Microsoft

Designing Microsoft Azure Infrastructure Solutions

Exam Ref AZ-305

Ashish Agrawal
Gurvinder Singh
Avinash Bhavsar
Mohamed Sabir Sopariwala

Exam Ref AZ-305 Designing Microsoft Azure Infrastructure Solutions

Ashish Agrawal
Gurvinder Singh
Avinash Bhavsar
Mohamed Sabir Sopariwala

Microsoft

Exam Ref AZ-305 Designing Microsoft Azure Infrastructure Solutions

Published with the authorization of Microsoft Corporation by:

Pearson Education, Inc.

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: 978-0-13-787878-9
ISBN-10: 0-13-787878-8

Library of Congress Control Number: 2022945426

ScoutAutomatedPrintCode

TRADEMARKS

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

WARNING AND DISCLAIMER

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an "as is" basis. The author, the publisher, and Microsoft Corporation shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the programs accompanying it.

SPECIAL SALES

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

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

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

CREDITS

EDITOR-IN-CHIEF

Brett Bartow

EXECUTIVE EDITOR

Loretta Yates

SPONSORING EDITOR

Charvi Arora

DEVELOPMENT EDITOR

Kate Shoup

MANAGING EDITOR

Sandra Schroeder

SENIOR PROJECT EDITOR
Tracey Croom

TECHNICAL EDITOR
Thomas Palathra

COPY EDITOR
Scout Festa

INDEXER
Timothy Wright

PROOFREADER
Donna E. Mulder

EDITORIAL ASSISTANT
Cindy Teeters

COVER DESIGNER
Twist Creative, Seattle

COMPOSITOR
codeMantra

Pearson’s Commitment to Diversity, Equity, and Inclusion

Pearson is dedicated to creating bias-free content that reflects the diversity of all learners. We embrace the many dimensions of diversity, including but not limited to race, ethnicity, gender, socioeconomic status, ability, age, sexual orientation, and religious or political beliefs.

Education is a powerful force for equity and change in our world. It has the potential to deliver opportunities that improve lives and enable economic mobility. As we work with authors to create content for every product and service, we acknowledge our responsibility to demonstrate inclusivity and incorporate diverse scholarship so that everyone can achieve their potential through learning. As the world’s leading learning company, we have a duty to help drive change and live up to our purpose to help more people create a better life for themselves and to create a better world.

Our ambition is to purposefully contribute to a world where:

- Everyone has an equitable and lifelong opportunity to succeed through learning.
- Our educational products and services are inclusive and represent the rich diversity of learners.
- Our educational content accurately reflects the histories and experiences of the learners we serve.
- Our educational content prompts deeper discussions with learners and motivates them to expand their own learning (and worldview).

While we work hard to present unbiased content, we want to hear from you about any concerns or needs with this Pearson product so that we can investigate and address them.

- Please contact us with concerns about any potential bias at https://www.pearson.com/report-bias.html.

Contents at a glance

Contents

Introduction

The purpose of the AZ-305 certification exam is to test your knowledge and understanding of the Microsoft Azure platform, including networking, virtualization, identity, security, business continuity, disaster recovery, data platforms, and governance. The exam is targeted toward Azure Solutions Architects, and includes coverage of advising the stakeholders responsible for translating business requirements into secure, scalable, and reliable cloud solutions. This book provides comprehensive coverage of exam domain objectives, including in-depth explanations and demonstrations of real-world design scenarios. Designed for modern IT professionals, this book focuses on the critical thinking and decision-making acumen needed for success at the Microsoft Certified Expert level.

While we’ve made every effort possible to ensure that the information in this book is accurate, Azure is rapidly evolving. So there’s a chance that some of the screens in the Azure Portal will have changed slightly since this book was written, which means some figures in this book might look different from what you see on your screen. It’s also possible that other minor interface changes have taken place, such as name changes and so on.

Azure supports a wide range of programming languages, frameworks, databases, and services. Given this, IT professionals must learn a vast range of technical topics in a short span of time. There is an overabundance of content available, which makes it difficult to find just enough study material to prepare for the AZ-305 exam—no more and no less. This book offers prescriptive guidance for people preparing for this exam.

This book covers every major topic area found on the exam, but it does not cover every exam question. Only the Microsoft exam team has access to the exam questions. Moreover, 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 More Info links found throughout the text to access more information. Take the time to research and study those topics. Great information is available on Microsoft Learn, docs.microsoft.com/azure, TechNet, and blogs and forums.

Organization of this book

This book is organized to reflect the "Skills measured" list published for the exam. 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 résumé 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 *microsoft.com/learn*. Microsoft Official Practice Tests are available for many exams at *aka.ms/practicetests*.

Note that this Exam Ref is based on publicly available information about the exam and on the authors' 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 *microsoft.com/learn*.

Check back often to see what is new!

Quick access to online references

Throughout this book are addresses to webpages that the authors have 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/ExamRefAZ305AzureArchitectDesign/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/ExamRefAZ305AzureArchitectDesign/errata*.

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

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

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

Stay in touch

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

About the authors

ASHISH AGRAWAL is a qualified technocrat, offering over two decades of multifaceted experience as a Cloud Engineering and transformation leader, trusted advisor, developer, consultant, and Enterprise Cloud Architect. He drives a profound influence in the cloud technology landscape with provocative thought leadership and communicates his ideas with clarity and passion. He has deep, hands-on technical expertise, having spearheaded numerous successful cloud engagements for global Fortune 500 companies in advisory, presales, consulting, architecture, leadership, and delivery execution, and he has played technology leadership roles in large, complex, cross-functional, and multi-enterprise project teams.

GURVINDER SINGH is a Microsoft Certified Azure Solutions Architect with 15 years of diversified IT experience working with the Microsoft Technology stack. In the past several years, Gurvinder has been guiding large enterprises in the transformation of legacy applications into cloud-native architecture with a focus on migration to the Microsoft Azure platform. He is extremely passionate about technology, especially with the Microsoft Azure platform (PaaS, IaaS, and Serverless).

AVINASH BHAVSAR is a Microsoft Certified Azure Professional with about

19 years of hands-on experience in all facets of cloud computing, such as discovery, assessment, cloud foundation build, datacenter transformation, cloud-native application development for Azure, and migration of applications and databases from on-premises to the Azure platform. He has an extensive application development background, which includes architecture, design, development, continuous integration, and continuous delivery to the Azure platform (IaaS, PaaS, and Serverless).

MOHAMED SABIR SOPARIWALA is a Senior Architect with key expertise in cloud computing. He is a Microsoft Certified Azure Solutions Architect working as a Cloud Solution Architect on cloud transformation and adoption engagements, helping customers and partners in their cloud and digital transformation journey with the effective use of a broad and continuously changing technology landscape to help them to meet their business goals. His areas of expertise include cloud-native architecture, serverless architecture, application modernization, cloud adoption, service-oriented architecture, performance engineering, and custom application development architecture and design.

Acknowledgments

Ashish Agrawal Let me start by admitting that authoring a book is a much harder project than I had imagined. This project would not have been possible without tremendous support from my family, who allowed me to devote many of my weekends and evenings to this book. In particular, I must acknowledge my wonderful and ever-patient wife, Swapna, for putting up with my crazy schedule, and my kids, Devansh and Yug, for being a continuous energy source. Their love and inspiration were instrumental in the success of this project. I would also like to thank my parents, Shobha and Omprakash Agrawal, for their encouragement and blessings. And my guru and inspiration, Sunil Poddar, always instilled in me the work ethic and dedication needed to get projects like this one across the finish line.

Next, I want to thank Corrado Azzarita, Mani Gopalakrishnan, Eric Nelson, Mike Halfman, Tim Regan, Steve Mintert, Chris Burgoyne, Manish Khatri, Tami Moore, and other leaders at Kraft Heinz company for their encouragement and motivation.

I am grateful to Microsoft Press at Pearson for taking on this project. Thank you, Loretta, for the opportunity to contribute to this book, and Charvi, for your excellent project management through conceptualization to publishing. Writing this book has been a fantastic experience. I also want to thank my amazing co-authors for their determination and outstanding teamwork. Finally, my special thanks go to our reviewers and editors for walking through the content with a fine-tooth comb and sharing incredible feedback.

Avinash Bhavsar First and foremost, I would like to thank my parents for bringing me up during challenging times and making me capable of seeing this day. Thanks to them and the Almighty for the blessings. Special thanks to my wonderful wife, Jyotsna, for her support and inspiration. A big thank you to my lovely kids, Atharva and Aayush, for their love and inspiration. Huge thanks to Loretta Yates and Charvi Arora for their support during this journey. Finally, I would like to thank my co-authors, Microsoft Press, and the Pearson team for the opportunity to work on this project. Cheers and happy reading.

Gurvinder Singh I am indebted to Microsoft Press for the opportunity to contribute to this book. All my co-authors are well known for their professional prowess and in-depth knowledge of the Microsoft Azure platform and need no introduction. A big thank you, too, to the reviewers for their well-coordinated efforts and due diligence, from the conceptualization to the publication of this volume. I am indeed grateful to the entire Pearson team, especially Loretta Yates and Charvi Arora, for their cooperation, support, and patience throughout this journey.

I am indeed grateful to my mom, Daljeet Kaur; my wife, Jaspreet Kaur; and our daughter, Amritleen Kaur, for the tremendous encouragement that helped me walk the insanely tight schedule of deadlines.

Finally, I submit myself in reverence to Guru Nanak, the great spiritual guru, whose blessings endowed an incredibly small and nondescript individual like me with wisdom and opportunity.

Mohamed Sabir Sopariwala I start by thanking Almighty for the countless blessings upon me. I would like to thank my parents for all the love, support, and guidance throughout my life. Authoring a book indeed requires a lot of commitment and time. I thank my wife, Aasma; my daughter, Fatima; and my son, Mohammed Hamzah, for their support and patience while I spent numerous late evenings and weekends on this book.

I am grateful to Microsoft Press and the Pearson team for giving me the opportunity to be part of this book as an author. A huge shout-out to my co-authors, Gurvinder Singh, Ashish Agrawal, and Avinash Bhavsar, and to our reviewers for their great collaboration. Finally, thanks to Loretta Yates and Charvi Arora for steering the team and supporting us throughout this journey.

Chapter 1

Design identity, governance, and monitoring solutions

While designing an IT solution, the obvious focus is on the business and function requirements. A person with a business lens focuses on ensuring that the solution addresses the business and function needs. But a person with a CIO, CISO, architect, or IT operations lens has a responsibility to ensure that the IT solution runs at the expected level of performance, is secure, is traceable, is compliant with regulatory and organization policies, and offers the optimal cost of ownership. These non-functional requirements (NFRs) are of utmost importance for an enterprise.

The Microsoft Azure Well-Architected Framework (WAF) is a set of guidelines for architects to address these NFRs for workloads targeted to be deployed in Azure. It articulates five pillars of architecture design for building a good-quality workload running in Azure and achieving excellence:

- Reliability
- Security
- Cost optimization
- Operational excellence
- Performance efficiency

The Microsoft Cloud Adoption Framework (CAF) provides documentation, tools, templates, guidance, and best practices to help enterprises in their cloud-adoption journey. The CAF meets an enterprise wherever it is in this journey. It identifies seven phases in the cloud-adoption journey and organizes all documentation, tools, templates, guidance, and best practices around these phases:

- **Strategy** Identify your business justification and expected outcome in terms of the business value of the cloud-adoption journey.
- **Plan** Create and agree on an actionable plan to drive desired business outcomes.
- **Ready** Set up a landing zone in a cloud environment where the workload will be deployed. These workloads can be greenfield workloads or existing workloads migrated to Azure.
- **Adopt** There are two possibilities for the workloads to be deployed in Azure:
    - **Migrate** Here, you migrate and modernize existing workloads in Azure.
    - **Innovate** Develop new cloud-native solutions in Azure or in a hybrid environment.
- **Govern** This phase deals with governance of workloads and the environment as a whole in Azure or in a hybrid environment.
- **Manage** This deals with IT operations in Azure. While workloads are running in Azure, they require management. Because the Azure operation management tool can be extended to hybrid scenarios, this phase also involves having a common set of tools for operation management, whether workloads are in Azure or in a hybrid environment.
- **Organize** This phase provides guidance on how to organize your teams and on what roles are needed to support your organization’s cloud-adoption journey.

The topic of this chapter is very much aligned with the Microsoft CAF and the Azure WAF, which could both be discussed in great depth. However, because the focus of this book is to prepare the reader for AZ-305 certification, this chapter discusses only the skills indicated in the certification curriculum.

More Info **CAF and WAF**

If you are interested in learning more about CAF and WAF, see the

Microsoft documentation at *https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/* and *https://learn.microsoft.com/en-us/azure/architecture/framework/*, respectively.

Skills covered in this chapter:

- Skill 1.1: Design a solution for logging and monitoring
- Skill 1.2: Design authentication and authorization solutions
- Skill 1.3: Design governance
- Skill 1.4: Design identities and access for applications

Skill 1.1: Design a solution for logging and monitoring

IT operations must have insights into the health, performance, compliance, and cost of the workloads for which they are accountable and responsible. Some common scenarios that IT operations deal with are as follows:

- Safeguarding IT systems and the health of applications
- Keeping track of IT systems and the availability of applications
- Monitoring system performance and ensuring adequate capacity during peak times
- Guaranteeing that the system meets service-level agreements (SLAs) with internal or external customers
- Securing systems, users, and their data
- Auditing for internal and regulatory compliance
- Managing issues from the time they are reported to their resolution, identifying their root cause and resolving them

Designing and implementing the correct level of logging and monitoring, and their integration across IT systems and applications, is key to helping operations teams efficiently monitor, detect, and respond to anomalies and ensuring systems run with expected criteria of reliability, availability, performance, and cost.

This section covers how to:

- Design a log routing solution
- Recommend an appropriate level of logging
- Recommend monitoring tools for a solution

Design a log routing solution

While discussing log routing solutions, it helps to understand the different types of logs available on the Azure platform. You can use a combination of Azure platform logs to comprehensively diagnose and audit workloads running in Azure. These logs are generated automatically. In some cases, you might need to forward these logs to various destinations, such as a third-party system, a long-term retention location, and so on.

The Azure platform generates three types of platform logs, at different layers:

- **Resource logs** These are generated at the Azure resources layer by Azure resources such as Azure Key Vault, Azure Cosmos DB, virtual machines (VMs), and so on. They provide insights into operations performed within Azure resources. The contents of resource logs vary depending on the type of Azure service or resource that generated them. Resource logs are not collected by default. You must configure diagnostic settings for each Azure resource to send resource logs to one or more destinations.

  Resource logs can be routed to any of the following:

  - **Azure Log Analytics workspace** Sending resource logs to an Azure Log Analytics workspace enables you to perform advanced analytics on the logs in Azure Monitor using log queries and to send alerts. For example, you can write complex queries in KQL to perform analysis and obtain insights into log data. You can also write complex KQL queries for alert conditions and then configure log alerts for these conditions. Finally, you can analyze resource log data in correlation with monitoring metrics and logs collected by

  Azure Monitor.

  - **Azure Storage account** To archive resource logs, you can send them to an Azure Storage account for long-term retention.
  - **Azure Event Hub** By routing logs to Event Hub, you can forward them to a third-party system or custom solution, such as an external third-party security information and event management (SIEM) or monitoring tool.
  - **Partner solution** At the time of this writing, there are a few Microsoft partners who have developed solutions that are integrated with Azure. Partner solutions are available through the Azure Marketplace and can be deployed in Azure. You can configure your diagnostic settings to forward resource logs to these partner solutions.

- **Activity logs** Activity logs are generated at the subscription layer. Each subscription has a single activity log that provides insights into administration and management operations performed on each resource on the subscription. You can use these activity logs to track administration and management activities on a resource to determine what operation was performed, who initiated or performed the operation, when the operation was performed, and the status of the operation. Activity logs are retained for 90 days and then deleted.

  Service health logs and metrics provide visibility into the health of an Azure service in the subscription on which your application workloads are running or relying. Service health log records are stored within the activity log.

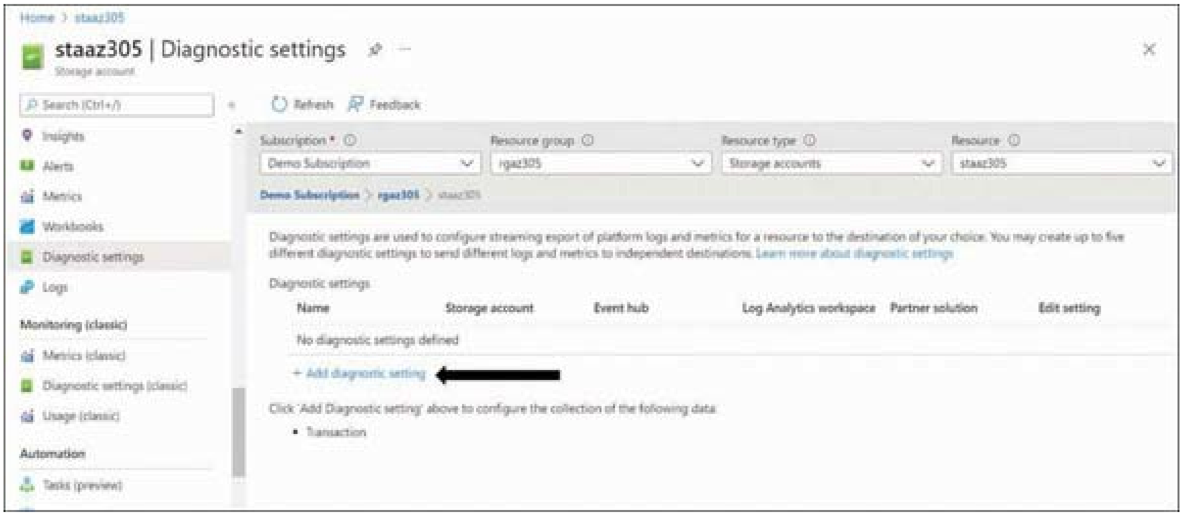
  As with resource logs, you can use diagnostic settings to forward activity logs to an Azure Log Analytics workspace, an Azure Storage account, or the Azure Event Hub. For example, if you need to retain activity logs for more than 90 days, you could forward them to an Azure Storage account.

- **Azure Active Directory (AAD) logs** These logs are generated at the Azure tenant layer. They provide insights into sign-in activities and maintain an audit trail of the changes made in that specific AAD tenant. There are three types of AAD activity logs.

- **Sign-in logs** These logs help track user sign-ins. In this way, you can identify which users are accessing which resources, and how they are accessing those resources, to capture user patterns and behaviors.
- **Audit logs** These logs trace changes made to the tenant object, such as the addition or removal of users, groups, and applications.
- **Provisioning logs** These logs trace the activities of provisioning services—for example, the creation of users in SaaS applications like ServiceNow, Salesforce, and so on.

  As with resource logs, you can forward AAD logs to an Azure Log Analytics work-space, an Azure Storage account, or the Azure Event Hub.

Now that you have an understanding of Azure Platform logs, you're ready to see an example of these logs in action. Figures 1-1 and 1-2 show how to configure diagnostic settings to send storage account resource logs to a Log Analytics workspace. This enables Azure Monitor log features that help with querying and analyzing logs using Kusto Query Language (KQL). Figure 1-3 shows the querying capability in the Log Analytics workspace.



Subscription * : Demo Subscription
Resource group : rgaz305
Resource type : Storage accounts
Resource : staaz305

Demo Subscription > rgaz305 > staaz305

Diagnostic settings are used to configure streaming export of platform logs and metrics for a resource to the destination of your choice. You may create up to five different diagnostic settings to send different logs and metrics to independent destinations. Learn more about diagnostic settings

Diagnostic settings

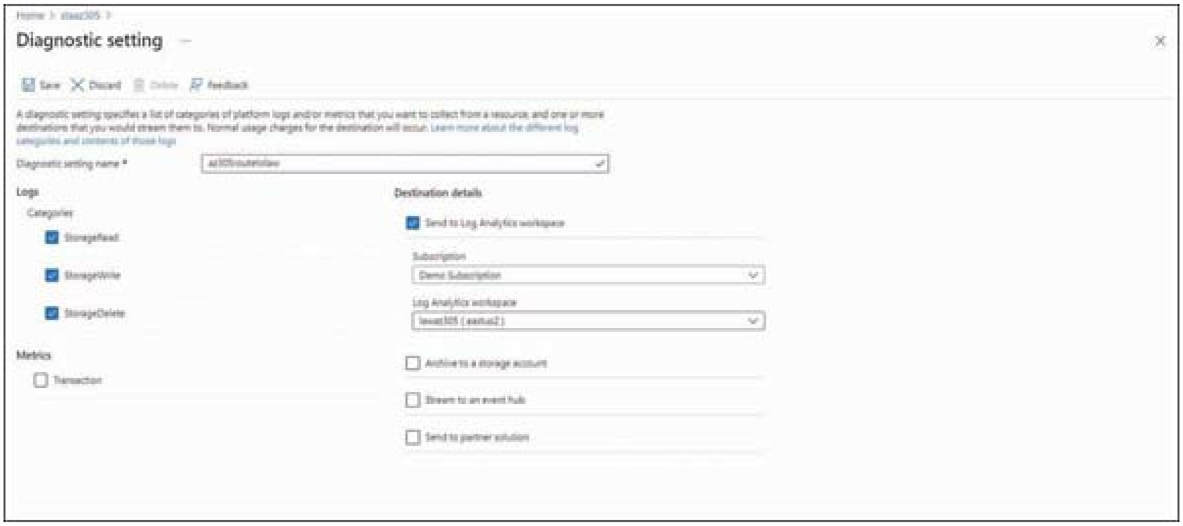
Name	Storage account	Event hub	Log Analytics workspace	Partner solution	Edit setting
No diagnostic settings defined					

+ Add diagnostic setting

Click 'Add Diagnostic setting' above to configure the collection of the following data:

- Transaction

FIGURE 1-1 Diagnostic settings for a storage account



Home > staaz305 >

Diagnostic setting

Save | Discard | Delete | Feedback

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. Learn more about the different log categories and contents of those logs

Diagnostic setting name * az305routetolaw

Logs

Categories

☑ StorageRead

☑ StorageWrite

☑ StorageDelete

Metrics

□ Transaction

Destination details

☑ Send to Log Analytics workspace

Subscription: Demo Subscription

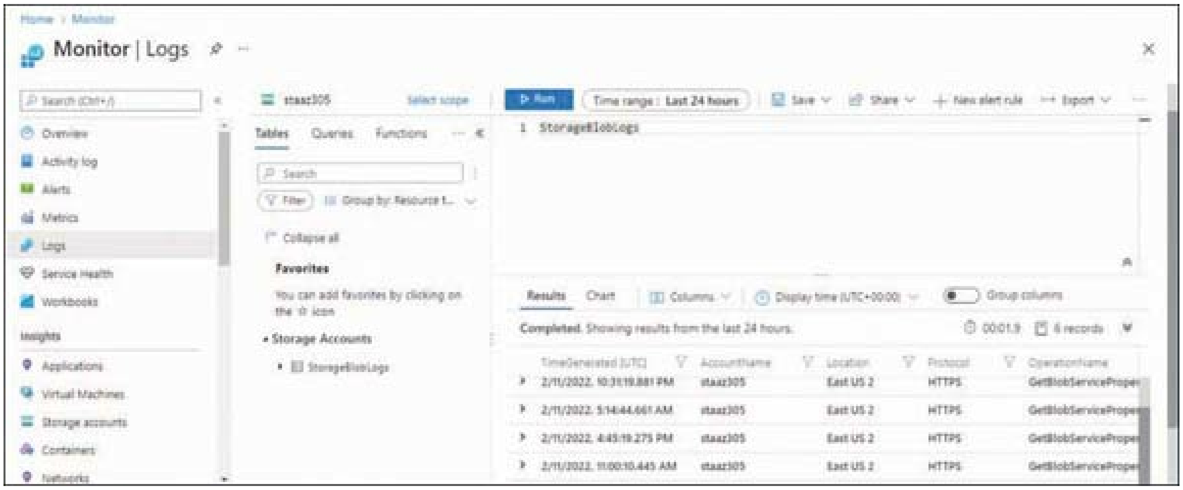
Log Analytics workspace: law305 (eastus2)

□ Archive to a storage account

□ Stream to an event hub

□ Send to partner solution

FIGURE 1-2 Configuring diagnostic settings to send logs to a Log Analytics workspace



Home > Monitor

Monitor | Logs

```
1  StorageBlobLogs
```

You can add favorites by clicking on the ☆ icon

Completed. Showing results from the last 24 hours. 00:01.9 6 records

TimeGenerated [UTC]	AccountName	Location	Protocol	OperationName
2/11/2022, 10:31:19.881 PM	staaz305	East US 2	HTTPS	GetBlobServicePrope
2/11/2022, 5:14:44.661 AM	staaz305	East US 2	HTTPS	GetBlobServicePrope
2/11/2022, 4:45:19.275 PM	staaz305	East US 2	HTTPS	GetBlobServicePrope
2/11/2022, 11:00:10.445 AM	staaz305	East US 2	HTTPS	GetBlobServicePrope

FIGURE 1-3 Query and analyzing storage account logs in a Log Analytics workspace in Azure Monitor

Metrics are another kind of data generated by Azure resources. As with logs, each Azure resource generates different metrics that enable you to monitor its health and performance. You can route these metrics to the same destinations as you can with logs (an Azure Log Analytics workspace, an Azure Storage account, the Event Hub, or a partner solution) by configuring their diagnostic settings. Logs and metrics are discussed further in upcoming sections in this chapter.

