeyVaultKey cmdlet, [300](#)
Update-AzKeyVaultSecret cmdlet, [298](#)
Update-AzStorageAccountADOObjectPassword cmdlet, [259](#)
Update-AzStorageAccountNetworkRuleSet cmdlet, [140](#)
Update-AzureKeyVaultSecret cmdlet, [296](#)
Update Management (in Azure Automation), [156-159](#)
updates
 software updates in Azure App Service, [176](#)
 system updates for VMs, [156-159](#)
UPN suffixes, nonroutable domains and, [25-27](#)
User Access Administrator role, [77](#)
User Account Administrator role, [76](#)

user delegation SAS, [251-253](#)
user principal objects, [2](#)
user resource permissions, viewing, [84-85](#)
user-risk policies, [61-63](#)
users, [13-15](#)
 assigning application access, [67-70](#)
 assigning to roles, [78-79](#)
 creating, [13-14](#)
 deleting, [14](#)
 recovering, [14](#)
 viewing role assignments, [77-78](#)

V

viewing
 audit logs, [271-273](#)
 Azure Monitor alerts, [188](#)
 blob encryption status, [262-263](#)
 endpoint protection, [151-154](#)
 resource audit history, [38-40](#)
 service principal list, [3](#)
 storage account access keys, [248-249](#)
 user resource permissions, [84-85](#)
 user role assignments, [77-78](#)
virtual network gateways, [91, 104-108](#)
 authentication, [104-106](#)
 ExpressRoute encryption, [106-107](#)
 point-to-site (P2S), [107-108](#)
 site-to-site (S2S), [108](#)
 types of, [104](#)
VMs (virtual machines)

disk encryption, [168-169](#)
endpoint security, [151-156](#)
system updates, [156-159](#)

VNets (virtual networks)
for Azure Key Vault, [283-285](#)
configuring, [90-95](#)
NAT (network address translation), [100-103](#)
peering, [97-100](#)
routing, [95-97](#)
security, [104-108](#)
service endpoints, [145-147](#)

VNet-to-VNet VPNs, [104](#)
VPN gateways, [91, 104-108](#)
authentication, [104-106](#)
ExpressRoute encryption, [106-107](#)
point-to-site (P2S), [107-108](#)
site-to-site (S2S), [108](#)
types of, [104](#)

vulnerability assessment with Azure Security Center, [196-200](#)
vulnerability management, [164-165](#)

W-Z

WAF (Web Application Firewall)
Azure Front Door integration, [133](#)
configuring on Azure Application Gateway, [133-135](#)
inbound HTTP/S protection, [118, 122](#)

Windows Hello for Business, [34](#)
workbooks in Azure Sentinel, [229-230](#)
workspaces in Azure Sentinel, [213](#)

x509 certificates, managing in Azure Key Vault, [288-296](#)

Exam Ref AZ-500: Microsoft Azure Security Technologies

LIST OF URLs

Chapter 1

<https://docs.microsoft.com/en-us/azure/active-directory/develop/app-objects-and-service-principals>

<https://docs.microsoft.com/en-us/powershell/azure/create-azure-service-principal-azureps>

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

<https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-groups-view-azure-portal>

<https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-manage-groups>

<https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-groups-membership-azure-portal>

<https://docs.microsoft.com/en-us/powershell/azure/active-directory/new-user-sample>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/tutorial-enable-sspr-writeback>

<https://secure.aadcdn.microsoftonline-p.com>

<https://docs.microsoft.com/en-us/azure/active-directory/connect/active-directory-aadconnect-user-signin>

<https://passwordreset.microsoftonline.com>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-sspr-howitworks>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-authentication-methods>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/active-directory-certificate-based-authentication-get-started>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-authentication-passwordless>

<https://docs.microsoft.com/en-us/azure/active-directory/b2b/add-users-administrator>

<https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-how-subscriptions-associated-directory>

<https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-how-to-use-audit-log>

<https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-how-to-perform-security-review>

<https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-configure>

<https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/subscription-requirements>

<https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-security-wizard>

<https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/overview>

<https://docs.microsoft.com/en-us/office365/admin/security-and-compliance/multi-factor-authentication-plan>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-mfa-howitworks>

<https://docs.microsoft.com/en-us/office365/admin/security-and-compliance/setup-multi-factor-authentication>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-mfa-mfasettings>