More Info **Configure Diagnostic Settings for Azure Resources**

To learn more about configuring diagnostic settings, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/diagnosticsettings?tabs=portal*.

Recommend an appropriate level of logging

Your IT operations team must be able to monitor the health and performance of various IT systems and workloads so they can take appropriate action when needed. To enable your IT operations team to detect, diagnose, and maintain an audit trail of the health and performance of your IT systems as a whole, you should enable system and platform monitoring at various levels.

By design, the Azure platform supports hybrid scenarios—for example, running a workload on-premises, on the edge, in Azure, or in a multicloud environment. You can broadly categorize the sources of logs and metrics based on where the workload is deployed as follows:

- **Azure platform logs and metrics** These are logs and metrics generated by Azure services in Azure, discussed in the preceding section. These logs are generated at various layers or levels:

  - **Azure tenant** AAD logs are generated at the tenant Level.
  - **Azure subscription** Activity logs and metrics as well as service health logs and metrics are generated at the subscription level.
  - **Azure resource** Azure resource logs and metrics are generated at the resource level.

  You can configure diagnostic settings to forward metrics and logs at each of these levels to the following destinations:

  - **Azure Monitor log (Log Analytics workspace)** To query logs, analyze logs, and set up alerts
  - **Azure Storage account** For archiving or long-term retention
  - **Azure Event Hub** To send logs to third-party or custom systems or applications

- **Logs and metrics by workloads** These are logs and metrics generated by workloads, by themselves, running on-premises, in multicloud environments, or in Azure itself. These logs and metrics are generated at two levels:

    - **Guest OS level** Logs and metrics generated at this level must be monitored for compute resources running in Azure, on-premises, or in another cloud. You can install Azure Monitor Agent (AMA) on compute resources. AMA then collects and sends telemetry of logs and metrics from these compute resources to Azure Monitor, where it can be analyzed, similar to Azure platform logs. Deploying AMA enables you to monitor and manage compute resources like Windows or Linux VMs, VMSS, and Azure Arc–enabled servers. (Arc-enabled servers are servers running on-premises, on another cloud, or on the edge, and are connected to Azure through the deployment of appropriate agents.) You can forward logs and metrics to Azure Monitor logs, Azure Monitor metrics, or both.

      AMA data collection rules define what performance counters and logs will be collected from compute resource. You can forward these counters and logs to Azure Monitor logs or Azure Monitor metrics. At the time of this writing, you can forward Windows event logs and Linux Syslog only to Azure Monitor logs, but you can send performance counters to Azure Monitor logs as well as Azure Monitor metrics. When logs and metrics are sent to Azure Monitor logs, performance counters go to the Perf table, Windows event logs go to the Events table, and Linux Syslog go to the Syslog table of the Log Analytics workspace.

      You can also enable the Azure Diagnostics extension for Azure VM. This extension collects logs and metrics at the guest OS level for Azure compute resources like Azure VMs, virtual machine scale sets (VMSS), and Azure Service Fabric. By installing and configuring the Azure Diagnostic extension, you can forward logs and metrics of the guest OS to an Azure Storage account, Azure Monitor metrics, or the Azure Event Hub for long-term storage, further analysis, or integration with third-party systems, respectively. The Azure Diagnostic extension can also send logs to Application Insight logs to enable troubleshooting for the application running on compute resources.

Azure VM Insights provides additional features over and above Azure Monitor. It provides a view of processes running on Windows and Linux VMs as well as a view of external process dependencies for the VM(s). To use Azure VM Insights, you must install Azure VM Dependency agents on Windows and Linux machines. These collect discovered information about processes running on VMs as well as external process dependencies and forward it to a Log Analytics agent, which in turn sends the data to an Azure Log Analytics workspace. Based on the dependency data, VM Insights provides additional capabilities to depict performance, network dependencies, and the health of the VMs in the form of performance and map visualizations.

- **Application-level monitoring** This is done with the help of Azure Application Insights—an application-monitoring service available in Azure Monitor. Application Insights collects logs and metrics for an application running in Azure, on-premises, or in multicloud scenarios. To enable Application Insights, you must install an instrumentation package (SDK) in the application or deploy an Application Insights agent. This enables the collection of operations- and performance-related logs and metrics. Collected logs and metrics then can be stored in the Azure Monitor log or Azure Monitor metric, respectively, where they can be further analyzed. They can also be sent to an Azure Storage account for long-term storage or archiving.

More Info **Configure Data-Collection Rules**

To learn more about configuring data-collection rules for the Azure Monitor Agent, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/azure-monitor/agents/data-collection-rule-azure-monitor-agent?tabs=portal*.

The previous two sections discussed log routing and log levels. Figure 1-4 summarizes these discussions.

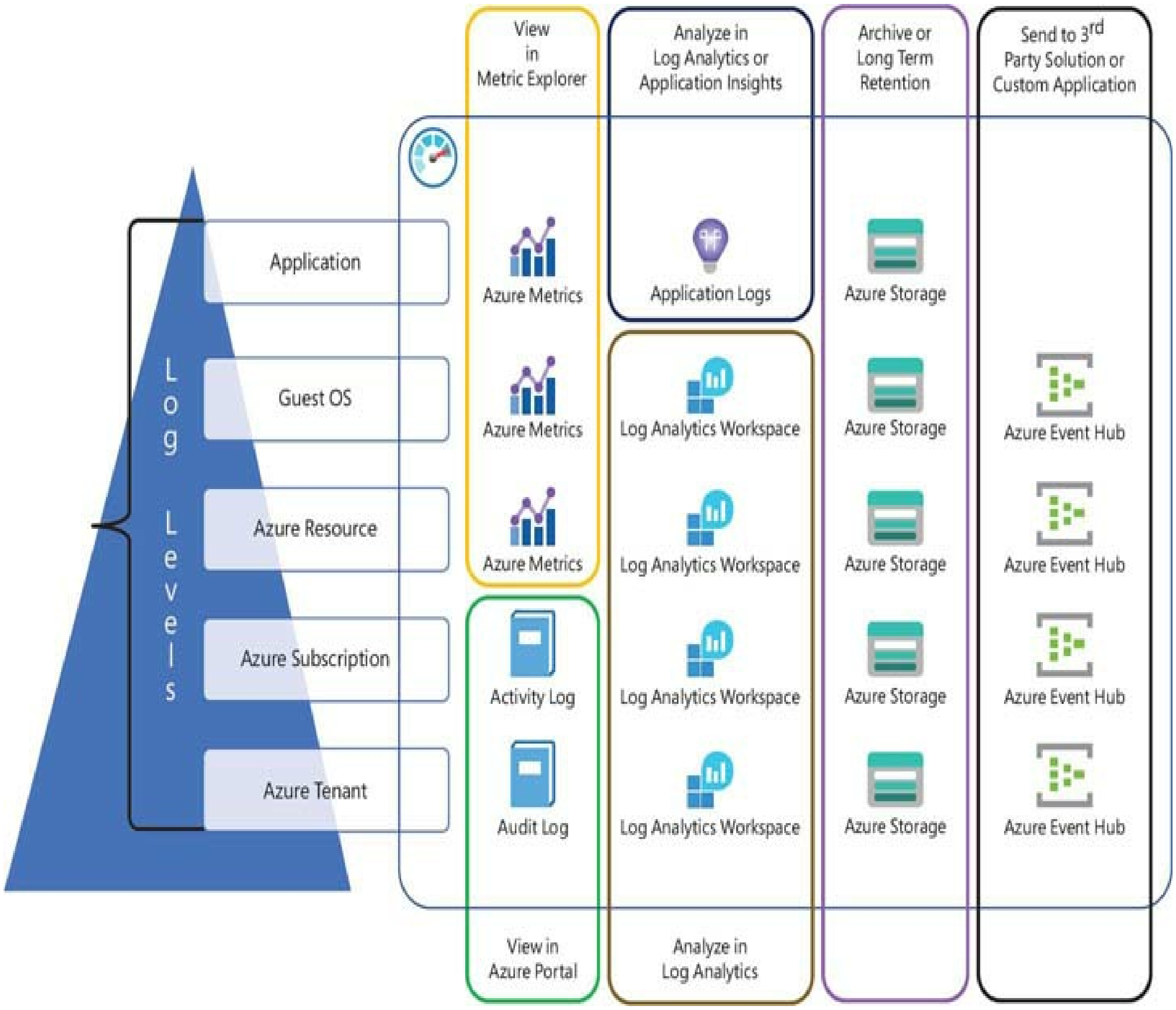


FIGURE 1-4 Summary diagram for log routing, logging levels, and log destination

Recommend monitoring tools for a solution

In addition to the logs discussed in the previous two sections, which also serve as data sources for Azure Monitor, Azure Monitor draws from other data sources. These include the following:

- **Third-party monitoring and customer solutions** These solutions can forward logs to Azure Monitor logs, Azure Monitor metrics, or both. Once logs and metrics are ingested in Azure Monitor, you can use the same set of tools to analyze, visualize, and take action as needed. Clients that can call REST APIs can also forward logs and metrics to a Log Analytics workspace or metric stores, respectively.

- **Microsoft Defender for Cloud** This tool uses a Log Analytics workspace to store security logs ingested from various Azure services and solution components. Again, once they are stored in the Log Analytics workspace, you can further analyze logs in conjunction with log data collected from other sources by Azure Monitor.
- **Microsoft Sentinel** This tool uses a Log Analytics workspace to store data collected from various sources. Microsoft Sentinel provides out-of-the-box connectors for Microsoft solutions to support real-time integrations, such as Microsoft 365 Defender, AAD, and Microsoft Defender for Cloud. In addition to these, there are built-in connectors for non-Microsoft solutions—for example, Palo Alto products like MineMeld and PAN-OS, and Cisco products like ASA. Another way to connect data to Microsoft Sentinel is to use the Common Event Format (CEF) or Syslog, or to send data through the REST API. The same set of tools in Azure Monitor can be used to analyze logs along with other data collected by Azure Monitor.

Azure Monitor

Azure Monitor provides visualizations and tools to monitor and analyze logs and metrics collected from various sources in Log Analytics workspaces and metric stores, respectively. You can configure alerts to send notifications for inferenced events and/or trigger autonomous workflows or actions to mitigate the event.

Some important tools available in Azure Monitor include the following:

- **Activity log** As mentioned, activity logs are stored at the subscription level. Azure Monitor provides the capability to query activity logs based on severity and timespan.
- **Alerts** Setting up alerts in Azure Monitor involves creating the following:
  - **Alert rule(s)** An alert rule articulates the condition for which an alert should be raised. Alert rules require a scope and a condition. The scope defines the scope of the alert. The scope of an alert could be a subscription or specific resources. The condition specifies the signal type, which could be activity log, Azure Monitor metric,

    Azure Monitor log, Resource Health, or Service Health depending on the scope, the resource you select for the alert, and the alert condition logic.

    - **Action group(s)** The action group indicates the action to be taken when an alert condition is met. You can choose to send a notification to select recipients, trigger an automatic action, or both. The rule can specify that a notification be sent to all users with a specific role in Azure Resource Manager, like owner or contributor, within the established scope. Or the rule could specify that the notification be sent to specified users via email, SMS, push notification, or voice message. Autonomous actions can be used to mitigate the alert condition automatically. These actions could be in the form of an Automation runbook, an Azure function, an Azure Logic app, a webhook, or an alert sent to the Event Hub for streaming to a third-party or custom solution. The alert can also be sent to an IT service-management (ITSM) tool.
    - **Alert processing rules** You can use these to suppress the alert in specific scenarios or to specify which action group(s) should be triggered when an alert is tripped at a specific scope.
- **Metrics** You can create visualizations and charts for any metrics you collect and pin them to your dashboard for easy monitoring. You can also create alert rules on metrics.
- **Logs** As with metrics, you can create visualizations and charts for any logs you collect and pin them to your dashboard for easy monitoring. You can also create custom queries to obtain specific insights and to generate alerts.
- **Service Health** This provides visibility on ongoing issues, security advisories, and the health history of Azure services. It also provides visibility into maintenance scheduled for Azure services. You can configure alerts for Azure services within a specific region for health events like service issues, planned maintenance, health advisories, and security advisories. You can also configure actions to mitigate these health events on Azure services, similar to the way you do alerts.
- **Insights** This offers curated visualizations and monitoring tools for many Azure services to provide insights into their health and

performance. The insights differ depending on the Azure service being monitored. Some important insights include the following:

- Application Insights
- VM Insights
- Container Insights
- Network Insights

Application Insights

Application Insights helps developer and DevOps teams investigate application health and performance issues and identify application usage patterns. It includes these useful tools:

- **Application map** This shows various application components and their dependencies. This view is useful for investigating bottlenecks in distributed applications.
- **Smart Detection** This helps detect anomalies in an application. It automatically raises alerts based on any unusual patterns in the telemetry ingested from the application.
- **Live Metrics** This tool helps you to monitor live metric telemetry coming from an application.
- **Availability** You can set up availability tests for an application to monitor its availability. Based on these tests, Application Insights provides metrics on application availability, which you can monitor for specific time intervals.
- **Failures** You can investigate failures within an application—for example, in application operations, dependencies, or server roles. You can also investigate exceptions raised by an application.
- **Performance** You can identify performance issues with regard to application operations, dependencies, and server roles.

To provide insights into an application's usage, the application must send custom telemetry in terms of events and page views to Application Insights. The application must include an instrumentation SDK and send this telemetry from within the code. In this way, Application Insights can provide the following usage insights:

- **Users** View how many users are using each page and feature in an application, identify the countries from which users visit the application, determine which browser they are using, and more.
- **Sessions** Track how many sessions are spent on a particular application page or feature, which sessions originate from which country, what browser is used, and more.
- **Events** See how many times a particular application page or feature is used, from which country, using which browser, and more.
- **Retention** Track how many users return to your application. This can help you understand why users return to your application, as well which aspects of your application seem to cause users to abandon it.
- **Funnel** Gauge users' navigation experience in your application to identify bottlenecks and other user pain points and remove them.
- **User Flows** Obtain a visualization of user navigation in your application across pages and features to analyze user navigation patterns.
- **Cohorts** Use this to define a cohort of users, events, sessions, and operations based on similar characteristics. Cohorts simplify queries in the other usage tools (Users, Sessions, Events, and User Flows).

VM Insights

You can use this to monitor the health and performance of Windows or Linux Azure VMs, Azure VMSS, and Azure Arc–enabled VMs located on-premises or in other cloud environments.

Container Insights

You can use this to monitor the performance and health of containers deployed in the following:

- Azure Kubernetes Service
- Azure Container Instance
- Self-managed Kubernetes clusters (which may be hosted in Azure, on

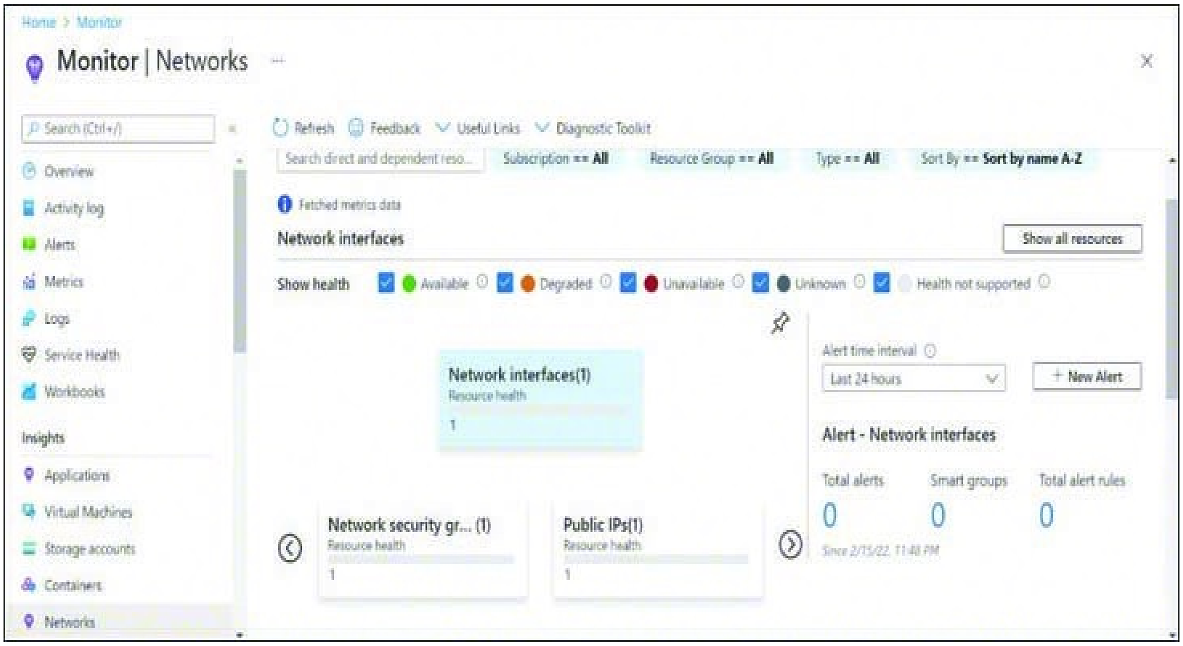
Azure Stack, or on-premises)

- Azure Red Hat OpenShift
- Arc-enabled Kubernetes clusters

Network Insights

This provides visualizations of the health and metrics of deployed network components. It offers three views in three different tabs:

- **Network Health** This tab shows the health of networking components and their dependencies. It also shows any alerts raised for network components. (See Figure 1-5.)



Subscription == All

Resource Group == All

Type == All

Sort By == Sort by name A-Z

Show health ☑ ● Available ☑ ● Degraded ☑ ● Unavailable ☑ ● Unknown ☑ ○ Health not supported

Alert time interval: Last 24 hours

FIGURE 1-5 Network Health tab in Network Insights

- **Connectivity** This tab shows connectivity tests configured in the Network Watcher Connection Monitor as well as any alerts associated with these connectivity tests. (See Figure 1-6.)

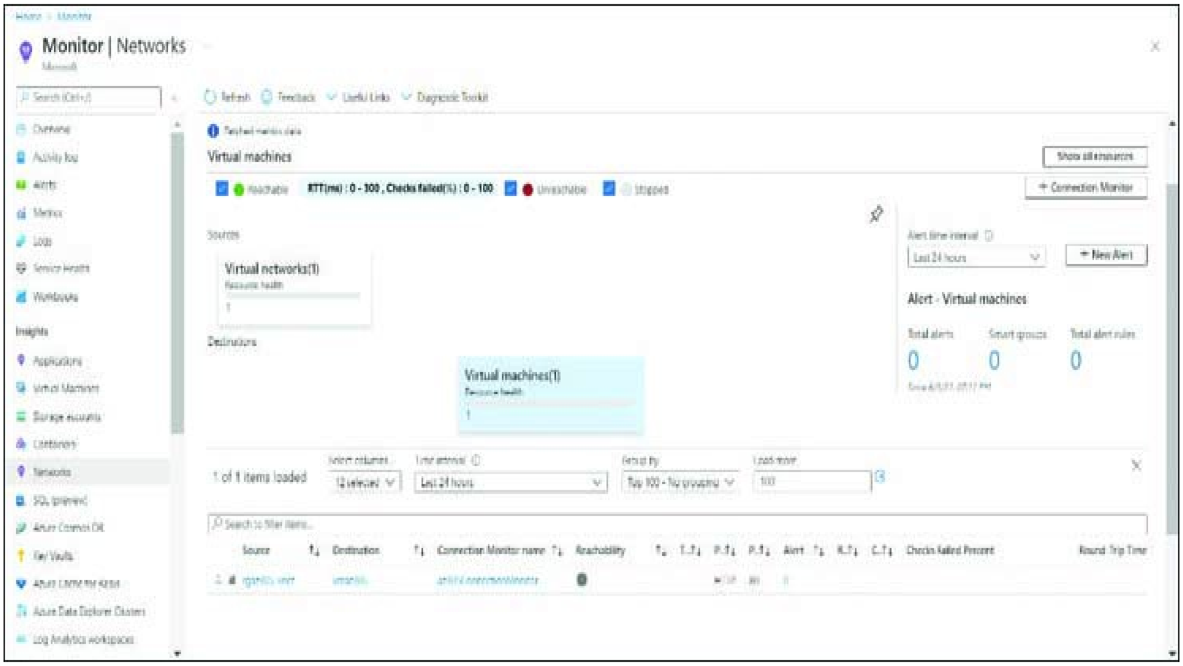


FIGURE 1-6 Connectivity tab in Network Insights

- **Traffic** This tab shows all network security groups (NSGs) that have been configured for NSG flow logs and Traffic Analytics in the selected subscription grouped by which-ever region you select. This tab also shows Traffic Analytics alerts. (See Figure 1-7.)

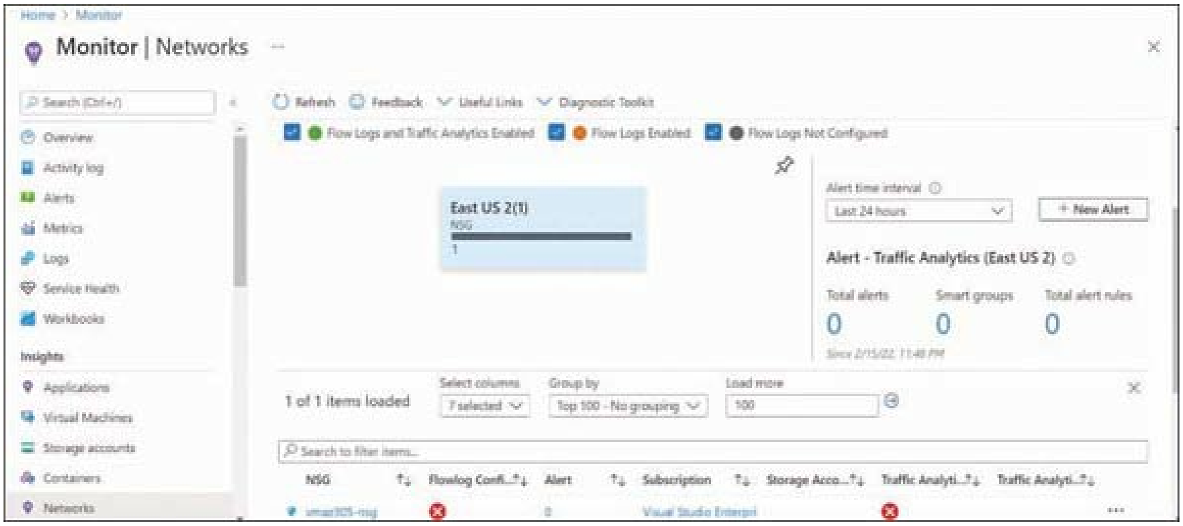


FIGURE 1-7 Traffic tab in Network Insights

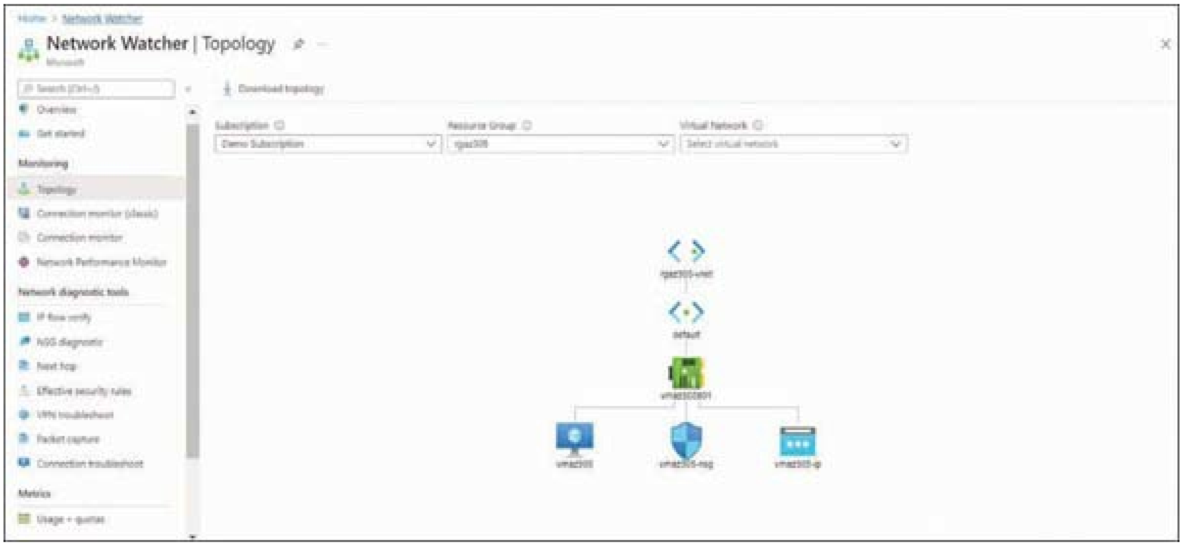
> ***More Info*** **Azure Monitor Insights**
>
> To learn more about insights and curated visualizations, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/azure-monitor/monitor-reference#insights-and-curated-visualizations*.

Azure Network Watcher

So far, you have explored some of the important tools available in Azure Monitor to monitor the health and performance of workloads deployed in Azure, in a hybrid environment, on-premises, or in another cloud. There is one more important tool available in Azure to help monitor your network: Azure Network Watcher.

Azure Network Watcher is a comprehensive set of network-monitoring and diagnostics tools. It provides a number of visualization, monitoring, diagnostics, and alerting capabilities.

- **Topology** This depicts the topology of the network in a resource group or of a specific virtual network. (See Figure 1-8.)



Subscription: Demo Subscription

Resource Group: rgaz305

Virtual Network: Select virtual network

FIGURE 1-8 Topology in Network Watcher

- **Connection monitor** This enables you to create network tests and to monitor network connections. It also enables you to raise alerts for detected network issues, based on network tests you create:
  - **Test group** You can create a group of tests for a specific pair of sources and destinations. To create a test group, you must specify the following, in order:

    Source This can be a VM in Azure or on-premises. The VM you choose must have the Azure network extension installed on it.

    Test configuration You can create multiple test configurations for a test group, which can be used for different protocols and ports.

    Destination This can be a VM in Azure or on-premises or some external endpoint.
  - **Alerts** You can configure alerts for a connection monitor. Creating an alert in this context is similar to creating or attaching an action group, as described in the preceding section.
- **IP Flow Verify** You can use this to test and verify inbound and outbound TCP/UDP connections for a VM for a targeted IP address. The IP address can be local or external.
- **NSG Diagnostics** This tool can help you understand and debug the network's security configuration. It identifies all NSGs that will be evaluated for a given source–destination pair. Based on this, it determines which rule, within each NSG, will be applied, and the final allow/deny status for the flow.
- **Next Hop** This identifies the next hop for traffic from a specified VM to a specific destination IP. This helps in testing scenarios in which you want the traffic from a VM to hop to a specific appliance before it goes to any destination.
- **VPN Troubleshoot** This diagnoses issues with virtual network gateway and VPN connections. Be aware that once it begins, it takes some time to detect and report the results.
- **Packet Capture** This captures packets for a VM. You can configure

the packet capture (.cap) file to be stored in Azure Blob storage, on the VM’s file system, or both.

- **NSG flow logs** You can configure NSG flow logs to capture flow logs for an NSG. An Azure Storage account is required to store Network flow logs.
- **Traffic Analytics** This provides analytics and visualizations for NSG flow logs and other Azure resource’s data. It helps identify traffic hotspots, which in turn can help you to identify areas for optimization. It also provides a drill-through geo-map, which you can use to gain insights into the network traffic across geographies.

Microsoft Defender for Cloud

Microsoft Defender for Cloud is an Azure-native security posture–management and threat-protection tool. As an Azure-native solution, Microsoft Defender for Cloud can be auto provisioned and easily enabled for various Azure services without any special deployment. It helps strengthen the security posture of cloud deployments by monitoring for security and compliance issues and by providing security-hardening tools for Azure resources.

Microsoft Defender for Cloud:

- Continuously assesses the security posture of connected Azure resources and services and provides a security score for your Azure security posture. The higher the score, the better the security posture. This helps with hardening connected resources by monitoring them and comparing them to an Azure security benchmark.
- Provides recommendations to fix identified vulnerabilities, and in many cases provides a Fix button, which you can click to fix the vulnerability automatically.
- Detects threats and raises alerts. Alerts are displayed in the Azure Portal, and can be sent via email to designated recipients, forwarded to a SIEM or SOAR solution (such as Microsoft Sentinel), and/or forwarded to an ITSM tool.

Cost Management

The Cost Management tool enables you to monitor the consumption and cost of Azure resources. It includes the following features:

- **Cost Analysis** With this tool, you can analyze costs at various levels, such as management group level, subscription level, resource group level, or resource level.
- **Budgets** You can set this according to monthly usage, and you can configure alerts for usage that exceeds the threshold cost you specify.
- **Advisor Recommendation** You can use this to optimize the cost of your Azure subscription. It offers recommendations—such as resizing or shutting down underutilized VMs and using reserved VM instances rather than paying as you go—to reduce your costs.
- **Invoices** You access these in the Billing section of the Cost Management tool.
- **Payment** You configure payment methods in the Billing section of the Cost Management tool.

Azure Advisor

Azure Advisor is a single-stop shop to keep watch over the following:

- Cost management
- Security
- Reliability
- Operational excellence
- Performance

Azure Advisor provides an advisor score. (See Figure 1-9.) A higher score indicates that your Azure Cloud deployment follows the best practices of the Azure WAF. Azure Advisor also provides recommendations to improve each of the WAF pillars for your Azure deployment.

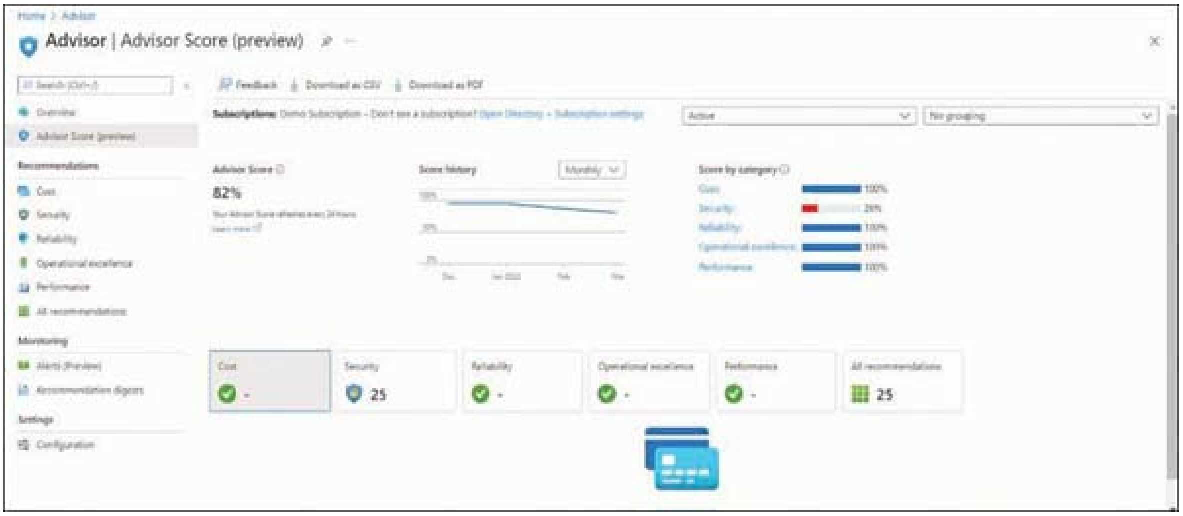


FIGURE 1-9 Azure Advisor score

Skill 1.2: Design authentication and authorization solutions

These days, many organizations are embarking on a digital-transformation journey to make themselves more agile and able to quickly and efficiently adapt to disruptions. At the same time, more and more employees, customers, vendors, and partners want to be able to access resources and information from anywhere and on any device. Not surprisingly, many organizations are moving their IT assets to the cloud to allow for the agility they require.

Although moving to the cloud, and enabling stakeholders to access resources and information as and when needed from any device, is good, it also poses a significant security threat to an organization's IT assets and data. This is because these assets and information no longer sit behind the corporate firewall.

To handle this, organizations are embracing proactive security with zero trust. There are three main principles of zero trust:

- **Verify explicitly** Every attempt to access resources and information must be authenticated and authorized based on all the information available within the access request. In addition to user identity, information such as location, device health, resource or service being accessed, and classification of data being accessed must be used for authentication and authorization purposes.

- **Use least privileged access** Limit user access by ensuring that access is provided just in time, and that just enough access is given to complete the job at hand. This means applying policies that adapt based on a risk assessment of the request. Also be sure to protect and secure data. Just remember: The goal is to ensure security without affecting user productivity.
- **Assume breach** Minimize exposure and implement isolated and segmented access to information and IT assets. Also implement end-to-end encryption and use threat intelligence to obtain visibility, detect threats, and improve protection.

This section covers how to:

- Recommend a solution for securing resources with role-based access control
- Recommend an identity management solution
- Recommend a solution for securing identities

Recommend a solution for securing resources with role-based access control

Attaining this skill requires an understanding of designing role-based access control (RBAC) for Azure resources such as VMs, Key Vault, Azure Web App, Azure Storage, and so on.

Azure RBAC

Azure RBAC is an authorization fabric built over Azure Resource Manager. It helps provide fine-grained access privileges for accessing Azure resources. These fine-grained privileges enable you to control things such as who can create VMs, who can manage storage accounts, who can perform data-plane operations within a Key Vault, and more.

Let’s understand some key concepts and terminology of Azure RBAC:

- **Security principal** This can be a user, a group, a service principal, or a managed identity. (Service principal and managed identities will be discussed shortly.)
- **Role or role definition** A role is basically a set of permissions—like read, write, and delete—for a specific resource. A role can also be a set or collection of permissions to work with the data plane of a resource like a Key Vault, an Azure Storage account, and so on. A role definition can be broad—as with Owner, Contributor, or Reader roles—or it can be granular, as with roles such as Storage Blob Data Reader, Key Vault Administrator, and so on. Here is a sample of the Storage Blob Data Reader role:

Click here to view code image

```
{
   "id": "/providers/Microsoft.Authorization/roleDefinitions/2a2b9908-6ea1-4ae2-
8e65-a410df84e7d1",
         "properties": {
                "roleName": "Storage Blob Data Reader",
                "description": "Allows for read access to Azure Storage blob
containers and data",
                "assignableScopes": [
                       "/"
                ],
                "permissions": [
                       {
                             "actions": [

"Microsoft.Storage/storageAccounts/blobServices/containers/read",

"Microsoft.Storage/storageAccounts/blobServices/generateUserDelegationKey/action
"
                              ],
                              "notActions": [],
                               "dataActions": [

"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/read"
                                ],
                                "notDataActions": []
                        }
                   ]
             }
    }
```

There are many built-in roles available out of the box for each Azure resource. If you have a specialized need that cannot be met by any of the available built-in roles, you can create a custom role.

More Info **Azure Built-in Roles and Custom Roles**

For more information about Azure built-in roles, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/role-based-access-control/built-in-roles*. For more information about custom roles, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/role-based-access-control/custom-roles*.

- **Scope** This can be a management group, a subscription, a resource group, or a specific Azure resource, like a VM, an Azure Storage account, a managed database, and so on. (The organization of Azure subscriptions and resource groups within the management group is discussed in detail in the upcoming “Design governance” section.)
- **Role assignment** Simply put, role assignment involves configuring a relationship between a security principal and a role definition. A single security principal can have one or more role assignments, with each role assigned on a particular scope. For example, a user can be assigned a Contributor role for one resource group, and the same user can be assigned an Owner role for another resource group.
- **Groups** Although you can assign a role to each security principal, it is a good practice to assemble security principals who need the same or a similar set of permissions into a group. Using groups makes it easier to manage the access assigned to security principals and is more secure too. Because the group itself is a kind of security principal, it is possible to nest groups inside other groups, creating a hierarchy of groups. You can assign roles at a group level within the hierarchy; this role assignment then applies down the hierarchy, with “child” groups inheriting permissions from their “parent” group.
- **Deny assignments** These are similar to role assignments, but whereas a role assignment allows permissions to a group or security principal, a deny assignment denies permissions to a group or security principal. Deny assignments are given priority over role assignments. This means if a user has a deny assignment for an action and role assignment for

the same action, the user will not be allowed to carry out that action.

More Info **How Azure Rbac Evaluates User Access Over a Resource**

To find out more about how Azure RBAC evaluates user access for a resource, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/role-based-access-control/overview#how-azure-rbac-determines-if-a-user-has-access-to-a-resource*.

Azure Active Directory (AAD) roles

In addition to Azure RBAC roles, there is another set of roles, called AAD roles. These roles allow for administration activities in the AAD tenant, such as creating users, managing subscriptions in the tenant, and changing user passwords. Table 1-1 lists a few important AAD roles.

TABLE 1-1 Important AAD roles

AAD role	Permissions	Notes
Global administrator	Manage access to all administrative features in AAD by assigning administrative roles to others. Manage administrative access for services that federate to AAD, like M365, Azure DevOps, Power BI, and so on. Reset passwords for all users and administrators.	This is the default AAD role assigned to the user who signs up for the AAD tenant.
User administrator	Create and manage all aspects of users and groups. Manage support tickets, monitor service health. Change passwords for all non-administrator users and specific administrators (help-desk	

	administrators and other user administrators).	
Billing administrator	Make purchases. Manage subscriptions in the AAD tenant. Manage support tickets and monitor service health.	

More Info **Azure AD Built-in Roles**

For more information about built-in AAD roles, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/active-directory/roles/permissions-reference*.

Recommend an identity management solution

One important component of security is the identity and access management (IDAM) solution. Identities of users, services, applications, and devices to be authenticated and authorized must be managed in an IDAM solution.

In addition to authentication, the IDAM solution should also support the protection of identities, identity governance, logging, auditing, and reporting. Hybrid scenarios may require the IDAM solution in the cloud to be in sync with on-premises IDAM solution, both working seamlessly together to provide a frictionless experience for users and administrators.

An IDAM solution should enable B2B collaboration by allowing partners, vendors, suppliers, and other collaborators to use their own identity to access your Microsoft or other enterprise applications. It should also support B2C scenarios in which your organization wants to publish applications that can be accessed by consumers and customers bringing their own identity.

Azure Active Directory (AAD)

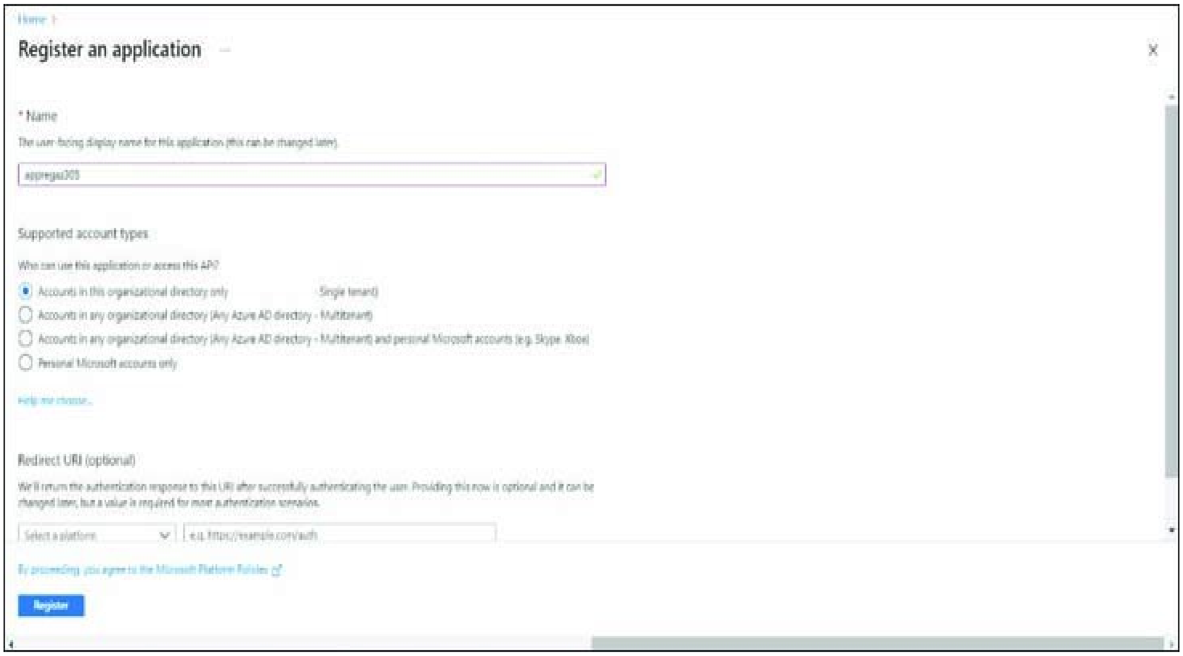
Azure Active Directory (AAD) is a comprehensive native IDAM solution in Azure. AAD not only helps secure access to line of business (LOB) applications, services, and SaaS applications in Azure and M365, but can also be extended to secure LOB applications, services, and SaaS applications deployed on-premises, on the edge, or on any other cloud.

Note You learned about users, groups, and AAD roles earlier. In this section, you will look into AAD features related to identity management.

Suppose you need to create accounts or identities in AAD to enable applications and services to assume those identities to access an Azure resource or application protected by AAD in an Azure or non-Azure environment. The account or identity created for this purpose is called the *service principal*.

This section alludes to the concept of application registration to explain the concept of the service principal; you will look further into integrating the application in AAD in later sections.

Application registration is a way by which an application can offload IDAM functions to AAD. Registering an application creates a globally unique identity for the application in the AAD tenant in which it is registered. This globally unique identity is called an *application object*. To register an application, you must specify its name, account type, and the URL where the user will be directed on successful login. (See Figure 1-10.)



Register an application

* Name

The user-facing display name for this application (this can be changed later).

appregaz305

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only Single tenant)

○ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

○ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

○ Personal Microsoft accounts only

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Select a platform ⌄ | e.g. https://example.com/auth

FIGURE 1-10 Application registration

When you register an application through the Azure Portal, an application’s *service principal object* is also created in the same tenant. The application object mainly has configuration settings that relate to which token will be issued by the AAD service to the consumer requesting access to this application, how it will be issued, and which APIs or services the application itself can access with user or admin consent.

To access any resource protected by AAD, a service principal object is required. The service principal object can be a user principal for a user or a service principal for an application or service. User principal and service principal are types of security principals. These security principals are core to authenticating users, applications, and services, and to authorizing their access to resources.

Service principals can be of the following types:

- **Application** As mentioned, an application service principal object is created along with the application object when app registration is done through the Azure Portal. This application service principal object aids in configuring who can access the application, what the application can do in the AAD tenant, and what Azure resource the application can access.

- **Managed identities** Azure resources can be assigned managed identities; this results in the creation of a service principal representing those services in AAD. You can enable managed identities for many Azure services using the Azure Portal. You can then leverage this managed identity to give access permission for other Azure services. There are two types of managed identities:
    - **System-assigned managed identity** When you enable a system-assigned managed identity for an Azure service, it follows the lifecycle of the resource itself. If the resource is removed, the system-assigned managed identity is removed too. It can be used only by the Azure service for which it has been enabled.
    - **User-assigned managed identity** The user-assigned managed identity is the Azure resource itself. This must be created similarly to any other Azure resource. You can assign a single user-assigned managed identity to multiple Azure services. A user-assigned managed identity is not automatically deleted when the resources associated with it are deleted; you must remove it explicitly.

> ***Note*** Skill 1.4 revisits the topic of application registration and service principal in the context of application integration with AAD.

Azure external identities

Being a comprehensive IDAM system, Azure AD External Identities provides for scenarios in which an external user can use their own identity. This might be their organizational identity or a social identity such as the one they use on Google or Facebook. This scenario could arise in a situation in which an organization wants to enable external users to securely access their organizational resources. These users could be from a partner, supplier, or vendor organization (a scenario called *B2B collaboration*). In this case, external B2B users are managed in the same AAD tenant as your organization’s employees. Or they could be external consumers or customers who need to be able to securely access your organization’s published applications (a B2C scenario). B2C user identity and access management is done in a separate directory, Azure AD B2C. Table 1-2 contains some important comparisons between these two types of external identities.

TABLE 1-2 B2B collaboration versus AAD B2C

	B2B collaboration	**AAD B2C**
Scenario	Provides access to external users while allowing them to bring their own identities. Access can be given to Microsoft applications or your applications (SaaS apps, custom-developed apps, and so on), which are protected by your organization’s AAD tenant.	Allows external consumers and customers to access your published application, which could be a SaaS application or a custom developed application. This application cannot be a Microsoft 365 application like Teams, SharePoint, Office, and so on.
Type of Users	Business partners from various organizations, like suppliers, partners, or vendors. These organizations may or may not have AAD.	End customers or consumers of products and services.
User Directory and Management	B2B users are onboarded or invited as guest users and appear as guest users in the organization’s AAD in which the organization’s employee identities are managed. These external user identities can be managed similarly to employee identities.	These users are managed in an AAD B2C directory that is separate from the organization’s AAD and any other partner’s AAD.
Identity Providers Supported	Work accounts, school accounts, email addresses, identities from SAML or WS-Fed based identity providers, and social identity providers like Google and Facebook.	Local application accounts (any email address, user name, or phone number), AAD, various supported social identities and consumer identities.
Single Sign-	Supported for all applications that are connected to AAD. These could be Microsoft 365 applications, applications running on-premises, or other SaaS	Supported only for the application registered in AAD B2C. This application cannot be a Microsoft 365 application.

On (SSO) Support	applications.	

AAD is a cloud-native identity solution. But in real-world implementations, large enterprises will likely continue to run at least some of their workloads on-premises—for example, for compliance purposes, because they still rely on some legacy systems, and so on. Such a hybrid environment calls for hybrid identity management. In this scenario, users should be able to use the same identity to access workloads in the cloud or on-premises.

Azure AD Connect

Azure AD Connect helps organizations sync their on-premises Active Directory to AAD. It requires the deployment of an Azure AD Connect application in an on-premises environment. This enables users to employ the same identity and password to access applications and workloads on-premises or in Azure Cloud.

Depending on the configured Azure AD Connect synchronization options for sign-in, authentication can take place in the cloud or on-premises. The three available authentication methods are as follows:

- **Password hash synchronization (PHS)** When this sign-in option is configured for Azure AD Connect synchronization, a hash of the password is synchronized in AAD. As a result, AAD can authenticate users in the cloud itself without any dependencies. Users can use the same password as the on-premises password.
- **Pass-through authentication (PTA)** This sign-in option also allows users to use the same password to authenticate and access applications or workloads on-premises or in the cloud. Although PTA is similar to PHS, there are fundamental differences. For example, passwords are not synchronized in the cloud. Rather, a user’s password is validated against the on-premises Active Directory. This requires a lightweight agent to be deployed on-premises; this agent performs the pass-through authentication. An important use case for this sign-in option is to

enable organizations to apply the on-premises Active Directory policies on passwords.

- **Federation** This is a mechanism by which trust is set up between authentication systems. With federation, the authentication process is completely handled by another trusted authentication system, such as Active Directory Federation Service (ADFS), which might be deployed in the on-premises environment. Authentication requests received by AAD are handed over to the federated authentication system to validate user identity and passwords. If ADFS is on-premises, then authentication happens on-premises.

AAD provides a feature to configure Seamless Single Sign-On (SSO). This allows users to access cloud-based applications on corporate devices connected to the corporate network without providing their password for every login. You can combine Seamless SSO with PHS and PTA where users are authenticated in the cloud.

Azure AD Connect Health

Synchronizing AAD with on-premises Active Directory provides users with seamless authentication across the cloud and on-premises. In addition, you can monitor the on-premises identity infrastructure using Azure AD Connect Health. This requires the deployment of an agent in related servers.

More Info **Azure AD Connect Cloud Sync**

Microsoft has a new offering to sync on-premises AD with AAD: Azure AD Connect Cloud Sync. This service is also a solution for hybrid identity. In addition to being used independently for synchronization, Azure AD Connect Cloud Sync can be used in conjunction with Azure AD Connect. For more information, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/active-directory/cloud-sync/what-is-cloud-sync*.