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-mfa-reporting>

<https://docs.microsoft.com/en-us/azure/active-directory/identity-protection/overview-identity-protection>

<https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-register-app>

<https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/what-is-access-management>

<https://docs.microsoft.com/en-us/azure/active-directory/manage-apps/methods-for-assigning-users->

and-groups

<https://docs.microsoft.com/en-us/azure/active-directory/develop/v2-permissions-and-consent>

<https://docs.microsoft.com/en-us/azure/api-management/api-management-policies>

<https://docs.microsoft.com/en-us/azure/role-based-access-control/overview>

<https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles>

<https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/roles-concept-delegation>

<https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-manage-roles-portal>

<https://docs.microsoft.com/en-us/rest/api/authorization/permissions/listforresourcegroup>

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

<https://docs.microsoft.com/en-us/azure/security/fundamentals/identity-management-best-practices>

<https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles>

<https://docs.microsoft.com/en-us/azure/role-based-access-control/role-definitions#management-and-data-operations>

<https://docs.microsoft.com/en-us/azure/role-based-access-control/check-access>

Chapter 2

<https://shell.azure.com>

<http://aka.ms/az500mfa>

<http://aka.ms/az500vpnsku>

<http://aka.ms/az500vpnexpressroute>

<https://aka.ms/az500s2sdevices>

<https://aka.ms/az500s2svpn>

<http://aka.ms/wafdecisionflow>

<https://owasp.org/www-project-top-ten>

<https://aka.ms/az500wafag>

<https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview>

<https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoint-policies-overview>

<http://aka.ms/az500DDoS>

<https://docs.microsoft.com/en-us/azure/aks/upgrade-cluster>

<https://kubernetes.io/docs/concepts/configuration/security>

<http://aka.ms/az500kvfw>

<http://aka.ms/az500ADELinux>

<http://aka.ms/az500ADEWin>

<https://aka.ms/az500AppCertificates>

<http://aka.ms/az500AppServiceAuth>

<https://docs.microsoft.com/en-us/powerapps/maker/common-data-service/data-platform-intro>

<https://github.com/projectkudu/kudu/wiki/Kudu-console>

Chapter 3

<https://aka.ms/ASICommunity>

Chapter 4

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

<https://docs.microsoft.com/en-us/azure/storage/common/storage-account-keys-manage>

<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

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

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

<https://docs.microsoft.com/en-us/rest/api/storageservices/define-stored-access-policy>

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

<https://docs.microsoft.com/en-us/azure/storage/files/storage-files-identity-auth-active-directory-domain-service-enable>

<https://docs.microsoft.com/en-us/azure/storage/common/storage-service-encryption>

<https://docs.microsoft.com/en-us/azure/storage/common/infrastructure-encryption-enable>

<https://docs.microsoft.com/en-us/azure/storage/blobs/encryption-scope-manage>

<https://docs.microsoft.com/en-us/azure/storage/common/storage-advanced-threat-protection?tabs=azure-security-center>

<https://docs.microsoft.com/en-us/azure/azure-sql/database/logins-create-manage>

<https://docs.microsoft.com/en-us/azure/azure-sql/database/auditing-overview>

<https://docs.microsoft.com/en-us/azure/azure-sql/database/threat-detection-overview>

<https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/sql-server-encryption?view=azuresqldb-current>

<https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine?view=sql-server-ver15>

<https://docs.microsoft.com/en-us/azure/key-vault/general/overview-security>

<https://docs.microsoft.com/en-us/azure/key-vault/general/network-security>

<https://docs.microsoft.com/en-us/azure/key-vault/general/group-permissions-for-apps>

<http://docs.microsoft.com/en-us/azure/key-vault/general/secure-your-key-vault>

<https://docs.microsoft.com/en-us/azure/key-vault/certificates/about-certificates>

<https://mykeyvault.vault.azure.net/certificates/mycert1/create?api-version={api-version}>

<https://docs.microsoft.com/en-us/azure/key-vault/certificates/certificate-scenarios>