Multi-factor authentication

Multi-factor authentication (MFA) is a way to authenticate users by applying more than one challenge to ascertain their identity. These challenges can be based on one or more of the following:

- Physical possession of an object, such as a phone, key, or access card
- Knowledge of something, such as a password or PIN
- Biometrics, such as the user's fingerprint, voice, or iris of the eye
- The location from which the user is trying to authenticate

In AAD, you can apply MFA using the per-user MFA configuration. However, a better way to configure MFA is by using conditional access policies in addition to password-based authentication. The methods by which to do MFA are as follows:

- Microsoft Authenticator
- Windows Hello for Business
- FIDO2 security key
- OATH hardware token
- OATH software token
- SMS
- Voice call

AAD also has a preconfigured set of conditional access policies called *security defaults* that can serve as a starting point for an organization to improve its security. These policies are as follows:

- Requiring all users to register for AAD MFA
- Requiring administrators to perform MFA
- Blocking legacy authentication protocols
- Requiring users to perform MFA when necessary, based on identified risks
- Protecting privileged activities like access to the Azure Portal

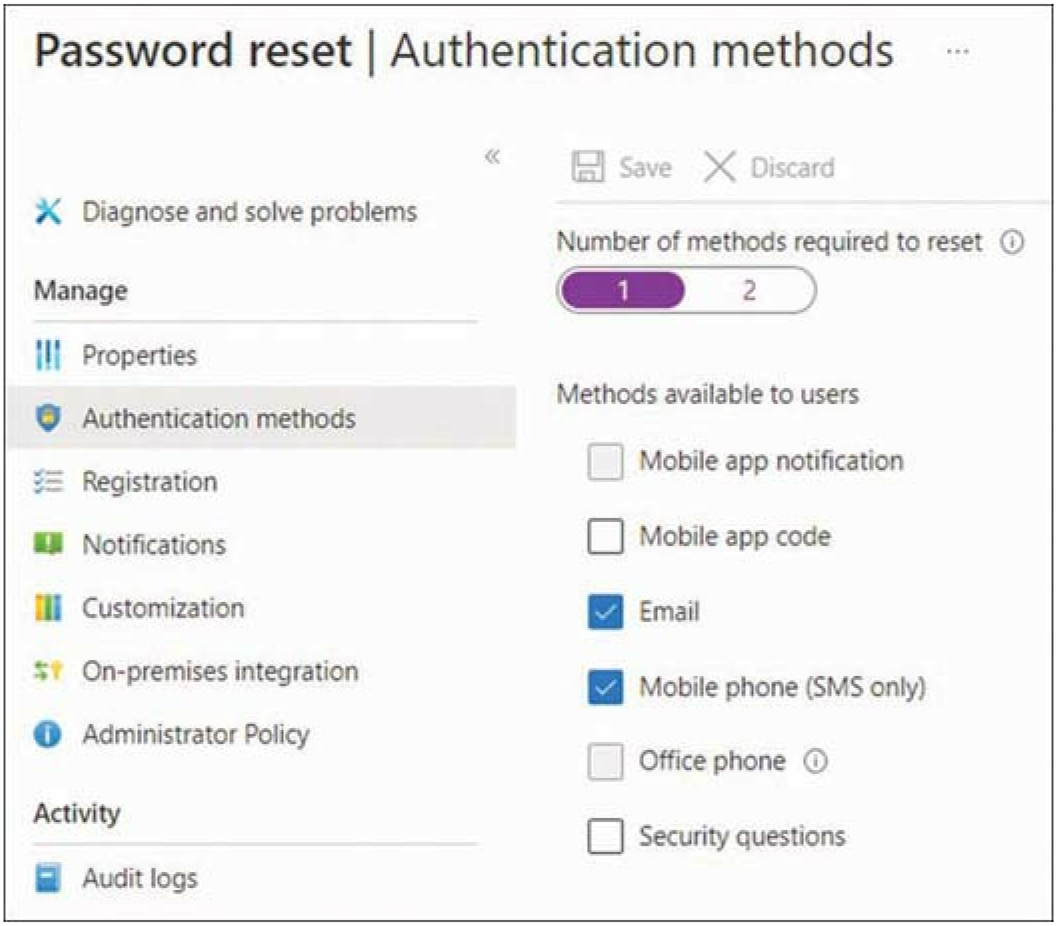
If, however, an organization wants to configure its own set of policies, it can disable these preconfigured policies. You will learn more about conditional access policies later in this chapter, in the section "Identity

Protection.”

Password reset

Another important aspect of AAD is allowing users to reset their password themselves in a self-service fashion. This greatly reduces the burden on the IT operations team.

You can set up self-service password reset (SSPR) for users only if MFA is configured for them. You can also establish one or two additional methods for identifying users before they can reset their password. Figure 1-11 depicts the options available to verify a user for a password reset.



Password reset | Authentication methods

Number of methods required to reset

[✓ 1] 2

Methods available to users

☐ Mobile app notification

☐ Mobile app code

☑ Email

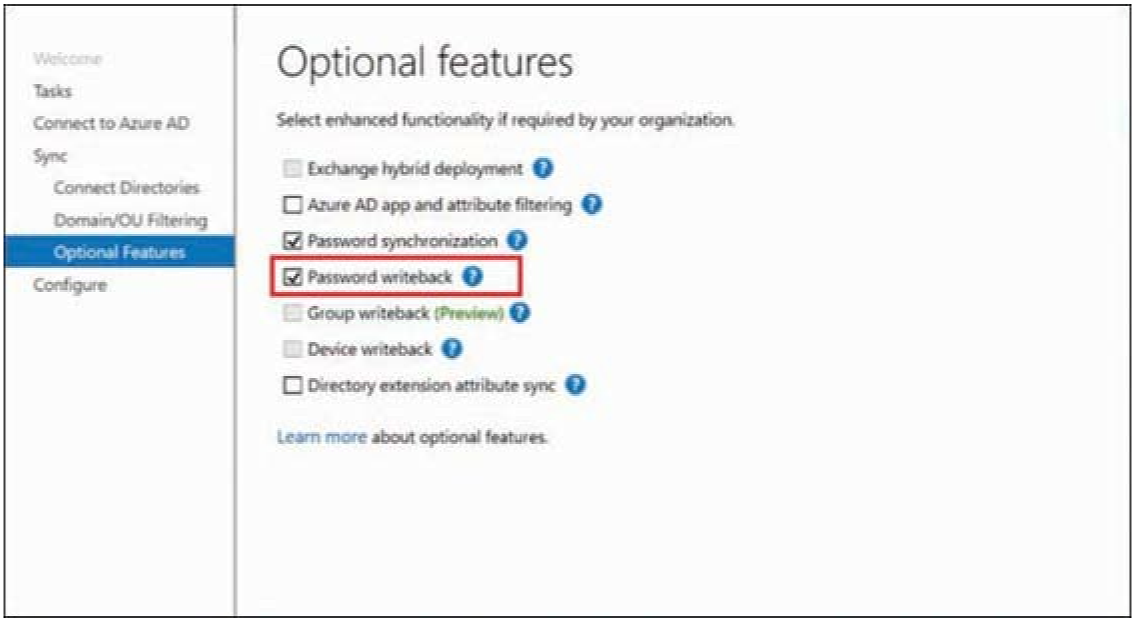
☑ Mobile phone (SMS only)

☐ Office phone

☐ Security questions

FIGURE 1-11 Self-service password authentication configuration

In addition, when you configure Azure AD Connect to synchronize the on-premises Active Directory to AAD, there is a Password Writeback checkbox. (See Figure 1-12.) By selecting this option in Azure AD Connect and then enabling the Password Writeback option in Azure AD SSPR (see Figure 1-13), you can ensure that any password reset or change done in AAD will write back to the on-premises Active Directory as well, resulting in the on-premises Active Directory having the same user password as the AAD in the cloud.



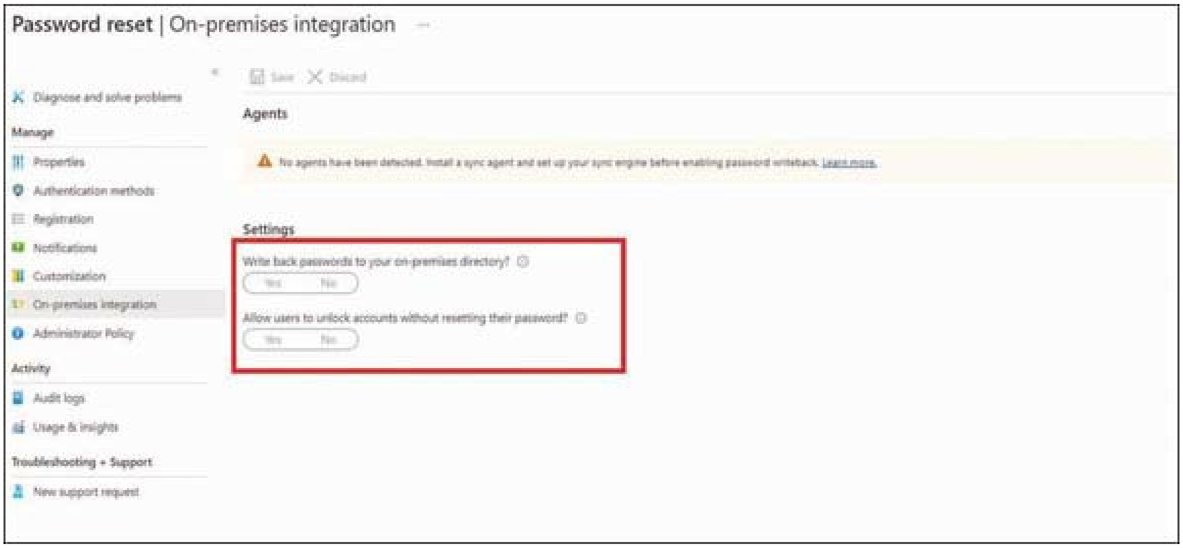
Optional features

Select enhanced functionality if required by your organization.

□ Exchange hybrid deployment
□ Azure AD app and attribute filtering
☑ Password synchronization
☑ Password writeback
□ Group writeback (Preview)
□ Device writeback
□ Directory extension attribute sync

Learn more about optional features.

FIGURE 1-12 Enable the Password Writeback option in Azure AD Connect.



Password reset | On-premises integration

Agents

No agents have been detected. Install a sync agent and set up your sync engine before enabling password writeback. Learn more.

Settings

Write back passwords to your on-premises directory?

Yes No

Allow users to unlock accounts without resetting their password?

Yes No

FIGURE 1-13 Self-service password reset password writeback options

Recommend a solution for securing identities

Identity verification is an important step in securing IT assets, including infrastructure and applications. It is also important to secure these identities to prevent them from being used with malicious intent to breach IT security. AAD provides features to secure these identities, including user identities and administrator identities. Administrator identities require a higher level of protection.

Identity Protection

Identity Protection is an important feature in AAD. Before delving into Identity Protection, however, it helps to understand what types of risks must be mitigated for. These risks can be categorized into two groups, and can be detected in real-time or offline by using threat intelligence SIEM tools like Microsoft Sentinel:

- **User risks** These relate to user credentials that may have been compromised and/or to user behavior patterns that could be suspicious —that is, similar to identified malicious behaviors and not specifically related to sign-in.
- **Sign-in risks** These include unusual sign-in requests, which might not be authorized by a user. An example could be a sign-in request coming from an IP address in a geographical location that is extremely far away from another location where the user recently signed in.

> ***Note*** User risks can be detected in offline assessment, whereas sign-in risks can be detected in real-time as well as in offline mode.

AAD Identity Protection provides built-in user and sign-in risk policies that define what action must be taken when a user or sign-in risk is detected. For example, you might choose to block access, or to allow it but require the user to change the password. Identity Protection also provides reports, which you can view and analyze in Azure Portal. These reports are related to risky users, risky sign-ins, and risk detection. Finally, you can export data from Identity Protection to Microsoft Sentinel for SIEM- and SOAR-related operations.

As you've learned, MFA is one of the most effective ways to protect identities and considerably reduces the risk of the malicious use of an identity. You also learned that MFA can be implemented using conditional access policies. However, you can also use conditional access policies to mitigate detected user or sign-in risks.

As the name suggests, conditional access policies can be configured with conditions to grant or deny access depending on whether the conditions are true. For example, you might use a conditional access policy to ensure that users are granted access to a cloud application only if they log in from a corporate device. You could also use one to apply MFA when a user tries to access an application from outside the corporate network. Or you could use one to block access when a user tries to access a particular application from a particular location.

Configuring conditional access policies includes specifying the following settings:

- **User or Workload Identities** Use this setting to select specific users, groups, or work-load identities (service principals) for which the policy will be triggered. You can also use it to configure exclusion options—for example, to apply the policy to all groups except for a particular one, or in a break-glass scenario.
- **Cloud App or Actions** Use this setting to configure a cloud application(s) or user action for which the policy should be evaluated. Cloud applications can be applications registered in AAD or M365 applications like Project Online, Microsoft Teams, and so on. User actions could be registering security information or registering or joining devices. You can also configure this setting to exclude cloud applications for which the policy should be exempted.
- **Condition** These are the conditions that are evaluated for the policy. If these conditions are met, the policy is enforced. You can configure conditions for various signals while the user is trying to access a particular cloud application or to perform a particular action, as noted

above. These signals are as follows:

- **User Risk** Specify the level of user risk—High, Medium, or Low—that will cause the condition to evaluate as true.
- **Sign-In Risk** Specify the level of sign-in risk—High, Medium, or Low—that will cause the condition to evaluate as true.
- **Device platform** Specify the device platform for which the condition will evaluate as true. Options include iOS, Android, Windows, macOS, Windows Phone, and Linux.
- **Location** Specify a named location for IP ranges or country IPs for which the condition will evaluate as true.
- **Client App** Specify the use of which client apps—including browsers, native mobile apps, or desktop clients—for which the condition will evaluate as true.
- **Grant** Specify the action that will be taken when the condition evaluates as true for any selected user, workload, app, or user action. The action could be to block access or to allow access with additional verifications or requirements.

Identity Governance

Another aspect of Identity Protection is identity governance. AAD provides various identity governance features, with two that are particularly important:

- **Access reviews** You can use these to review and manage user access to enterprise applications. You create and assign access reviews to relevant users, like managers, to review access for other users, such as their subordinates. You can also set up access reviews to be performed on a periodic basis.
- **Privileged identity management (PIM)** PIM can control privileged access permissions, like administrative permissions. It allows just-in-time privilege access for users, which can be configured to be revoked. PIM allows you to do the following:
  - **Assign** You can configure administrative access assignments for groups or users. These assignments can be activated right away. Alternatively, they can be marked for eligibility, such that eligible

users can request privileged access as needed. You can also mark assignments as permanent or as valid only for a specified duration.

- **Activate** Users who are eligible for administrative access can activate that access as needed for the period specified within the assignment.
- **Approve** You can configure requests for activation of privileged access to require approval. All activation requests requiring approval can be approved by the user who created the assignment.
- **Audit** A history of all assignments and activations is available for auditing and traceability purposes.

EXAM TIP

Access reviews and PIM are available only in AAD Premium P2. Conditional access policies are available in AAD Premium P1 and P2. To see what other features are available in which AAD plans, see *https://www.microsoft.com/en-in/security/business/identity-access/azure-active-directory-pricing*.

Skill 1.3: Design governance

Governance management is an important aspect of Azure management. It mainly deals with ensuring that an organization’s Azure deployment complies with required regulatory and organizational policies and standards. However, another aspect of governance management is cost management. This means continuously tracking, reporting, and keeping in check costs and expenditures in Azure and other cloud providers. Governance management mainly involves two services in Azure: Azure Policy and Cost Management + Billing. This skill deals with Azure Policy.

This skill covers the following topics:

- Recommend an organizational and hierarchical structure for Azure resources
- Recommend a solution for enforcing and auditing compliance

Recommend an organizational and hierarchical structure for Azure resources

Azure provides a way to organize your subscriptions and resources in a hierarchical structure. Each entity at a particular hierarchical level is defined as a scope. You can apply and monitor access, governance, and cost budgetary controls at each scope.

> ***Note*** We briefly touched on the topic of scope in an earlier section, in the context of role assignments.

The CAF suggests enterprise-scale landing zones, which provide guidance, recommendations, and templates to organize subscriptions and resources in the organization’s AAD tenant.

Figure 1-14 provides a representation of hierarchical levels for organizing subscriptions and resources in Azure. These include the following:

- **Management group** This is a container for your subscriptions and allows you to organize them into logical groups such as departments, functions, and environments within an organization. Governance conditions applied at the management group level are inherited by all the subscriptions within that particular management group. This helps in applying governance policies at the logical group level—for example, at the department, function, or environment level.

  As shown in Figure 1-14, an AAD tenant provides a root management group. Any access or governance policy applied at the root management group level applies to all the subscriptions and resources in an organization. It is possible to create subscriptions directly under the root management group, which might be a good choice if your organization has only a few subscriptions. However, in the real world, organizations often have many subscriptions, and each one must comply with different regulatory and organizational standards. Therefore, grouping these in separate management groups under the root management level is a good practice.

  When designing a hierarchy, there are some important points to consider:

  - A single AAD can support a maximum of 10,000 management groups.
  - The management group hierarchy tree can be up to six levels deep, not including the root management group level or the subscription level.
  - Although each management group can have many children, it can have only one parent management group.
  - Within a single AAD, there is an only one hierarchy tree. All subscriptions and management groups roll up to a single root management group in a directory.

- **Subscription** A subscription is a unit of billing, scale, and management. Various limits on Azure resources and services are placed at the subscription level. In addition to being a hierarchy level, a subscription is a scope to which you can apply access and governance policies. Each subscription can have only one management group parent. Governance policies and access assignments at the subscription level are inherited by child resource groups and resources.
- **Resource group** Resource groups provide a way to logically organize Azure resources within a subscription for management purposes. Resource groups also serve as a level at which access and governance policies can be applied—for example, if a specific set of policies must be applied to logically grouped resources under a subscription.
- **Resources** Azure resources such as VMs, App Service, Azure Function, Azure Load Balancer, and so on, are leaf nodes in the hierarchy. Based on the policies applied at the preceding levels, access and governance policies are monitored and reported for resources.

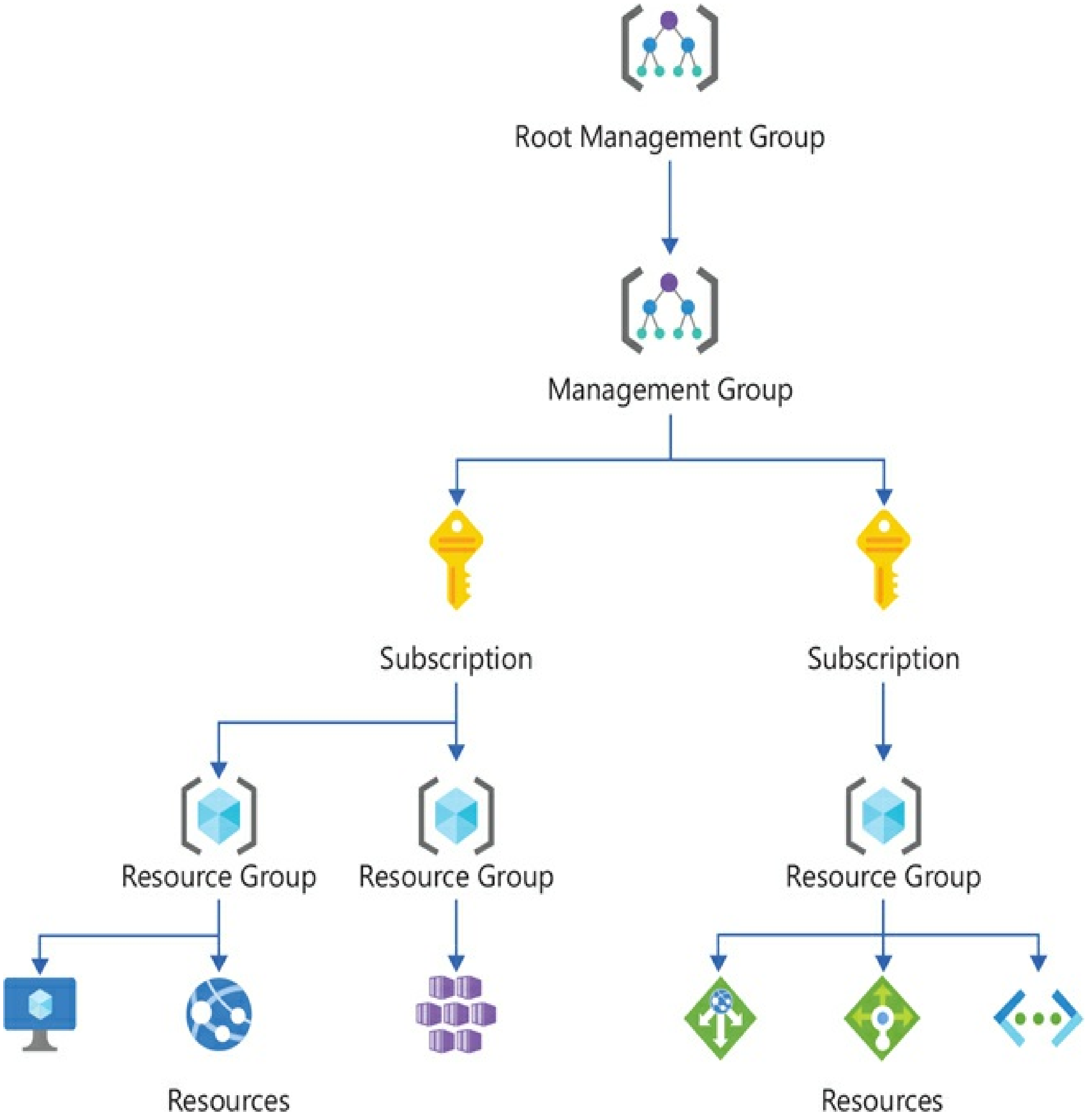


FIGURE 1-14 Hierarchy levels in organizing subscriptions and resources

Management group, subscription, and resource group levels can serve as scope for applying governance policies and budgetary controls. You can also apply access control (that is, Azure RBAC) at all these levels as well as at the resource level.

More Info **Enterprise-Scale Landing Zone**

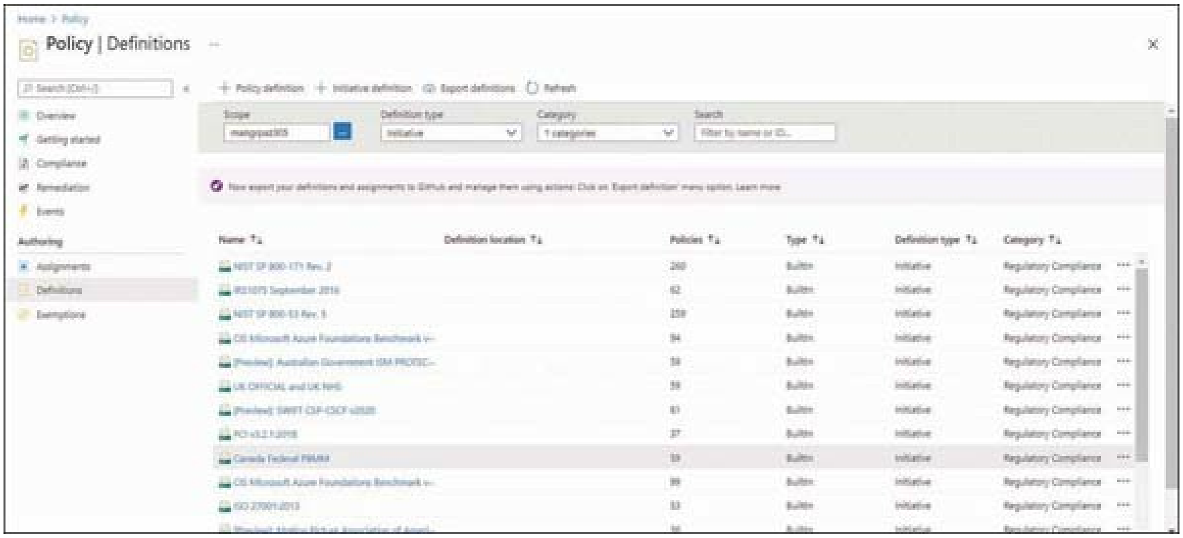
Review the CAF enterprise-scale landing zone at *https://learn.microsoft.com/en-in/azure/cloud-adoption-framework/ready/enterprise-scale/implementation*. This documentation suggests landing zones for different size enterprises and offers recommendations on organizing Azure resources in different hierarchies.

Recommend a solution for enforcing and auditing compliance

As you now understand various scopes at which governance policies can be applied, let's look into the ways and tools available in Azure to enforce policies, perform assessments to identify areas of noncompliance, and remediate those areas.

Azure Policy helps organizations to enforce regulatory, organizational, cost-related, and security-related policies on workloads at scale. Azure policies can be grouped into entities called *initiatives* and applied all at once to a particular scope. There are many built-in policy and initiative definitions available in Azure that can be applied to a scope for most common compliance requirements.

Figure 1-15 show a few built-in initiatives in the Regulatory Compliance category. These help organizations enforce regulatory compliance. The figure shows some examples of initiatives for regulatory compliance, such as Canada Federal PBMM, UK OFFICIAL and UK NHS, and many more. It also shows how many policies are collected in each initiative to allow for the enforcement of compliance with specific regulations. Azure provides initiatives to ensure compliance with almost all regulations out of the box.