<https://docs.microsoft.com/en-us/azure/key-vault/secrets>

<https://docs.microsoft.com/en-us/azure/key-vault/keys>

<https://docs.microsoft.com/en-us/azure/key-vault/secrets/tutorial-rotation>

<https://docs.microsoft.com/en-us/azure/key-vault/general/backup>

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.

```
$servicePrincipal = New-AzADServicePrincipal -DisplayName "ExampleServiceprincipal"
```

```
Get-AzAdServicePrincipal | format-table
```

```
$servicePrincipal = New-AzADServicePrincipal -DisplayName "ExampleServiceprincipal"  
New-AzRoleAssignment -RoleDefinitionName "Reader" -ApplicationId $servicePrincipal.  
ApplicationId
```

```
Az ad group create --display-name "Accounting Users" --mail-nickname "accounting.users"
```

```
az ad group member add --group "Accounting Users" --member-id ac5ebbf8-22c7-4381-b91d-  
12aeb3093413-----
```

```
az ad user show --id delta.user@tailwindtraders.net
```

```
Get-ADUser -Filter {UserPrincipalName -like "*@epistemicus.internal"} -SearchBase  
"DC=epistemicus,DC=internal" |  
ForEach-Object {  
$UPN =  
$_ .UserPrincipalName.Replace("epistemicus.internal","epistemicus.onmicrosoft.com")  
Set-ADUser $_ -UserPrincipalName $UPN  
}-----
```

```
Get-AzRoleDefinition "Contributor" | FL Actions, NotActions
```

```
$AZ500Subnet = New-AzVirtualNetworkSubnetConfig -Name AZ500Subnet -AddressPrefix  
"10.3.0.0/24"  
New-AzVirtualNetwork -Name AZ500VirtualNetwork -ResourceGroupName ContosoCST -Location  
centralus -AddressPrefix "10.3.0.0/16" -Subnet $AZ500Subnet
```

```
New-AzVm  
  -ResourceGroupName "ContosoCST"  
  -Location "East US"  
  -VirtualNetworkName "AZ500VirtualNetwork"  
  -SubnetName "AZ500Subnet"  
  -Name "AZ500VM"
```

```
$routeTableAZ500 = New-AzRouteTable `  
    -Name 'AZ500RouteTable' `  
    -ResourceGroupName ContosoCST `  
    -Location EastUS
```

```
Get-AzRouteTable  
  -ResourceGroupName "ContosoCST"  
  -Name "AZ500RouteTable"  
  | Add-AzRouteConfig  
    -Name "ToAZ500Subnet"  
    -AddressPrefix 10.0.1.0/24  
    -NextHopType "MyVirtualAppliance"  
    -NextHopIpAddress 10.0.2.4  
  | Set-AzRouteTable
```

```
$virtualNetwork | Set-AzVirtualNetwork  
Set-AzVirtualNetworkSubnetConfig  
    -VirtualNetwork $virtualNetwork  
    -Name 'CustomAZ500Subnet'  
    -AddressPrefix 10.0.0.0/24  
    -RouteTable $routeTableAZ500 |  
Set-AzVirtualNetwork
```

```
Add-AzVirtualNetworkPeering -Name 'NameOfTheVNetPeering' -VirtualNetwork SourceVNet  
-RemoteVirtualNetworkId RemoteVNet -----
```

```
New-AzNatGateway -ResourceGroupName "AZ500RG" -Name "nat_gt" -IdleTimeoutInMinutes 4  
-Sku "Standard" -Location "eastus2" -PublicIpAddress PublicIPAddressName
```

```
New-AzNetworkSecurityGroup -Name "AZ500NSG" -ResourceGroupName "AZ500RG" -Location  
"westus"-----
```

```
$MyRule1 = New-AzNetworkSecurityRuleConfig -Name ftp-rule -Description "Allow FTP"  
-Access Allow -Protocol Tcp -Direction Inbound -Priority 100 -SourceAddressPrefix *  
-SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 21
```

```
New-AzApplicationSecurityGroup -ResourceGroupName "MyRG" -Name "MyASG" -Location "West  
US"
```

```
New-AzFirewall -Name "azFw" -ResourceGroupName MyRG -Location centralus -VirtualNetwork  
MyVNet -PublicIpAddress MyPubIP
```

```
$MyAppRule = New-AzFirewallApplicationRule -Name AllowBing -SourceAddress * `  
    -Protocol http, https -TargetFqdn www.bing.com  
$AppCollectionRule = New-AzFirewallApplicationRuleCollection -Name App-Coll01 `  
    -Priority 100 -ActionType Allow -Rule $MyAppRule  
$Azfw.ApplicationRuleCollections = $AppRuleCollection  
Set-AzFirewall -AzureFirewall $Azfw
```

```
New-AzFirewallNetworkRule -Name "DNSOutbound" -Protocol UDP -SourceAddress  
"10.30.0.0/24" -DestinationAddress IP_of_the_DNSSErver -DestinationPort 53
```

```
Set-AzDiagnosticSetting -ResourceId /subscriptions/<subscriptionId>/  
resourceGroups/<resource group name>/providers/Microsoft.Network/  
azureFirewalls/<Firewall name> `  
-StorageAccountId /subscriptions/<subscriptionId>/resourceGroups/<resource group  
name>/providers/Microsoft.Storage/storageAccounts/<storage account name> `  
-Enabled $true
```

```
AzureDiagnostics  
| where Category == "AzureFirewallNetworkRule"
```