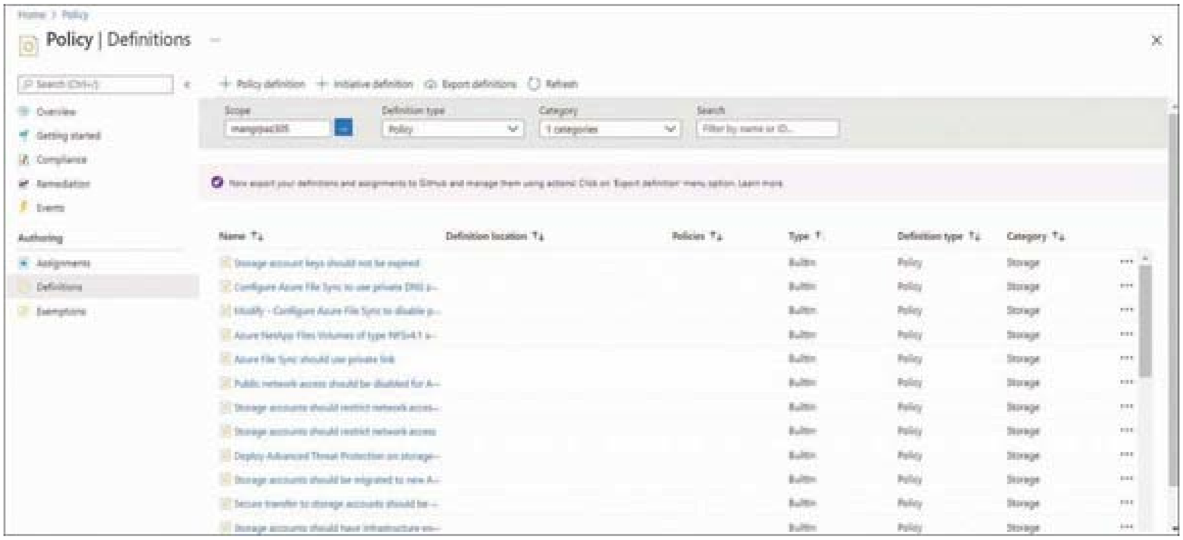
Scope: mangopac305
Definition type: Initiative
Category: 1 categories
Search: Filter by name or ID...

Now export your definitions and assignments to GitHub and manage them using actions! Click on 'Export definition' menu option. Learn more

Name ↑↓	Definition location ↑↓	Policies ↑↓	Type ↑↓	Definition type ↑↓	Category ↑↓
NIST SP 800-171 Rev. 2		260	Builtin	Initiative	Regulatory Compliance
IRS1075 September 2016		62	Builtin	Initiative	Regulatory Compliance
NIST SP 800-53 Rev. 5		259	Builtin	Initiative	Regulatory Compliance
CIS Microsoft Azure Foundations Benchmark v...		94	Builtin	Initiative	Regulatory Compliance
[Preview]: Australian Government ISM PROTEC...		59	Builtin	Initiative	Regulatory Compliance
UK OFFICIAL and UK NHS		59	Builtin	Initiative	Regulatory Compliance
[Preview]: SWIFT CSP-CSCF v2020		61	Builtin	Initiative	Regulatory Compliance
PCI v3.2.1:2018		37	Builtin	Initiative	Regulatory Compliance
Canada Federal PBMM		59	Builtin	Initiative	Regulatory Compliance
CIS Microsoft Azure Foundations Benchmark v...		99	Builtin	Initiative	Regulatory Compliance
ISO 27001:2013		53	Builtin	Initiative	Regulatory Compliance
[Preview]: Motion Picture Association of Ameri...		56	Builtin	Initiative	Regulatory Compliance

FIGURE 1-15 Some of the built-in initiatives to enforce regulatory compliance

Similar to the built-in initiatives, there are built-in Azure policies for many Azure services. Figure 1-16 shows some built-in policies for Azure Storage.



Scope: mangopac305
Definition type: Policy
Category: 1 categories
Search: Filter by name or ID...

Now export your definitions and assignments to GitHub and manage them using actions! Click on 'Export definition' menu option. Learn more

Name ↑↓	Definition location ↑↓	Policies ↑↓	Type ↑	Definition type ↑↓	Category ↑↓
Storage account keys should not be expired			Builtin	Policy	Storage
Configure Azure File Sync to use private DNS z...			Builtin	Policy	Storage
Modify - Configure Azure File Sync to disable p...			Builtin	Policy	Storage
Azure NetApp Files Volumes of type NFSv4.1 s...			Builtin	Policy	Storage
Azure File Sync should use private link			Builtin	Policy	Storage
Public network access should be disabled for A...			Builtin	Policy	Storage
Storage accounts should restrict network acces...			Builtin	Policy	Storage
Storage accounts should restrict network access			Builtin	Policy	Storage
Deploy Advanced Threat Protection on storage...			Builtin	Policy	Storage
Storage accounts should be migrated to new A...			Builtin	Policy	Storage
Secure transfer to storage accounts should be ...			Builtin	Policy	Storage
Storage accounts should have infrastructure en...			Builtin	Policy	Storage

FIGURE 1-16 Some of the built-in policies for a storage account

Although Azure provides many out-of-the-box policies and initiatives, there could be scenarios in which organizations want to enforce a policy or initiative that is not available out of the box. For such scenarios, Azure

Policy allows you to define custom policies and initiatives using Azure Portal, the REST API, PowerShell, or the Azure CLI. For example, suppose an organization wants to restrict public access to the Azure Key Vault and Azure Kubernetes cluster in a particular resource group. Because these policies are not available out of the box, they must be created on a custom basis. They can then be grouped into a custom initiative, which can in turn be assigned to the specific resource group.

When you create a custom policy or initiative, you must provide a definition location. This definition location can be the root management group, a management group, or a subscription. Policies and initiatives can be defined only at these levels, although defined policies and initiatives can be assigned at the root management group, management group, subscription, or resource group scope.

When assigning a policy or initiative to a particular scope, you must specify the following information:

- Scope
- Exclusions (if the policy must exclude certain resources within the same scope)
- Parameters required by the policy or initiative (like location, Log Analytics workspace, and so on)
- Remediation identity (this can be a system-managed identity or a user-managed identity)
- A noncompliance message (the user will see this if they initiate a noncompliant action for an Azure resource)

More Info **Custom Policies Available at GitHub**

To review custom policies available at GitHub, see *https://github.com/Azure/Enterprise-Scale/blob/main/docs/ESLZ-Policies.md*.

Figure 1-17 shows the Compliance dashboard, which provides visibility of any policies or initiatives with which your deployment is not complying.

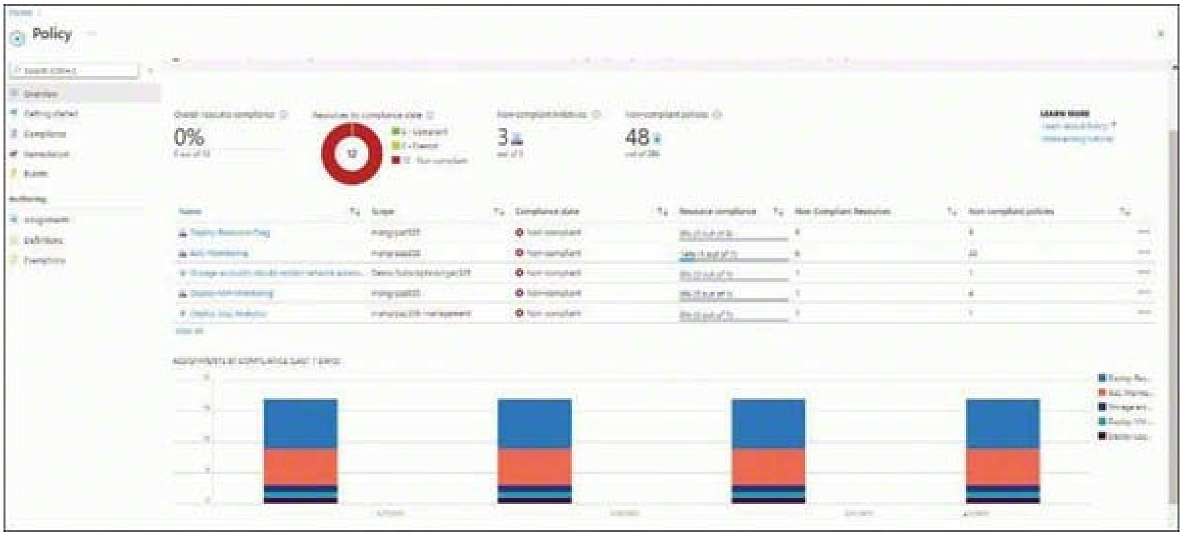


FIGURE 1-17 Compliance dashboard

With each policy definition, there is a provision to specify what response should occur when a resource is identified as noncompliant. This response is configured as an *effect*. Each policy definition in Azure Policy can have a single effect. An effect can be any one of the following:

- **Append** This effect appends additional fields to the request before it is sent to the resource provider to create or update the resource. If the original request does not have the field to be appended, it is added to the request. If the append effect would override a value in the original request with a different value, then it acts as a deny effect and rejects the request.

  If the field in question is of type array, the append effect can specify that the field be added to that array if the original incoming request specifies the values in that array type field. This is done by appending an alias [*] for the field while defining an append effect in Azure Policy.

  If the policy with the append effect is evaluated during the evaluation cycle (rather than during a create or update action on the resource) and an existing resource is evaluated to be noncompliant, the append effect simply marks that resource as noncompliant.

  An example of using the Append effect would be for a key vault. You would enable the Azure Disk Encryption for Volume Encryption Key

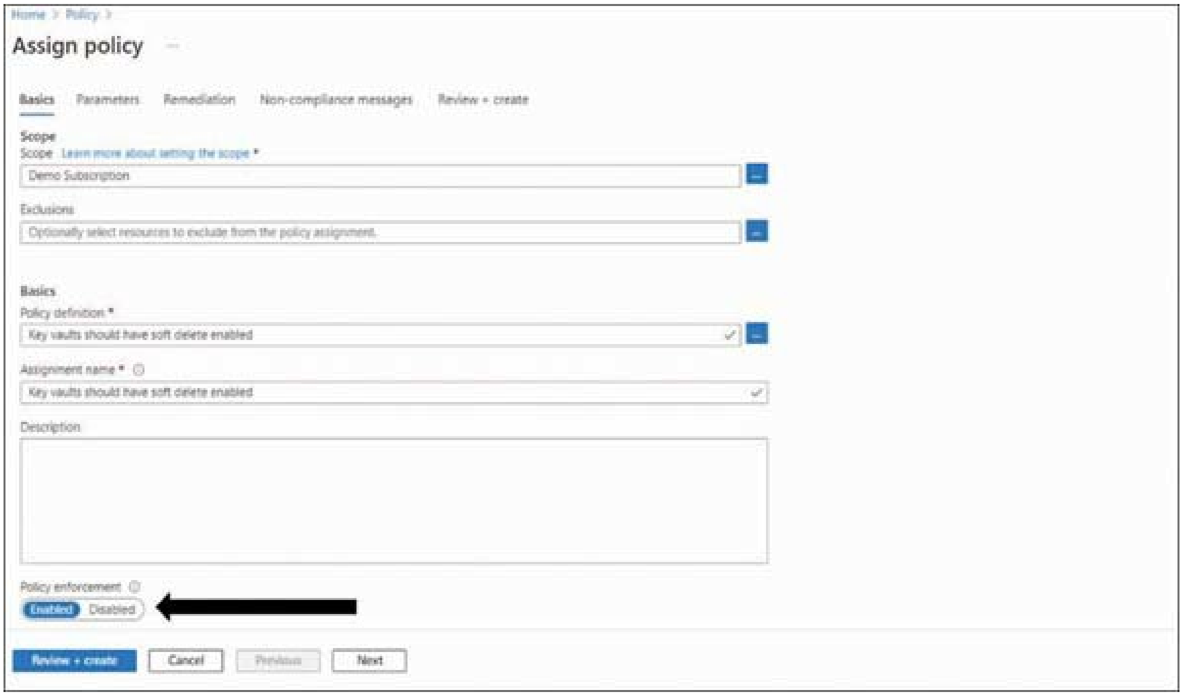
Vault access policy while the key vault is being created or updated.

- **Audit** This effect records a warning in the activity log for the resource if the resource is evaluated to be noncompliant while being created or updated. It does not stop the update or creation of the resource, however. If the policy with the audit effect is evaluated during the evaluation cycle and an existing resource is evaluated to be noncompliant, then the audit effect simply marks that resource as noncompliant.
- **AuditIfNotExists** This effect audits the existence of related resources that are relevant to the resource being evaluated. If you are creating or updating a resource, this effect runs after the resource has been created or updated. If the policy evaluates that the related resource does not exist, this effect creates an audit record in the activity log for the resource that got created or updated and marks the resource as noncompliant.

  If a policy is being evaluated as part of an evaluation cycle and the related resources are not found to exist, then the resource is marked noncompliant. For example, suppose an organization wants to enforce a policy whereby every VM that is created should have a particular extension when it is provisioned. A policy could define an AuditifNotExists effect to audit any VM being created or updated, identify any VMs in the evaluation cycle that do not have the extension, and mark those VMs as noncompliant.
- **Deny** As the name suggest, this effect will deny the creation or update of any non-compliant resources. Requests are denied with a 403 (forbidden) code. If the policy evaluates the resource as noncompliant during an evaluation cycle, this effect marks the resource as noncompliant.
- **DeployIfNotExists** This effect is similar to AuditIfNotExists, except that this effect executes a template to deploy needed resources for the identified noncompliant resource rather than marking the resource as noncompliant. The policy assignment of a policy having the DeployIfNotExists effect requires managed identity to take remediation action. Although this effect runs a template to remediate noncompliant resources when it is created or updated, during the triggered evaluation

cycle it simply marks noncompliant resources as such without carrying out remediation. Existing noncompliant resources can be remediated using a remediation task.

- **Disabled** You can use this effect when a policy definition accepts the effect itself as a parameter during policy assignment, such that a user has the option to select the effect for policy enforcement. If the user does not select the effect, the policy will not be enforced. While this effect is available, there is also an option to disable policy enforcement at the time of policy assignment. If policy enforcement is marked as Disabled in the policy assignment, then effects will not be enforced. For example, the Deny effect will not deny noncompliant requests. (See Figure 1-18.)



Scope: Demo Subscription

Exclusions: Optionally select resources to exclude from the policy assignment.

Policy definition *: Key vaults should have soft delete enabled

Assignment name *: Key vaults should have soft delete enabled

Description:

Policy enforcement: Enabled

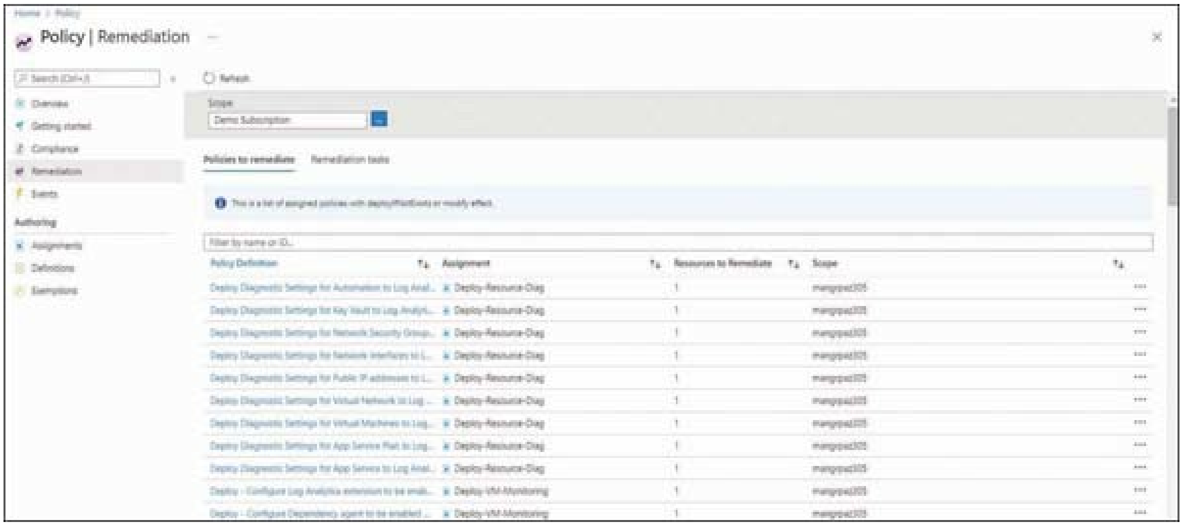
FIGURE 1-18 Option to disable policy enforcement

- **Modify** This effect enables you to add, update, or remove properties or tags on a subscription or resource during creation or update. You can use this effect to update tags for a resource. A single modify rule can have one or more operations—for example, removing certain existing tags and then adding new tags to a resource. Similar to the

DeployIfNotExists effect, the policy assignment of a policy with the Modify effect requires a managed identity to carry out remediation tasks. And like DeployIfNotExists, a Modify effect simply marks a noncompliant resource as such while evaluating the policy in the triggered evaluation cycle; no remediation action will be done. Existing noncompliant resources can be remediated using a remediation task.

The managed identity—either system-managed or user-managed—that is configured at policy assignment for policies with a Modify or DeployIfNotExists effect must have appropriate permissions to carry out remediation tasks.

You can see resources identified as noncompliant during evaluation cycles on the Remediation page in Azure Policy. (See Figure 1-19.) You can create a remediation task to remediate noncompliant resources.



This is a list of assigned policies with deployIfNotExists or modify effect.

Policy Definition	Assignment	Resources to Remediate	Scope
Deploy Diagnostic Settings for Automation to Log Anal...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for Key Vault to Log Analyti...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for Network Security Group...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for Network Interfaces to L...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for Public IP addresses to L...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for Virtual Network to Log ...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for Virtual Machines to Log...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for App Service Plan to Log...	Deploy-Resource-Diag	1	mangrpaz205
Deploy Diagnostic Settings for App Service to Log Anal...	Deploy-Resource-Diag	1	mangrpaz205
Deploy - Configure Log Analytics extension to be enab...	Deploy-VM-Monitoring	1	mangrpaz205
Deploy - Configure Dependency agent to be enabled ...	Deploy-VM-Monitoring	1	mangrpaz205

FIGURE 1-19 Noncompliance identified by policy having a Modify or DeployIfNotExists effect during evaluation cycle

More Info **Remediation Tasks**

See the Microsoft documentation for more information about remediation tasks, at *https://learn.microsoft.com/en-us/azure/governance/policy/how-to/remediate-resources?tabs=azure-portal*.

Skill 1.4: Design identities and access for applications

In this section, you learn how applications and services—whether deployed in Azure, on-premises, or on any other cloud—authenticate against AAD to access Azure resources and services. You also learn how applications and services can rely on AAD to authenticate users, irrespective of where those applications and services are deployed. Finally, this section discusses how to securely store secrets and passwords in Azure, which can be referenced by applications.

You learned about security principals earlier in this chapter, in Skill 1.2. To summarize:

- A service principal is a kind of security principal.
- There are two types of service principals: application and managed identity.
- A managed identity can be a system-assigned managed identity or a user-assigned managed identity.

TIP To review these concepts, refer to Skill 1.2.

This section covers how to:

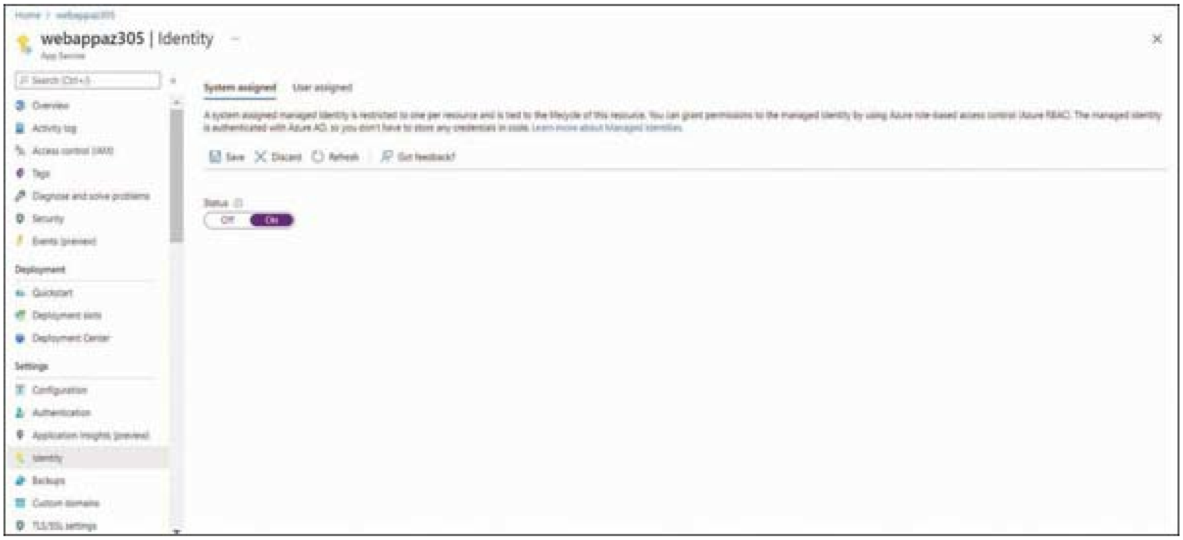
- Recommend solutions to allow applications to access Azure resources
- Recommend a solution that securely stores passwords and secrets
- Recommend a solution for integrating applications into Azure Active Directory (Azure AD)
- Recommend a user consent solution for applications

Recommend solutions to allow applications to access

Azure resources

Whether deployed in Azure VM, in an Azure PaaS service, or outside Azure, applications must have a service principal in AAD to provide them appropriate authorization and access to Azure resources. For example, consider a web application deployed in Azure App Service as an Azure web app. You need the Azure web app to access an Azure storage blob. Because the application is deployed in Azure, you can use a managed identity to provide the Azure web app necessary access to the Azure storage blob.

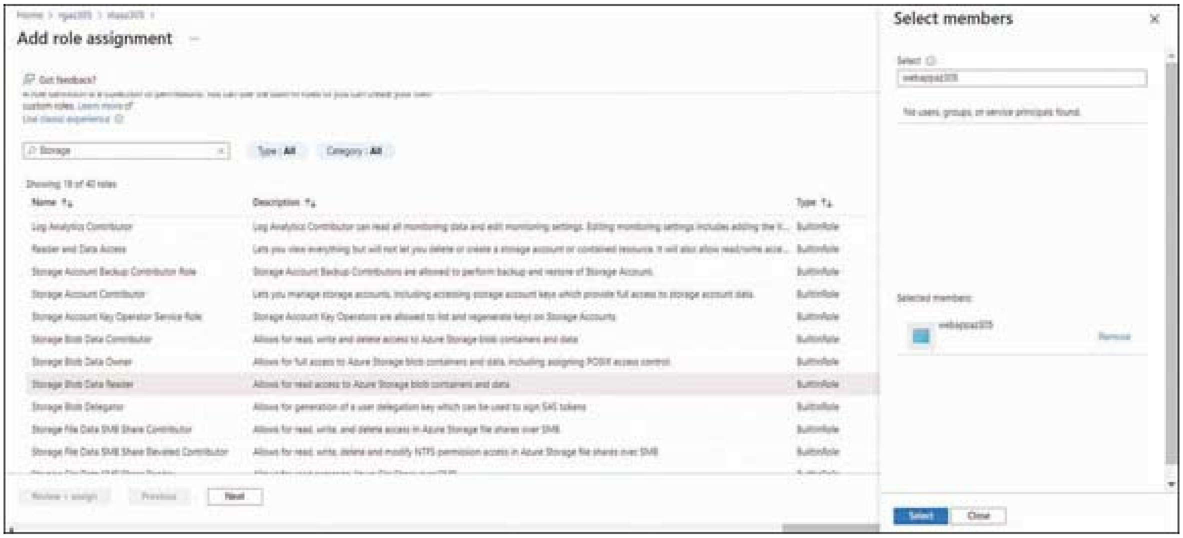
Now suppose that this Azure web app must be given only read access on the storage blob. This would involve assigning a managed identity (either a system-assigned managed identity or a user-assigned managed identity) to the Azure web app. Then, in the storage account's Access Control (IAM), you would assign the Storage Blob Data Reader role to the Azure web app managed identity. (See Figures 1-20 and 1-21.)



A system assigned managed identity is restricted to one per resource and is tied to the lifecycle of this resource. You can grant permissions to the managed identity by using Azure role-based access control (Azure RBAC). The managed identity is authenticated with Azure AD, so you don't have to store any credentials in code. Learn more about Managed identities.

Status: Off [✓ On]

FIGURE 1-20 Enable a system-assigned managed identity for an Azure web app.



Home > rgaz305 > staaz305 >

Add role assignment

Got feedback?

[illegible] custom roles. Learn more

Use classic experience

Storage

Type : All

Category : All

Showing 19 of 40 roles

Name	Description	Type
Log Analytics Contributor	Log Analytics Contributor can read all monitoring data and edit monitoring settings. Editing monitoring settings includes adding the V...	BuiltInRole
Reader and Data Access	Lets you view everything but will not let you delete or create a storage account or contained resource. It will also allow read/write acce...	BuiltInRole
Storage Account Backup Contributor Role	Storage Account Backup Contributors are allowed to perform backup and restore of Storage Account.	BuiltInRole
Storage Account Contributor	Lets you manage storage accounts, including accessing storage account keys which provide full access to storage account data.	BuiltInRole
Storage Account Key Operator Service Role	Storage Account Key Operators are allowed to list and regenerate keys on Storage Accounts	BuiltInRole
Storage Blob Data Contributor	Allows for read, write and delete access to Azure Storage blob containers and data	BuiltInRole
Storage Blob Data Owner	Allows for full access to Azure Storage blob containers and data, including assigning POSIX access control.	BuiltInRole
Storage Blob Data Reader	Allows for read access to Azure Storage blob containers and data	BuiltInRole
Storage Blob Delegator	Allows for generation of a user delegation key which can be used to sign SAS tokens	BuiltInRole
Storage File Data SMB Share Contributor	Allows for read, write, and delete access in Azure Storage file shares over SMB	BuiltInRole
Storage File Data SMB Share Elevated Contributor	Allows for read, write, delete and modify NTFS permission access in Azure Storage file shares over SMB	BuiltInRole

Select members

Select: webappaz305

No users, groups, or service principals found.