Mandatory rule. Cannot be disabled. Inbound Anomaly Score Exceeded (Total Inbound Score: 5 - SQLI=0,XSS=0,RFI=0,LFI=0,RCE=0,PHP=0,HTTP=0,SESS=0): Missing User Agent Header;
individual paranoia level scores: 3, 2, 0, 0

```
Update-AzStorageAccountNetworkRuleSet -ResourceGroupName "MyRG" -Name "mystorage"  
-DefaultAction Deny
```

```
$storagea = New-AzStorageAccount -ResourceGroupName ContosoResourceGroup -Name  
fabrikavaultlogs -Type Standard_LRS -Location 'East US'  
$kvault = Get-AzKeyVault -VaultName 'ContosoKeyVault'  
Set-AzDiagnosticSetting -ResourceId $kvault.ResourceId -StorageAccountId $storagea.Id  
-Enabled $true -Category AuditEvent
```

```
az ad sp create-for-rbac --skip-assignment --name myAKSClusterSP
```

```
Set-AzKeyVaultAccessPolicy -VaultName "<your -keyvault-name>" -ResourceGroupName  
"MyResourceGroup" -EnabledForDiskEncryption
```

```
$AKeyVault = Get-AzKeyVault -VaultName MyAKV -ResourceGroupName MyRG  
Set-AzVMDiskEncryptionExtension -ResourceGroupName MyRG -VMName MyVM  
-DiskEncryptionKeyVaultUrl $AKeyVault.VaultUri -DiskEncryptionKeyId $AKeyVault.  
ResourceId-----
```

```
OsVolumeEncrypted      : Encrypted
DataVolumesEncrypted   : NoDiskFound
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.
DiskEncryptionSettings
ProgressMessage        : Provisioning succeeded-----
```

```
SecurityEvent | where EventID == 4799
```

```
SecurityEvent | where EventID == 4688 and CommandLine contains "cm"
```

```
SecurityEvent | where EventID == 4624 and Account contains "anonymous logon" and  
LogonType == 3
```

```
AzureActivity  
| where OperationNameValue contains "Microsoft.Compute/virtualMachines/delete"
```

```
$storageAccountKey =  
    (Get-AzStorageAccountKey  
        -ResourceGroupName <resource-group>  
        -Name <storage-account>).Value[0]
```

```
az storage account keys list \
--resource-group <resource-group> \
--account-name <storage-account>
```

```
New-AzStorageAccountKey -ResourceGroupName <resource-group>
-Name <storage-account>
-KeyName key1
```

```
az storage account keys renew \
--resource-group <resource-group> \
--account-name <storage-account>
--key primary
```

```
$ctx = New-AzStorageContext -StorageAccountName <storage-account> -UseConnectedAccount
```

```
New-AzStorageContainerSASToken -Context $ctx  
  -Name <container>  
  -Permission racwdl  
  -ExpiryTime <date-time>
```

```
New-AzStorageBlobSASToken -Context $ctx  
  -Container <container>  
  -Blob <blob>  
  -Permission racwd  
  -ExpiryTime <date-time>  
  -FullUri
```

```
Revoke-AzStorageAccountUserDelegationKeys -ResourceGroupName <resource-group> `  
    -StorageAccountName <storage-account>  
-----
```

```
az storage container generate-sas \  
    --account-name <storage-account> \  
    --name <container> \  
    --permissions acdlrw \  
    --expiry <date-time> \  
    --auth-mode login \  
    --as-user
```

```
az storage blob generate-sas \
--account-name <storage-account> \
--container-name <container> \
--name <blob> \
--permissions acdrw \
--expiry <date-time> \
--auth-mode login \
--as-user
--full-uri
```

```
az storage account revoke-delegation-keys \
--name <storage-account> \
--resource-group <resource-group>
```

```
<?xml version="1.0" encoding="utf-8"?>
<SignedIdentifiers>