Selected members:

webappaz305

FIGURE 1-21 Assign the Storage Blob Data Reader role to the managed identity of the Azure web app.

If an application is not deployed in Azure and hence cannot use a managed identity, or if for any reason you simply do not want to use a managed identity, you must create an application service principal for the application. This service principal can then be given access in a similar manner to managed identity on Azure resources, as described earlier.

Recommend a solution that securely stores passwords and secrets

There are situations in which applications need to deal with secrets—for example, database connection strings, credentials, passwords, certificates, and so on. Storing these secrets as part of the application code and configuration can result in a security breach if a malicious actor gets hold of them. Azure Key Vault is an Azure service whose purpose is to securely store secrets.

It is important to understand the objects that Key Vault helps to secure:

- **Keys** These are the encryption keys used for data encryption.
- **Secrets** These can be passwords, connection strings, API keys, or any other secrets.
- **Certificates** TLS/SSL certificates are used to encrypt data in transit.

Key Vault offers features for secret management, key management, and certificate management.

- Secret management enables the secure storage and disposal of passwords, API keys, access tokens, and so on.
- Key management allows for the creation, importing, storage, recycling, and disposal of data encryption keys.
- Certificate management provides for the provisioning, importing, and management of TLS/SSL certificates.

Key Vault provides three ways to generate certificates:

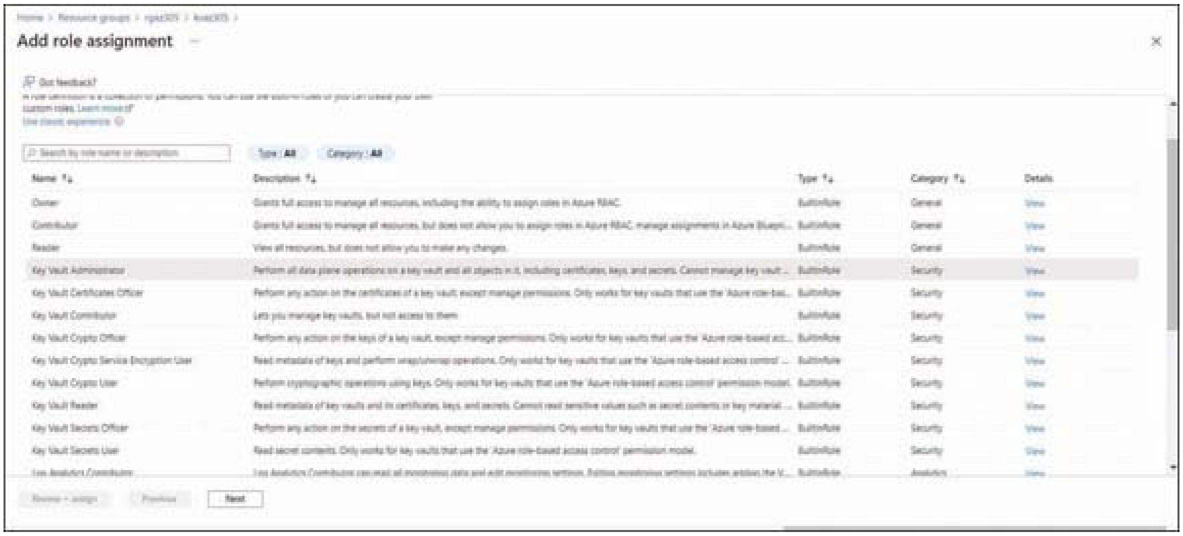
- **Self-signed certificates** These can be generated for development and testing purposes.
- **Integrated certificate authority (CA) certificates** You can configure DigiCert and GlobalSign accounts in Key Vault to enable the generation of certificates from these CAs.
- **Non-integrated CAs** You can generate certificates manually if your CA is not integrated with Key Vault. Key Vault also supports the importing of existing certificates.

Key Vault has two planes from an operations perspective:

- **Management** This operations plane relates to the management of the Key Vault itself, such as creating, updating the access policy for, and deleting a Key Vault.
- **Data** This operations plane relates to the management of data stored in Key Vault, such as creating, reading, updating, and deleting keys, secrets, and certificates.

Key Vault relies on AAD to authenticate requests for operations in both planes. Requests for management-plane operations are authorized with the help of Azure RBAC, while requests for data-plane operations are authorized using a Key Vault access policy as well as Azure RBAC for Key Vault data operations.

A managed identity or service principal can be given an appropriate role on Key Vault to enable operations in both planes. This is similar to setting up Access Control (IAM) on Azure services, as you saw earlier for the storage account. Here, roles can be assigned to a security principal in Key Vault, as shown in Figure 1-22.



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

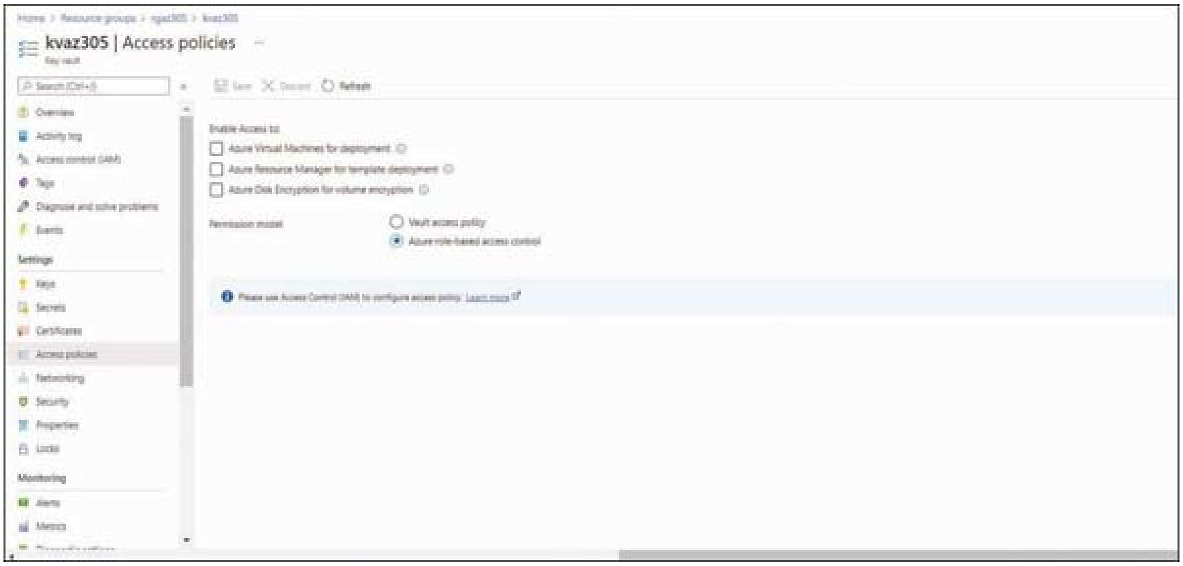
Name	Description	Type	Category	Details
Owner	Grants full access to manage all resources, including the ability to assign roles in Azure RBAC.	BuiltInRole	General	View
Contributor	Grants full access to manage all resources, but does not allow you to assign roles in Azure RBAC, manage assignments in Azure Bluepri...	BuiltInRole	General	View
Reader	View all resources, but does not allow you to make any changes.	BuiltInRole	General	View
Key Vault Administrator	Perform all data plane operations on a key vault and all objects in it, including certificates, keys, and secrets. Cannot manage key vault ...	BuiltInRole	Security	View
Key Vault Certificates Officer	Perform any action on the certificates of a key vault, except manage permissions. Only works for key vaults that use the 'Azure role-bas...	BuiltInRole	Security	View
Key Vault Contributor	Lets you manage key vaults, but not access to them.	BuiltInRole	Security	View
Key Vault Crypto Officer	Perform any action on the keys of a key vault, except manage permissions. Only works for key vaults that use the 'Azure role-based acc...	BuiltInRole	Security	View
Key Vault Crypto Service Encryption User	Read metadata of keys and perform wrap/unwrap operations. Only works for key vaults that use the 'Azure role-based access control' ...	BuiltInRole	Security	View
Key Vault Crypto User	Perform cryptographic operations using keys. Only works for key vaults that use the 'Azure role-based access control' permission model.	BuiltInRole	Security	View
Key Vault Reader	Read metadata of key vaults and its certificates, keys, and secrets. Cannot read sensitive values such as secret contents or key material. ...	BuiltInRole	Security	View
Key Vault Secrets Officer	Perform any action on the secrets of a key vault, except manage permissions. Only works for key vaults that use the 'Azure role-based...	BuiltInRole	Security	View
Key Vault Secrets User	Read secret contents. Only works for key vaults that use the 'Azure role-based access control' permission model.	BuiltInRole	Security	View
Log Analytics Contributor	Log Analytics Contributor can read all monitoring data and edit monitoring settings. Editing monitoring settings includes adding the V...	BuiltInRole	Analytics	View

FIGURE 1-22 Role assignment for Azure Key Vault

An access policy helps provide authorization access to the data plane. Authorization can be done using Azure RBAC or a Key Vault access policy. It allows you to enable access of Azure services on Azure Key Vault and to specify a permission model for data-plane authorization.

Access can be enabled for Azure services as follows (see Figure 1-23):

- **Azure VM for deployment** VMs can retrieve certificates from secrets in a Key Vault.
- **Azure Resource Manager for template deployment** Azure Resource Manager can retrieve secrets from a Key Vault while deploying a template.
- **Azure Disk Encryption for volume encryption** The Azure Disk Encryption service can retrieve a key from a Key Vault and unwrap it, as required to encrypt disks.



Enable Access to:

□ Azure Virtual Machines for deployment

□ Azure Resource Manager for template deployment

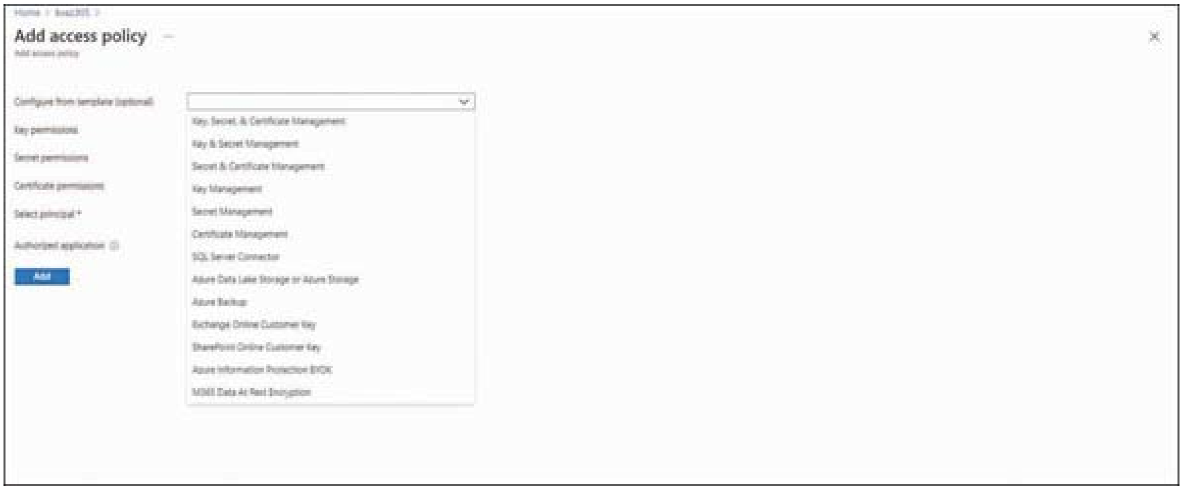
□ Azure Disk Encryption for volume encryption

Permission model ○ Vault access policy ● Azure role-based access control

Please use Access Control (IAM) to configure access policy. Learn more

FIGURE 1-23 Access policies for Azure Key Vault

You can also configure vault access policies. A vault access policy is an alternative to Azure RBAC to provide permission on the Key Vault data plane. Vault access policies have a number of permission templates, as shown in Figure 1-24.



Key, Secret, & Certificate Management
Key & Secret Management
Secret & Certificate Management
Key Management
Secret Management
Certificate Management
SQL Server Connector
Azure Data Lake Storage or Azure Storage
Azure Backup
Exchange Online Customer Key
SharePoint Online Customer Key
Azure Information Protection BYOK
M365 Data At Rest Encryption

FIGURE 1-24 Permission templates for assigning permissions while configuring the Key Vault access policy

Each permission template provides a specific set of permissions for keys, secrets, and certificates. For example, selecting the Key Management permission template provides key management operations permissions and rotation policy operations permissions, as described in Table 1-3.

TABLE 1-3 Key Management permission template permissions

Key Management Operations Permissions	Rotation Policy Operations Permissions
Get	Rotate
List	Get Rotation Policy
Update	Set Rotation Policy
Create	
Import	
Delete	
Recover	
Backup	
Restore	

> ***Note*** Although permission templates are available, their use is optional. If you prefer, you can set permissions individually for keys, secrets, and certificates.

> ***Note*** For new deployments, it is recommended to use the Azure RBAC model for dataplane operation authorization.

> ***More Info*** **Key Vault Access Policy**
>
> To see the step-by-step procedure for assigning Key Vault access policies, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/key-vault/general/assignaccess-policy?tabs=azure-portal*.

Azure Key Vault provides two types of storage for cryptographic keys: vault and managed hardware security module (HSM). Table 1-4 compares these two types of storage.

TABLE 1-4 Types of storage for cryptographic keys

Vault	Managed HSM
Supports software-protected keys and HSM-protected keys. HSM protection is available only in the Azure Key Vault Premium SKU.	Supports only HSM-protected keys.
Multitenant	Single tenant
Software-protected key: FIPS 140-2 Level 1 HSM-protected key: FIPS 140-2 Level 2	FIPS 140-2 Level 3
Used for low cost. Can be used where compliance requirements are less than FIPS 140-2 Level 3.	Used for high-value keys. Used when there is a specific requirement for FIPS 140-2 Level 3 compliance.

Recommend a solution for integrating applications into Azure Active Directory (Azure AD)

Applications, whether deployed in Azure, on-premises, on the edge, or in another public cloud, can rely on AAD for authenticating the users. You will learn in this section how to integrate applications with AAD for authenticating the users.

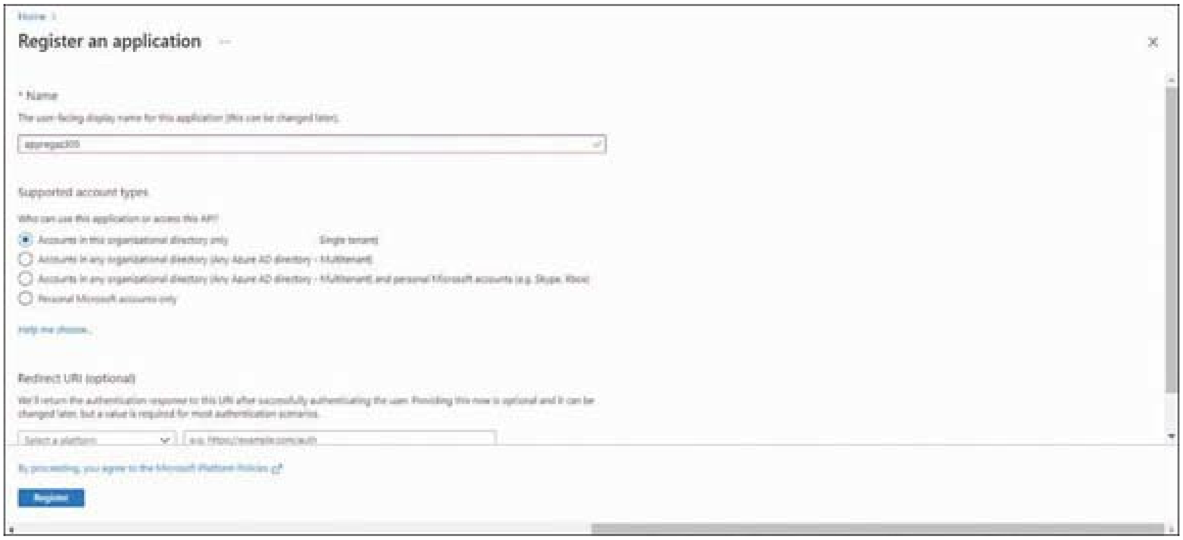
Application registration

We touched on application registration in the context of IDAM and the service principal earlier in this chapter. We will continue to discuss application registration in the context of the topic of this section.

Application developers can offload identity management and authentication functions to AAD. This requires the registration of the application in AAD. Registering an application creates a globally unique application object in the home tenant where the application is registered.

While registering an application in AAD, a developer can provide application access to the following (see Figure 1-25):

- Users belonging to the application’s home AAD tenant
- Users belonging to any AAD tenant of any organization
- Users with a Microsoft account



Home >

Register an application

* Name

The user-facing display name for this application (this can be changed later).

appregad05

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only (Single tenant)

○ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

○ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

○ Personal Microsoft accounts only

Help me choose...

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Select a platform | e.g. https://example.com/auth

By proceeding, you agree to the Microsoft Platform Policies

Register

FIGURE 1-25 Application registration

Developers writing line of business (LOB) applications can use the single-tenant option, as only users within a specific organization are expected to use the application. If an application is meant for a B2B or B2C scenario, the multitenant option or the multitenant and personal Microsoft accounts option can be used. With a multitenant option, users with an account in another organization’s AAD can be authenticated to consume the application. (You will learn more about this option in the next section.)

As discussed, when an application is registered through the Azure Portal, the application service principal (service principal object) is also created in the home AAD tenant. If, however, an application is registered through Microsoft Graph API (AAD API is part of Microsoft Graph API), the application service principal must be created separately.

Once an application is registered in AAD, various configuration options are made available for it. Some important configuration options are as follows:

- **Branding and organization properties** Use these options to supply a logo, home page URL, terms of service page URL, privacy statement page URL, the domain that users see on the consent screen, and so on.
- **Authentication** These settings enable you to specify additional settings based on the platform or device this application registration is targeting. You can also specify the logout URL and, importantly, the token that will be issued when the request is successfully authenticated. This can be an access token, an ID token, or both. In the case of an implicit grant flow, in which a single page application or web API is consumed by JavaScript, both an access token and an ID token can be sent as a response upon successful authentication. If the application is an ASP.net Core web app or any other web app using a hybrid authentication flow, only an ID token can be sent.

More Info **Understand Implicit Grant Flow**

For more information about OAuth 2.0 implicit grant flow, see the Microsoft documentation at *https://learn.microsoft.com/en-us/azure/active-directory/develop/v2-oauth2-implicitgrant-flow?WT.mc_id=Portal-Microsoft_AAD_RegisteredApps*.

- **Certificates and secrets** You can configure these settings to provide a higher level of assurance that the authentication request or the request to retrieve the tokens is from an authentic source.
- **Token configuration** These settings allow you to include additional claims in the token returned in response to a successful authentication request.
- **Expose API** A developer can use this setting to integrate their REST APIs with AAD to enable authorized users or client applications to access these APIs with delegated permissions. Multiple scopes can be defined for the API such that each one can be configured to require consent from the admin, the user, or both.

- **API permission** Use this setting to give the client app being registered access to other APIs, such as Microsoft APIs or any APIs within an organization exposed through the Expose API option in a separate application registration in AAD.

Enterprise applications

An application service principal is essentially a local representation of the globally unique application object. The service principal inherits certain configurations and properties from the global application object. In a sense, the application object serves as a template for the application's service principal object. Service principals of all the applications registered in a particular AAD tenant are available as Enterprise applications within the same AAD tenant.

Enterprises can integrate SaaS applications from the AAD application gallery, nongallery applications, and on-premises applications. Integration of on-premises applications requires the deployment of an Application Proxy connector in the on-premises environment on the Windows server. Integrating applications in AAD as enterprise applications creates a service principal of globally exposed SaaS applications (whether from the AAD application gallery or not) or an on-premises application.

Once the service principal is created for an application, the following settings can be applied to it:

- You can assign users and groups to access the application.
- You can configure Single Sign-On (SSO) for applications integrated from the AAD application gallery and even for nongallery applications. Note that SSO is not available for applications registered by the developer using the application-registration approach discussed in the preceding section. Developers can use OpenID Connect and OAuth to provide SSO features, however. SDKs are available for a number of programming languages as part of the Microsoft Authentication Library (MSAL) to easily enable the use of the Microsoft Graph API.
- You can configure automatic provisioning of accounts for all registered applications, whether they are gallery applications, nongallery applications, or on-premises applications. You can manage the identity

lifecycle in AAD; the lifecycle of user accounts is automatically managed in the application. Note that this feature is not available for applications registered by the developer using the application-registration approach described in the preceding section.

- You can configure conditional access policies for registered applications, in a manner similar to the one described earlier in this chapter.
- You can enable self-service access requests for enterprise applications.

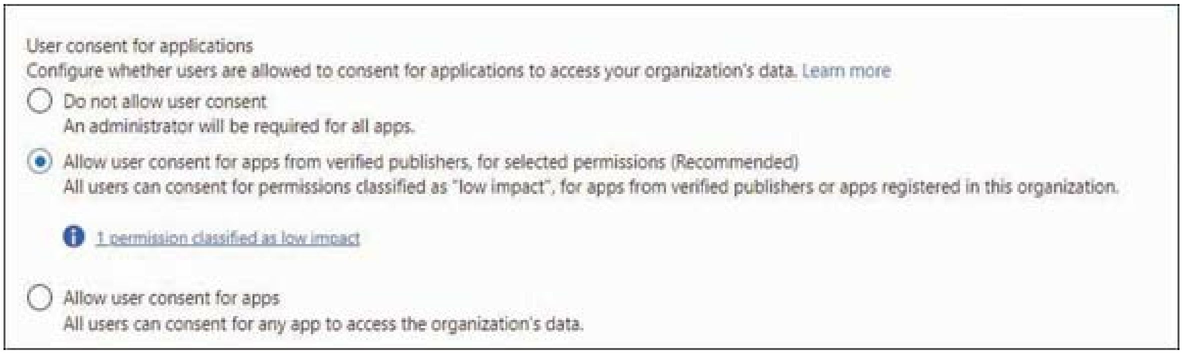
Recommend a user consent solution for applications

An application or service that tries to access organizational data or services should not be able to do so without proper consent in place. Applications and services should obtain consent in one of the following ways:

- By asking a user to use their identity to access organization data or services
- By having the administrator provide consent to the application on behalf of all users

Enterprise applications allow for consent-related configuration at the AAD tenant level for an organization.

Global administrators can configure user consent (see Figure 1-26) and group owner consent (see Figure 1-27) at the AAD tenant level for applications accessing organization data.



User consent for applications

Configure whether users are allowed to consent for applications to access your organization's data. Learn more

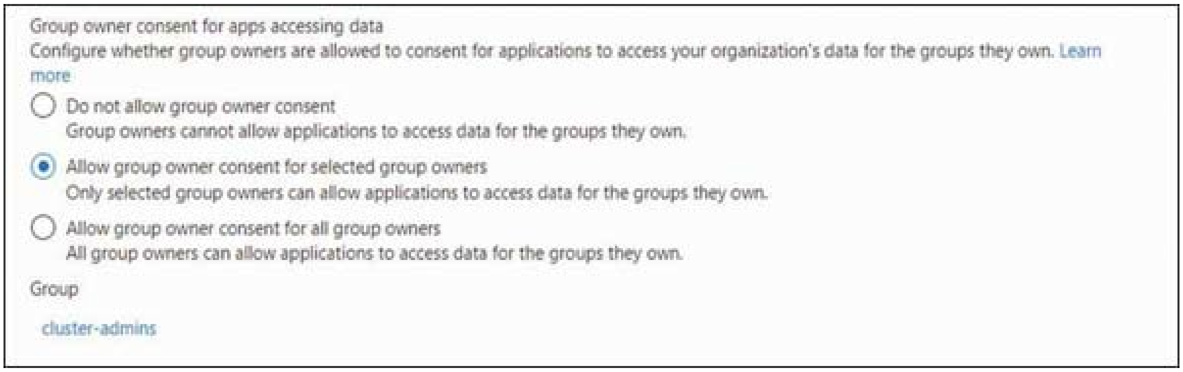
○ Do not allow user consent
An administrator will be required for all apps.

● Allow user consent for apps from verified publishers, for selected permissions (Recommended)
All users can consent for permissions classified as "low impact", for apps from verified publishers or apps registered in this organization.

1 permission classified as low impact

○ Allow user consent for apps
All users can consent for any app to access the organization's data.

FIGURE 1-26 User consent for application configuration



Group owner consent for apps accessing data

Configure whether group owners are allowed to consent for applications to access your organization's data for the groups they own. Learn more

○ Do not allow group owner consent
Group owners cannot allow applications to access data for the groups they own.

● Allow group owner consent for selected group owners
Only selected group owners can allow applications to access data for the groups they own.

○ Allow group owner consent for all group owners
All group owners can allow applications to access data for the groups they own.

Group

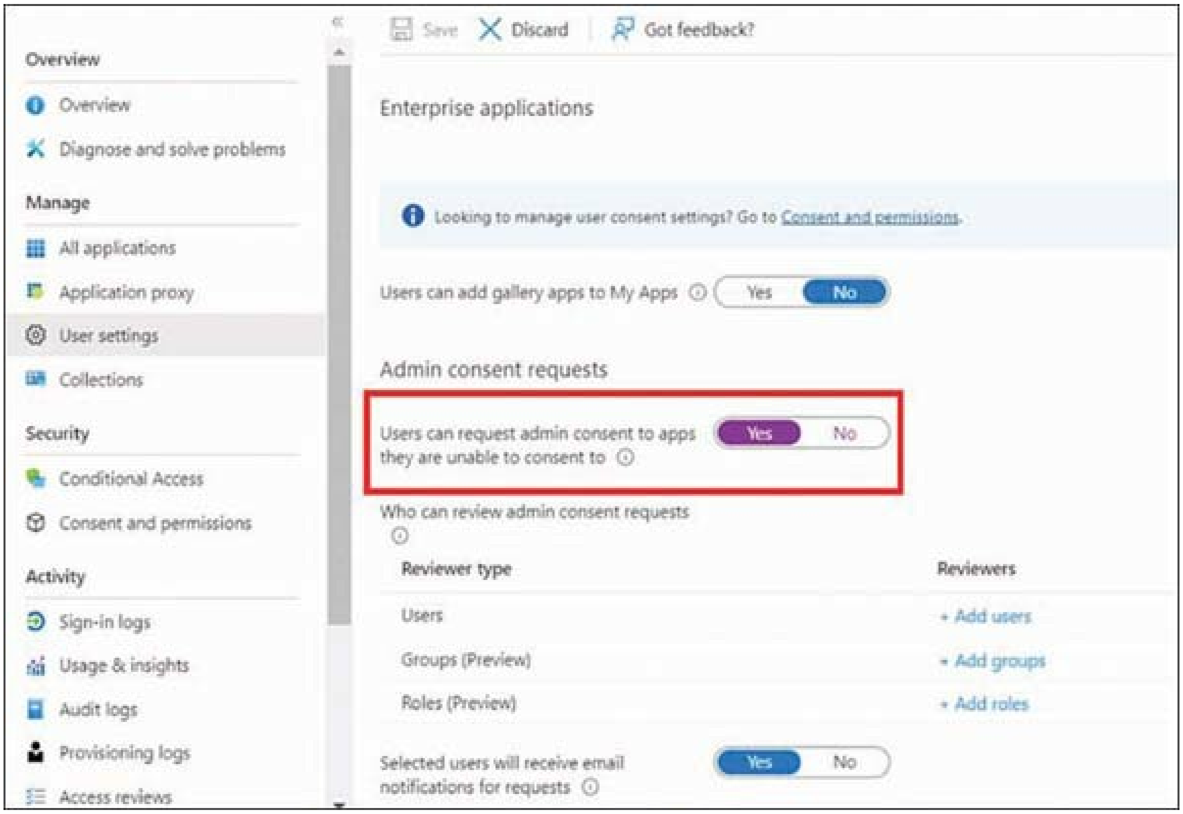
cluster-admins

FIGURE 1-27 Group owner consent for application configuration

Global administrators, application administrators, and cloud application administrators can configure permission classifications for user consent for enterprise applications. Presently, only low-risk permission classification can be done for permissions. Permissions that do not require admin consent can be added to this classification. Examples of low-risk permissions include the following:

- The User.Read permission on Microsoft Graph, which gives permission to read users from the AAD tenant
- The email permission on Microsoft Graph, which gives permission to view a user’s email address

The Enterprise Application experience also enables a scenario in which a user can request that an admin provide consent when they (the user) cannot provide the required consent to the application to access specific data or a specific service. The admin then reviews the request and provides the necessary admin consent (or not). The reviewing admin must be a global administrator, an application administrator, or a cloud application administrator. (See Figure 1-28.)



Enterprise applications

Looking to manage user consent settings? Go to Consent and permissions.

Users can add gallery apps to My Apps: Yes [✓ No]

Admin consent requests

Users can request admin consent to apps they are unable to consent to: [✓ Yes] No

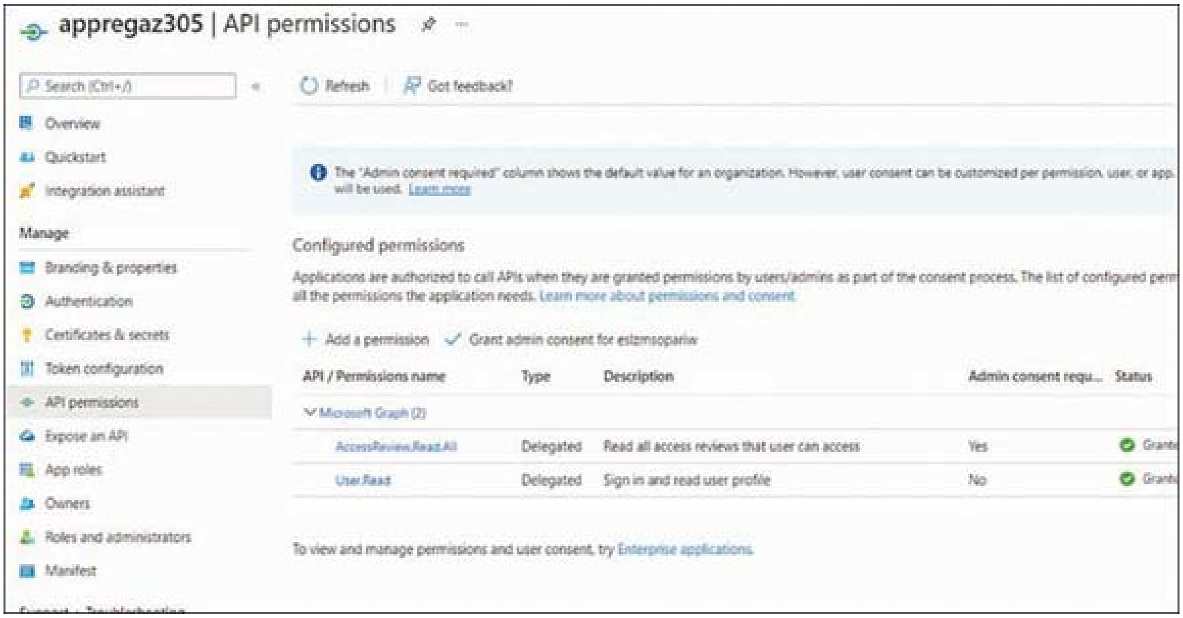
Who can review admin consent requests

Reviewer type	Reviewers
Users	+ Add users
Groups (Preview)	+ Add groups
Roles (Preview)	+ Add roles

Selected users will receive email notifications for requests: [✓ Yes] No

FIGURE 1-28 Enabling admin consent request for users in Enterprise Application experience

You can also configure consent at the application object level (see Figure 1-29) and at the application service principal level (see Figure 1-30). To configure consent at the application object level, you can use API permissions, as discussed in the section about application registration.



The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. will be used. Learn more

Configured permissions

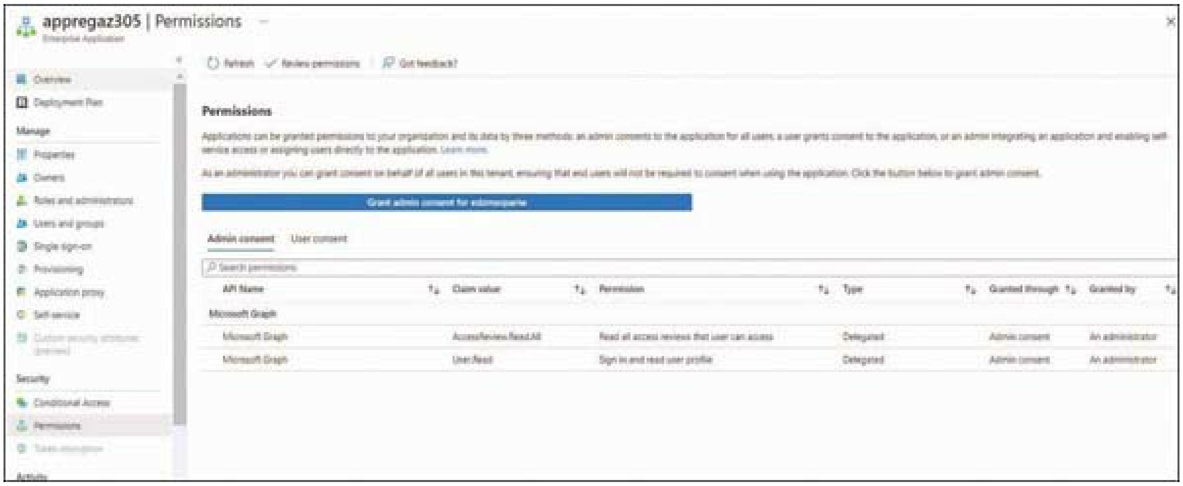
Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured perm all the permissions the application needs. Learn more about permissions and consent

+ Add a permission ✓ Grant admin consent for eslzmsopariw

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (2)				
AccessReview.Read.All	Delegated	Read all access reviews that user can access	Yes	✓ Grant
User.Read	Delegated	Sign in and read user profile	No	✓ Grant

To view and manage permissions and user consent, try Enterprise applications.

FIGURE 1-29 Consent configured in application registration



Permissions

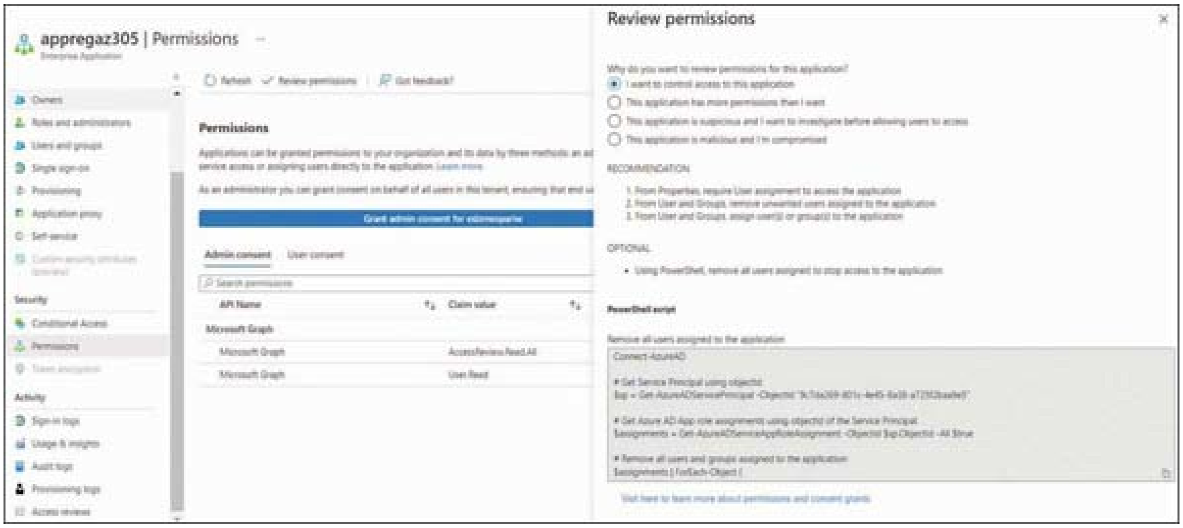
Applications can be granted permissions to your organization and its data by three methods: an admin consents to the application for all users, a user grants consent to the application, or an admin integrating an application and enabling self-service access or assigning users directly to the application. Learn more

As an administrator you can grant consent on behalf of all users in this tenant, ensuring that end users will not be required to consent when using the application. Click the button below to grant admin consent.

API Name	Claim value	Permission	Type	Granted through	Granted by
Microsoft Graph					
Microsoft Graph	AccessReview.Read.All	Read all access reviews that user can access	Delegated	Admin consent	An administrator
Microsoft Graph	User.Read	Sign in and read user profile	Delegated	Admin consent	An administrator

FIGURE 1-30 Grant admin consent in service principal for an application.

While there are numerous ways to configure consent—either admin or user consent—it is important to be able to review consent. As shown in Figure 1-31, you can review consent given to an application service principal within enterprise applications.



Permissions

Applications can be granted permissions to your organization and its data by three methods: an ad… service access or assigning users directly to the application.

As an administrator you can grant consent on behalf of all users in this tenant, ensuring that end u…

API Name	Claim value
Microsoft Graph	
Microsoft Graph	AccessReview.Read.All
Microsoft Graph	User.Read

Review permissions

Why do you want to review permissions for this application?

● I want to control access to this application

○ This application has more permissions than I want

○ This application is suspicious and I want to investigate before allowing users to access

○ This application is malicious and I'm compromised

RECOMMENDATION

1. From Properties, require User assignment to access the application
2. From User and Groups, remove unwanted users assigned to the application
3. From User and Groups, assign user(s) or group(s) to the application

OPTIONAL

- Using PowerShell, remove all users assigned to stop access to the application

PowerShell script

Remove all users assigned to the application

```
Connect-AzureAD

# Get Service Principal using objectId
$sp = Get-AzureADServicePrincipal -ObjectId "9c7da269-801c-4e45-8a38-a72352baa9e5"

# Get Azure AD App role assignments using objectId of the Service Principal
$assignments = Get-AzureADServiceAppRoleAssignment -ObjectId $sp.ObjectId -All $true

# Remove all users and groups assigned to the application
$assignments | ForEach-Object {
```

Visit here to learn more about permissions and consent grants

FIGURE 1-31 Review permission in service principal for an application.

Chapter summary

- You can route Azure platform logs—including resource logs, activity logs, and Active Directory logs—to Azure Storage, the Azure Event Hub, an Azure Log Analytics work-space, or a partner solution by configuring diagnostic settings.
- Azure provides logging capability at various levels:
  - Application
  - Guest OS
  - Azure resource
  - Azure subscription
  - Azure tenant
- Forwarding logs and metrics to a Log Analytics workspace and metric store, respectively, enables you to use the same set of Azure Monitor tools to visualize and analyze those logs and metrics.
- Azure Monitor provides various tools for visualization and log analysis. These include the following:
  - Activity log
  - Alerts

  - Logs
  - Metrics
  - Insights
- Azure Network Watcher provides tools for visualization and network monitoring.
- Microsoft Defender for Cloud enables you to monitor the security posture of workloads.
- Cost Management enables you to monitor and control Azure costs.
- Microsoft Sentinel is a cloud SIEM and SOAR solution that delivers intelligent security, analytics, and threat intelligence across the enterprise.
- Azure Advisor provides visibility and scores the security, cost, reliability, operational excellence, and performance posture of your Azure subscription.
- Azure role-based access controls provide wide-ranging access controls to Azure resources.
- Azure Active Directory roles enable you to manage objects and perform administrative tasks in AAD.
- AAD is a comprehensive identity-management solution in Azure. It provides solutions for hybrid identity, identity protection, and identity governance.
- Azure provides a hierarchical tree-based organization in which each level is described as a scope.
- An AAD tenant can have only one root management group. Management groups can contain nested management groups or subscriptions. Subscriptions can have resource groups, and a resource group, in turn, can have Azure resources and services.
- Azure Policy defines policies at the management group or subscription level. However, Azure policies can be applied at the management, subscription, and resource group level.
- Azure policies are evaluated for each applicable Azure resource in the scope at which the policy is assigned.

- Azure policies can be grouped into initiatives. An initiative enables you to assign a number of policies as one unit at a particular scope.
- Azure Policy offers built-in policies and initiatives out of the box to enforce compliance of certain regulations, including FedRAMP, PCI, ISO 27001:2013, and many more.
- Applications deployed in an Azure service like Azure Web App can use a managed identity to access other Azure services, like Azure Storage and Azure Key Vault.
- Managed identity can be system-assigned or user-assigned.
- Applications deployed outside Azure can use a service principal to access other Azure services.
- Azure Key Vault securely stores keys, secrets, and certificates, and supports key management, secret management, and certificate management.
- Application developers can register their applications in AAD to integrate them with AAD. Application registration creates a globally unique application object in the AAD tenant where the application was registered.
- An enterprise application is basically the list of service principals.
- Enterprise applications enable you to set up consent for applications to access organization data or services. This consent can be given by a user or an admin.

Thought experiment

Now it is time to validate your skills and knowledge of the concepts you learned in this chapter. You can find answers to this thought experiment in the next section, “Thought experiment answers.”

Suppose you are an Azure solutions architect for a company called Contoso. The company wants to create a landing zone in Azure. You are expected to design the landing zone such that existing applications can be migrated to the cloud and new cloud-native applications can be developed in the cloud. You have had several meetings with Contoso leadership and you have recorded their requirements for the landing zone as follows:

1. The Contoso leadership wants to ensure that the right guardrails are in place for the landing zone to ensure that any deployments in the cloud comply with organizational and regulatory requirements. To achieve this, there will need to be regular monitoring of the cloud to identify noncompliant workloads and remediate them as quickly as possible.
2. The identity team is looking for an identity solution to provide seamless integration with their on-premises Active Directory that requires zero to minimal maintenance. At the same time, user passwords should be authenticated on-premises. Also, users should be able to access applications from their corporate device on the corporate network without being prompted for a password.
3. Administrative access should be provided on a just-in-time basis, and granular RBAC provided for resources in the cloud.
4. Logs should be maintained for auditing for two years.

Thought experiment answers

This section contains the answers to the thought experiment questions.

1. You must use Azure Policy to set up guardrails for the Azure deployments. There are built-in policies and initiatives available to ensure compliance with most common regulations. You can also create custom policies and initiatives if needed. You should apply policies and initiatives at an appropriate scope. To ensure that a minimum set of policies are applied organization-wide, those policies must be applied at the root-management level.
2. AAD can be used as an IDAM in the cloud. You must set up Azure AD Connect on-premises to enable synchronization of users and groups in AAD. You must configure Azure AD Connect with pass-through authentication to ensure that passwords are verified on-premises. Seamless SSO should be configured in AAD.
3. Privileged identity management can provide a solution for just-in-time administrative privilege access. For granular access control on Azure resources, you can leverage Azure RBAC. You should create custom

roles with the correct granular permissions and assign them to the right scope.

4. To retain logs for two years, you can route them to an Azure Log Analytics workspace. You can configure the Azure Log Analytics workspace to retain logs for 30 to 730 days.

Chapter 2

Design data storage solutions

In today’s information era, data is growing rapidly and exponentially. The generation of this vast amount of data opens a door for organizations to use it effectively to make business decisions.

Like a wide variety of IoT devices and social networking sites, database applications generate massive amounts of data. Handling this volume of data with a traditional relational database approach can be challenging and inefficient. The heterogeneity and complexity of the data—also known as *big data*—emitted by numerous connected devices also make it challenging to manage traditional database storage solutions.

Because the AZ-305 exam is an expert-level exam, you must thoroughly understand Microsoft’s data storage services, use your architectural thinking, and design a precise data storage solution. In this chapter, you will learn the critical concepts of designing data storage solutions and data integration on the Microsoft Azure cloud platform.

Skills covered in this chapter:

- Skill 2.1: Design a data storage solution for relational data
- Skill 2.2: Design data integration
- Skill 2.3: Recommend a data storage solution
- Skill 2.4: Design a data storage solution for nonrelational data

Skill 2.1: Design a data storage solution for relational data

A database is the foundation of any application. An accurate database design provides consistent data, high performance, scalability, less management, and, ultimately, user satisfaction. A modern database must address new challenges, such as massive amounts of data, diverse data sources, multiregion deployment, and so on. The Azure cloud platform helps overcome these challenges by providing sets of Azure database services.

In this skill, you will examine the solutions for relational databases' service tiers, scalability, and encryption in Azure.

This section covers how to:

- Recommend database service tier sizing
- Recommend a solution for database scalability
- Recommend a solution for encrypting data at rest, data in transmission, and data in use

Recommend database service tier sizing

The selection of service tiers for the Azure platform's database depends on the database type and whether it is a single database, an elastic pool, or a managed instance. Also, in a single instance or an elastic pool, the selection of service tiers depends on the purchasing model—virtual core (vCore)–based or database transaction unit (DTU)–based. Let's start with database types.

Following are the database service tiers based on the purchasing model:

- **DTU-based purchasing model:**
  - Basic
  - Standard
  - Premium
- **vCore-based purchasing model:**
  - General purpose

- Business critical
- Hyperscale

DTU-based purchasing model

Let's look at the DTU-based purchasing model. DTU stands for database transaction unit, and it blends CPU, memory, and I/O usage. The more DTUs, the more powerful the database. This option is suitable for customers who would like to use a simple preconfigured resource bundle.

When migrating a database from on-premises to Azure, you can get the current CPU, disk read/write, log bytes, and flushed/sec information from the current on-premises server and calculate the required DTU value on the target Azure SQL Database.

Table 2-1 lists the characteristics of DTU-based service tiers.

TABLE 2-1 DTU-based service tiers

	Basic	**Standard**	**Premium**
MAXIMUM STORAGE SIZE	2 GB	1 TB	4 TB
CPU	Low	Low, medium, high	Medium, high
MAXIMUM DTUs	5	3,000	4,000
I/O THROUGHPUT	1–5 IOPS per DTU	1–5 IOPS per DTU	25 IOPS per DTU
UPTIME SLA	99.99 percent	99.99 percent	99.99 percent
I/O LATENCY	Read: 5 ms Write: 10 ms	Read: 5 ms Write: 10 ms	Read/write: 2 ms
MAXIMUM BACKUP RETENTION	7 days	35 days	35 days
COLUMNSTORE INDEXING	N/A	S3 and above	Supported
IN-MEMORY OLTP	N/A	N/A	Supported
ACTIVE GEO-REPLICATION	Yes	Yes	Yes
READ SCALE-OUT	No	No	Yes

vCore-based purchasing model

In the vCore purchasing model, you have the flexibility to independently pick compute, memory, and storage based on your workload needs. So with this flexibility, you can easily map the on-premises database’s vCore, memory, and storage, and choose the matching Azure database tier.

The vCore-based purchasing model offers Azure Hybrid Benefit (AHB), which allows you to use existing licenses for a discounted rate on Azure SQL Database and Azure SQL Managed Instance. AHB enables you to save 30 percent or more on your SQL Database and SQL Managed Instance by using your existing SQL Server licenses with Software Assurance.

Table 2-2 lists the characteristics of vCore-based service tiers.

More Info **AHB Calculator**

For more details, see the AHB calculator at *https://azure.microsoft.com/en-us/pricing/hybrid-benefit/*.

TABLE 2-2 vCore-based service tiers

	Database	**General Purpose**	**Business Critical**	**Hyperscale**
DATABASE SIZE	SQL Database	5 GB–4 TB	5 GB–4 TB	Up to 100 TB
	SQL Managed Instance	32 GB–8 TB	32 GB–4 TB	N/A
COMPUTE SIZE	SQL Database	1 to 80 vCores	1 to 80 vCores	1 to 80 vCores
	SQL Managed Instanc	4, 8, 16, 24, 32, 40, 64, and 80 vCores	4, 8, 16, 24, 32, 40, 64, and 80 vCores	N/A

	e			
AVAILABILITY	All	99.99 percent	99.99 percent; 99.995 percent with zone redundant single database	99.95 percent with one secondary replica; 99.99 percent with more replicas
STORAGE TYPE	All	Premium remote storage (per instance)	Super-fast local SSD storage (per instance)	De-coupled storage with local SSD cache (per instance)
BACKUP	All	RA-GRS, 7–35 days (7 days by default)	RA-GRS, 7–35 days (7 days by default)	RA-GRS, 7 days, constant time, point-in-time recovery (PITR)
IN-MEMORY OLTP	All	N/A	Available	N/A
READ SCALE-OUT	All	No	Yes	No

Recommend a solution for database scalability

One of the objectives of moving an application to the cloud is to support a growing load. An application should be able to increase resources (compute, storage, and so on) to sustain the on-demand load and decrease resources when demand goes down. This flexibility is called *elastic scaling*. With elastic scaling, you can use optimal resources and pay only for what you use.

Following are two methods of scaling:

- **Vertical scaling** With this method, the capacity of the same resource is changed to meet the requirement. For example, you can increase (scale up) VM size from Standard_D2_v2 to Standard_D3_v2 and similarly decrease (scale down) VM size from Standard_D3_v2 to Standard_D2_v2. When you change the size of the same VM, a restart is required, which means the application deployed on the VM is unavailable until the VM restarts and comes back online. Therefore, this method is generally not automated. This method is also called *scale-up and scale-down*.

- **Horizontal scaling** In this method, capacity is increased or decreased by adding or removing instances of resources. For example, you can add one more VM to the load balancer set to meet the increasing load on the application. Similarly, you can remove an existing VM from the load balancer set when there is less load on the application. During this scaling, the application does not become unavailable or experience down-time. Therefore, this is the preferred method for autoscaling. All Azure services that support autoscaling are based on this method only.