  <SignedIdentifier>
    <Id>unique-64-char-value</Id>
    <AccessPolicy>
      <Start>start-time</Start>
      <Expiry>expiry-time</Expiry>
      <Permission>abbreviated-permission-list</Permission>
    </AccessPolicy>
  </SignedIdentifier>
</SignedIdentifiers>
```

```
$ResourceGroupName = "<resource-group-name-here>"  
$StorageAccountName = "<storage-account-name-here>"  
New-AzStorageAccountKey -ResourceGroupName $ResourceGroupName -Name $StorageAccountName  
-KeyName kerb1  
Get-AzStorageAccountKey -ResourceGroupName $ResourceGroupName -Name $StorageAccountName  
-ListKerbKey | where-object{$_._KeyName -contains "kerb1"}-----
```

```
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>"`-----
```

```
Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
```

```
Import-Module -Name AzFilesHybrid
```

```
$SubscriptionId = "<your-subscription-id-here>"  
$ResourceGroupName = "<resource-group-name-here>"  
$StorageAccountName = "<storage-account-name-here>"  
Select-AzSubscription -SubscriptionId $SubscriptionId
```

```
Join-AzStorageAccountForAuth ` 
    -ResourceGroupName $ResourceGroupName ` 
    -StorageAccountName $StorageAccountName ` 
    -DomainAccountType "<ComputerAccount|ServiceLogonAccount>" ` 
    -OrganizationalUnitDistinguishedName "<ou-distinguishedname-here>" #  
-----
```

```
Debug-AzStorageAccountAuth -StorageAccountName $StorageAccountName -ResourceGroupName  
$ResourceGroupName -Verbose
```

```
# Update the password of the AD DS account registered for the storage account
# You may use either kerb1 or kerb2
Update-AzStorageAccountADObjectPassword `

    -RotateToKerbKey kerb2 `

    -ResourceGroupName "<your-resource-group-name-here>" `

    -StorageAccountName "<your-storage-account-name-here>"`
```

```
Set-AzStorageAccount -ResourceGroupName "FilesRG" `  
-Name "tailwind-files" `  
-EnableAzureActiveDirectoryDomainServicesForFile $true
```

```
az storage account update -n tailwind-files -g FilesRG --enable-files-adds $true
```

```
$account = Get-AzStorageAccount -ResourceGroupName <resource-group> `  
    -Name <storage-account>  
$blob = Get-AzStorageBlob -Context $account.Context `  
    -Container <container> `  
    -Blob <blob>  
$blob.ICloudBlob.Properties.IsServerEncrypted-----
```

```
az storage blob show \  
    --account-name <storage-account> \  
    --container-name <container> \  
    --name <blob> \  
    --query "properties.serverEncrypted"
```

```
Set-AzKeyVaultAccessPolicy -VaultName <your-key-vault-name> -PermissionsToKeys  
<permissions-to-keys> -PermissionsToSecrets <permissions-to-secrets>  
-PermissionsToCertificates <permissions-to-certificates> -ObjectId <Id>  
-----
```

```
az keyvault set-policy -n <your-unique-keyvault-name> --spn <ApplicationID-of-your-  
service-principal> --secret-permissions <secret-permissions> --key-permissions <key-  
permissions> --certificate-permissions <certificate-permissions>
```

```
{  
  "policy": {  
    "x509_props": {  
      "subject": "CN=TailwindCertSubject1"  
    },  
    "issuer": {  
      "name": "mydigicert",  
      "cty": "OV-SSL",  
    }  
  }  
}
```

```
$secretvalue = ConvertTo-SecureString 'Omega' -AsPlainText -Force
$secret = Set-AzKeyVaultSecret -VaultName 'TailwindKV' -Name 'Omega' -SecretValue
$secretvalue
```