Autoscaling is a feature of Azure services that automatically adds or removes resources based on the actual load on the services. Autoscaling eliminates the overhead of the operation team to monitor utilization and adjust resources.

The following sections examine the options available to scale SQL databases.

Azure SQL Database Serverless

Serverless is a vertical scaling option that has been introduced as a new compute tier. This tier automatically scales up or scales down the database's compute based on the actual load. You can specify the minimum and maximum vCore range that the database can use. Memory and I/O limits are proportional to the specified vCore range. The cost of the Serverless tier is the sum of compute and storage cost. The compute cost is calculated based on the number of vCores used per second. The Serverless tier is available under the General Purpose tier in the vCore purchasing model.

Another exciting feature of the Serverless tier is autopause. When the database is inactive, the Serverless compute tier pauses the database automatically, and it resumes the database when activity returns. There is no compute cost when the database is in the paused state, but you do pay for storage costs.

Autopause delay is the time duration for which the database must be in an inactive state before it is automatically paused. The minimum autopause delay is one hour. Figure 2-1 depicts the minimum and maximum vCore configuration, actual CPU utilization, autopause delay period, and autopause.

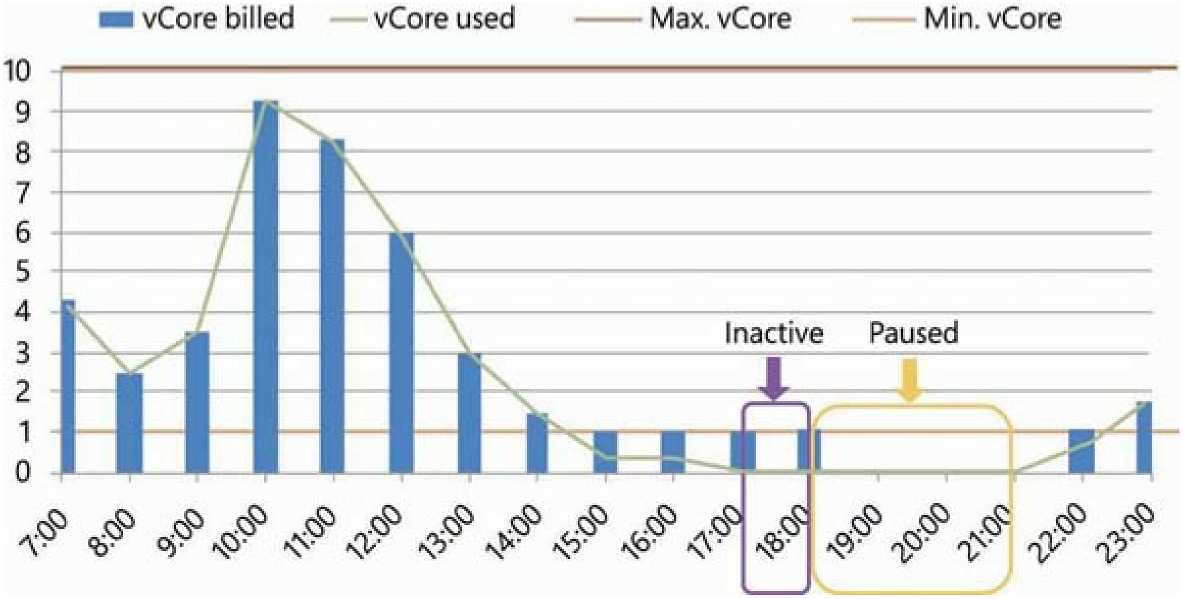


FIGURE 2-1 Serverless database configuration and its vCore utilization

In this example, between 7:00 to 14:00 hours, the number of vCores used is more than 1. During this period, vCores used and vCores billed are the same. From 15:00 to 18:00 hours, the vCore used is below 1. However, even though it is below 1 vCore, it will be billed as 1 vCore because that is the minimum vCore configuration. From 17:00 to 18:00 hours, vCore utilization is 0 because of database inactivity. The Azure SQL Database Serverless tier monitors this for one hour, which is called autopause delay. After one hour, the database is paused at 19:00 hours. At 21:00 hours, SQL Database resumes responding to activity.

Following are scenarios in which you would use SQL Database Serverless:

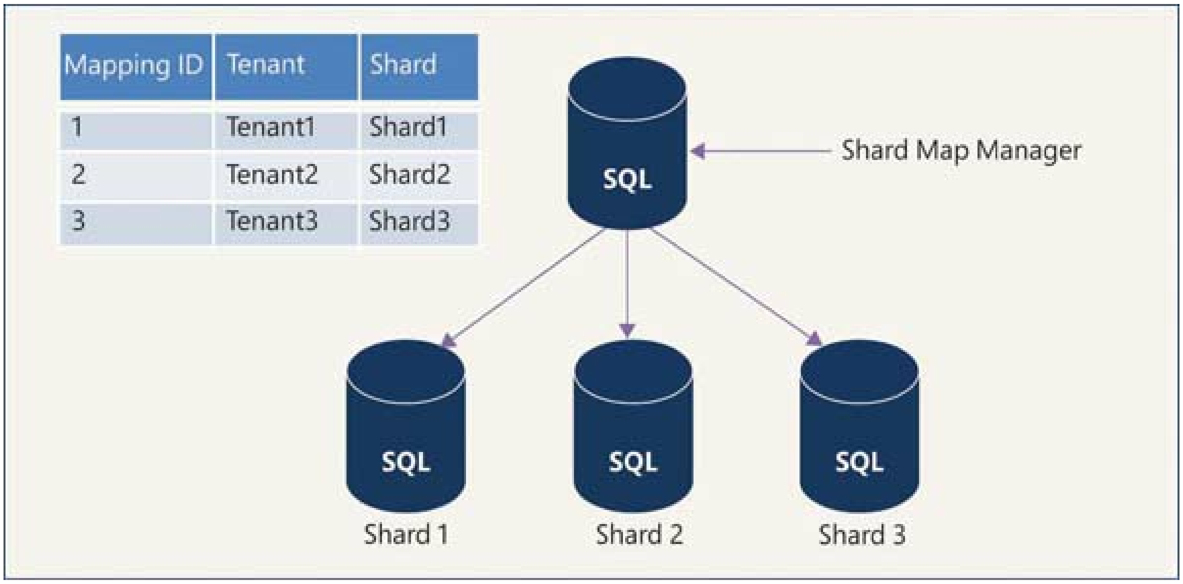
- A new single database (either migrated from on-premises or freshly deployed on Azure) in which vCore and memory requirements are unknown
- A single database with an unpredictable usage pattern, with an inactive period and below-average vCore utilization

Sharding

Sharding is an architecture pattern in which a large set of data is distributed into multiple identically structured databases deployed into separate compute nodes called *shards*. Data is distributed into shards based on a list of values or ranges of values called *sharding keys*. This metadata information (mapping) about data distribution is stored in a separate database called a *shard map manager*.

List-based mapping is called *list mapping*, whereas range-based mapping is called *range mapping*. The shard map manager database is used by the application to identify the correct database (shard) using the sharding key to perform database operations.

This sharding method is most suitable for software as a service (SaaS) applications. SaaS application developers created sharding patterns to support a large volume of data and a large user base. Customers of the SaaS application are referred to as *tenants*. If all the data pertaining to one customer is stored in a single database, then it is called a *single-tenant model*. For this model, the shard map manager stores the global mapping information using a list of tenant IDs. This mapping is called *list mapping*. Figure 2-2 depicts the single-tenant model.



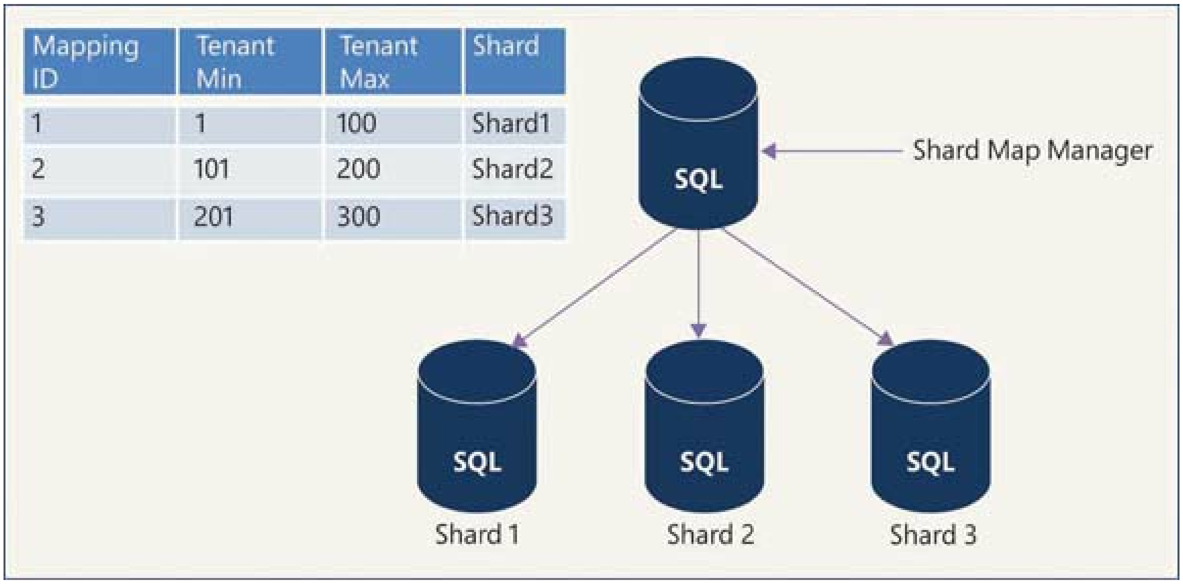
Mapping ID	Tenant	Shard
1	Tenant1	Shard1
2	Tenant2	Shard2
3	Tenant3	Shard3

FIGURE 2-2 Single-tenant model

When the application needs a small amount of data for one tenant, then data from multiple tenants is stored in one database using a *multitenant model*. This model uses range mapping in which the shard map manager keeps the mapping between ranges of the tenant ID and the shard. Figure 2-3 shows the multitenant model.

The Elastic Database tools are a set of libraries and tools that create and manage shards:

- **Elastic database client library** This is a .NET and Java library that is used to create and maintain sharded databases.
- **Elastic database split-merge tool** This tool is useful for moving data between sharded databases.
- **Elastic database jobs** This tool is used for schema changes, credential management, reference data updates, and telemetry collection.
- **Elastic database query** This tool allows you to run a transact-SQL query that spans multiple databases.
- **Elastic transactions** This tool allows you to run transactions that span multiple databases.



Mapping ID	Tenant Min	Tenant Max	Shard
1	1	100	Shard1
2	101	200	Shard2
3	201	300	Shard3

FIGURE 2-3 Multitenant model

Following are some scenarios in which you would use sharding:

- You need to store customers' data in different geographies for geopolitical, performance, or compliance reasons.
- The volume of data is enormous and cannot fit into a single database.
- The transaction throughput requirement is very high and cannot be accommodated by a single database.
- Certain customers' data must be isolated from other customers' data.

Sharding provides high availability, more bandwidth, more throughput, and faster query response and processing. It also helps to mitigate the outage impact in the following scenarios:

- databases are stored in different geographies and one of the locations is experiencing an outage.
- All databases are stored in a single region and one of the databases is experiencing an issue/outage.

In the preceding scenarios, only one customer (tenant) will be affected if you have chosen the single-tenant model, and only a few customers will be affected if you have chosen a multitenant model. Thus, the application's overall impact will be less than the non-sharding application, in which the whole application will crash.

While sharding offers many benefits, it also adds complexity, with creating and managing shards and moving data between shards. You must carefully design your sharding architecture and choose the right sharding keys, which are discussed in the following sections.

Read Scale-Out

There might be some scenarios in which the latest data is not immediately available in the read-only replica because of latency issues. You must consider this small latency when selecting read-only replicas for your application. You can use `sys.dm_database_replica_states` dynamic management views (DMVs) to monitor the replication status and synchronization statistics. When the client/application tries to connect to the database, the gateway internally checks connections strings for the `ApplicationIntent` parameter. If the value of the parameter is `ReadOnly`, then it routes the request to a read-only replica. If the value of the parameter is `ReadWrite`, then it routes that request to a read-write replica. `ReadWrite` is the default value of the `ApplicationIntent` parameter.

Following are some scenarios when you would use read scale-out:

- An analytics workload that only reads data for analysis purposes
- A reporting application that only reads data and generates various reports
- An integration system that only reads data

Elastic Pool

An elastic pool is a collection of databases deployed on a single server that shares resources allocated to the pool. The capacity of the pool is fixed and does not change automatically. So within a fixed capacity of the pool, databases scale automatically within a minimum and maximum capacity defined by the Per Database setting on the Configure blade of the elastic pool settings in Azure Portal.

The elastic pool can use either DTU-based or vCore-based models. In a DTU-based model, databases can scale between a minimum and maximum DTU that is specified by the Per Database setting. Similarly, in a vCore-based model, a database can scale between a minimum and maximum vCore that is specified by the Per Database setting.

The size of the elastic pool can be changed with minimal downtime. A database can be added or removed from an elastic pool. The cost of the elastic pool depends on the size of the pool and not on the individual databases allocated in the pool. So more databases in the pool means more cost savings.

Following are some scenarios in which to use an elastic pool:

- For an application or a group of applications with a large number of databases having low utilization and few, infrequent spikes
- For a SaaS application that requires multiple databases with low to medium utilization

Table 2-3 provides a quick comparison of scaling methods.

TABLE 2-3 Scaling methods

	Azure SQL Data- base Serverless	Sharding	Read Scale-Out	Elastic Pool
SCALING METHOD	Vertical	Horizontal	Horizontal	Vertical
AUTOSCALING	Yes	No	No	Autoscaling within the minimum and maximum defined settings
EASE OF IMPLEMENTATION	Yes	No	Yes	Yes
MANAGEABILITY	Fully managed	Customer managed	Fully managed	Fully managed
AUTOPAUSE TO SAVE COMPUTE COST	Yes	No	No	No
READ-ONLY VERSUS READ-WRITE REPLICA	Read-write	Read-write	Read-only	Read-write

Recommend a solution for encrypting data at rest, data in transmission, and data in use

Encryption is the process of scrambling or encoding data so that only authorized users can decrypt and read that data. Encryption is required when data is stored, in motion, or in use. Effective encryption is the key to securing an organization’s confidential data at rest, transit, and use. Encryption adds an additional layer of data protection. Even if unauthorized users gain access to encrypted data storage, they can’t read data from that encrypted storage. In this skill, you learn how to protect the data storage on Azure platforms using encryption for data at rest, in transit, and in use.

Symmetric and asymmetric key encryption

There are two main types of encryption:

- **Symmetric** With symmetric encryption, the same key is used to encrypt and decrypt data.
- **Asymmetric** This encryption uses two keys—a public key and a private key. The public key is used to encrypt data, which is shared with everyone, whereas the private key is used to decrypt data, and is kept securely and shared with only intended users.

Encrypting data at rest

Encryption at rest is the data protection method for data stored in persistent storage on physical media. Microsoft uses symmetric key encryption for data at rest. Encryption at rest is mandatory for an organization to be compliant with HIPAA, PCI, and FedRAMP standards.

Microsoft uses key hierarchy models for implementing data at rest. It has two types of keys:

- **Data encryption key (DEK)** This key is used to encrypt and decrypt actual data.
- **Key encryption key (KEK)** This key is used to encrypt the data encryption key.

These keys must be secured. It is recommended that you store them in Azure Key Vault. You can use Azure Active Directory to manage and control access to the keys stored in Azure Key Vault. Encryption can be done at the client side or server side, based on your needs.

The encryption models shown in the following sections provide more details about the implementation of encryption:

- Client-side encryption model
- Server-side encryption model

Client-side encryption model

In this model, encryption is done at the client side before storing data in the Azure services. You must handle the encryption, decryption, and key management (such as key rotation) in the client application.

Server-side encryption model

In this model, encryption and decryption are performed by the Azure service, and you or Microsoft can manage the encryption keys. The server-side encryption model is classified into the following three types:

- **Using service-managed keys** The Azure service performs encryption, decryption, and key management.
- **Using customer-managed keys in Azure Key Vault** You must manage keys using Azure Key Vault. The Azure service performs encryption and decryption using the Key Vault.
- **Using customer-managed keys on customer-controlled hardware** The Azure service performs encryption and decryption, and you must manage keys using your hardware.

Microsoft's Azure platform supports encryption at rest for platform as a service (PaaS), infrastructure as a service (IaaS), and software as a service (SaaS).

Encrypting data in transmission

Encrypting data in transmission is the data protection method for data that is actively moving from one component to another. It could be moving across the internet or through a private network.

Microsoft offers the following features for encrypting data in transmission:

- **Transport Layer Security (TLS)** TLS is a cryptographic protocol that provides data integrity, privacy, and authentication during communication between two components over a network. Microsoft protects data using TLS when data is traveling between cloud services and client systems.
- **Azure App Services** With Azure App Services, you can enforce an encrypted connection by setting the HTTPS value to ON. Once enabled, any HTTP connection to your Azure App Service is redirected to an HTTPS URL.
- **Azure SQL Database and SQL Managed Instance** Both the Azure

SQL Database and SQL Managed Instance features always enforce an SSL/TLS connection, irrespective of the `encrypt` or `TrustServerCertificate` setting in the connection string.

- **Azure Storage** Azure Storage supports both HTTP and HTTPS protocols. You can enforce HTTPS by enabling the Secure Transfer Required property. When you do, any call to Azure Storage using the HTTP protocol is rejected. Similarly, any SMB connection without encryption to the Azure file share will also be rejected. By default, this property is enabled when you provision a new storage account.
- **Azure virtual machine** The remote desktop protocol (RDP) connection to Azure VMs uses TLS to protect data in transit. Also, data in transit is encrypted when you connect to a Linux VM using the Secure Shell (SSH) protocol.
- **VPN connection** A site-to-site VPN connection uses IPsec, while a point-to-site VPN connection uses the secure socket tunneling (SSTP) protocol to encrypt the communication.
- **Data-link layer encryption** Microsoft applies the IEEE 802.1AE MAC security standard for data in transit between datacenters. This encryption method is also known as MACsec. This encryption is enabled for all the traffic within a region or between regions.

Encrypting data in use

Data in use describes data that is actively being used by the user or system for processing. This data is stored in nonpersistent storage such as RAM.

Always Encrypted is a client-side encryption technique that protects sensitive data, such as Social Security numbers (SSN), credit card numbers, and personally identifiable information (PII) stored in SQL Server databases and SQL Azure databases. A database driver inside the client application encrypts data before storing it in the database, and it decrypts encrypted data retrieved from the database.

Because encryption is happening at the client side, the keys used to encrypt data are never revealed to the database. Thus, by using this feature, even a database administrator or cloud database operator who manages the database server and who has full control of the database cannot see original decrypted data.

The Always Encrypted feature uses the following keys:

- **Column encryption keys** These keys are used to encrypt data before storing it in the database.
- **Column master keys** These keys are encrypted by using column master keys.

Column encryption keys are stored in the database in encrypted form, and column master keys are stored outside the database—for example, in a local key management system or Azure Key Vault.

This feature encrypts data at rest, in transit, and in use. Hence, it is called Always Encrypted. However, transparent data encryption (TDE) is the recommended option for encrypting data at rest.

Skill 2.2: Design data integration

In the current information age, large amounts of data are generated by many applications, and the amount of data being generated is growing exponentially. An organization must collect data from multiple sources, such as business partners, suppliers, vendors, manufacturers, customers, social media, and so on. This exploding volume of data, disparate data sources, and cloud adoption are crucial factors for organizations that need to redesign or adopt a new data integration solution to meet business needs. In this skill, you look at various options available in the Microsoft Azure cloud platform for data integration and data analysis.

This section covers how to:

- Recommend a solution for data integration
- Recommend a solution for data analysis

Recommend a solution for data integration

Microsoft's Azure Data Factory is a solution for today's data integration needs. Let's look at Azure Data Factory and its capabilities.

Azure Data Factory (ADF) is a cloud-based, fully managed, serverless, and cost-effective data integration and data transformation service that allows you to create data-driven work-flows and to orchestrate, move, and transform data. It is designed for complex hybrid extract, transform, load (ETL) and extract, load, transform (ELT) patterns.

ADF does not store data; it ingests data from various sources, transforms it, and publishes it to data stores called *sinks*. You can also run SQL server integration services (SSIS) packages in Azure Data Factory, which provides assistance in migrating existing SSIS packages.

An Azure Data Factory pipeline can be created by using these tools or APIs:

- Azure Portal
- Visual Studio
- PowerShell
- .NET API
- REST API
- Azure Resource Manager template

Azure Data Factory supports the following file formats:

- Avro
- Binary
- Common Data Model
- Delimited text
- Delta
- Excel
- JSON
- ORC
- Parquet
- XML

Let’s look at the Azure Data Factory components before delving into how ADF works:

- **Linked services (connectors)** Linked services contain configuration settings required for ADF to connect various external resources outside ADF. This information can include a server name, database name, credentials, and the like. This is similar to the connection string used to connect to the SQL Server database. Depending on an external resource, it can represent data stores—such as SQL Server, Oracle, and so on—or compute resources such as HDInsight to perform the execution of an activity. For example, an Azure Storage–linked service represents a connection string to connect to the Azure Storage account.
- **Dataset** This component represents structures of data within data stores and provides more granular information about the data from linked sources you will use. For example, an Azure Storage–linked service represents a connection string to connect to the Azure Storage account, and the Azure Blob dataset represents the blob container, the folder and path, and the blob’s file name.
- **Activities** This component represents the action taken on the data. A pipeline can contain one or more activities. Azure Data Factory currently provides three types of activities: data-movement activities, control activities, and data transformation activities.
- **Pipeline** A pipeline is a logical grouping of activities that perform a task together.
- **Triggers** This component is a unit of processing that decides when to commence a pipeline execution. Azure Data Factory supports the following three types of triggers:
  - **Schedule trigger** This invokes a pipeline on a scheduled time.
  - **Tumbling window trigger** This invokes a pipeline at an aperiodic interval, while retaining its state.
  - **Event-based trigger** This invokes a pipeline to respond to an event.
- **Integration runtime (IR)** This component is a compute infrastructure used by ADF to carry out integration activities such as data movement, data flow, activity dispatch, and SSIS package

execution. There are three types of integration runtimes:

- **Azure IR**   This is a fully managed, serverless compute used to perform data flow, data movement, and activity dispatch on a public and private network.
- **Self-hosted IR**   You can install a self-hosted IR inside on-premises networks secured by the Azure Storage Firewall or inside a virtual network. It makes only out-bound HTTPS calls to the internet. Currently, it is supported only on Windows.
- **Azure-SSIS IR**   This is used to natively execute SSIS packages.

Figure 2-4 shows how Azure Data Factory works.

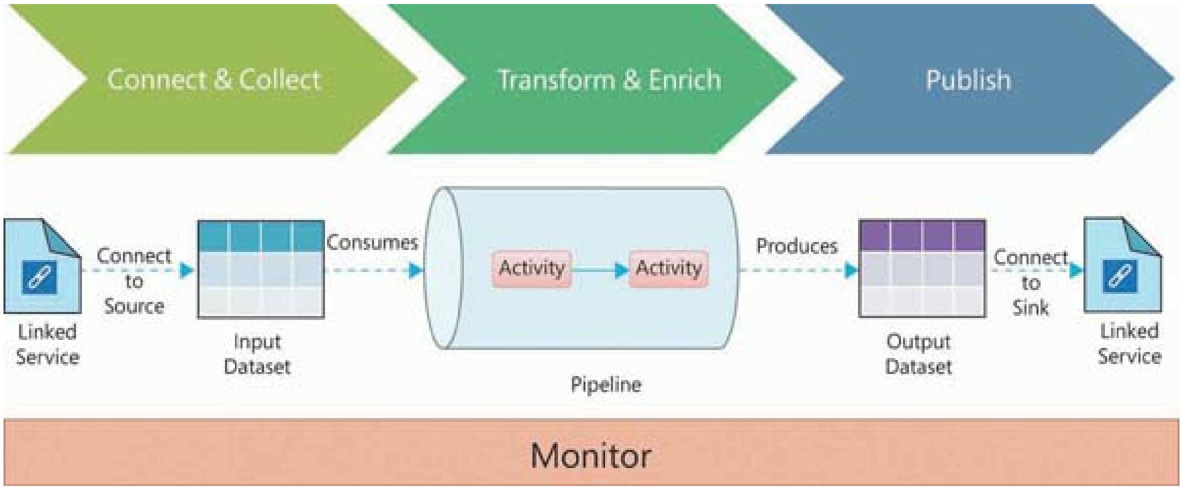


FIGURE 2-4   Azure Data Factory

The pipeline in Azure Data Factory is executed based on a schedule (for example, hourly, daily, or weekly) or is triggered by an external event. In the execution, the pipeline performs the following steps (refer to Figure 2-4):

1. **Connect and collect**   Connect to the source system and collect the required data, as mentioned in the source-linked service and input dataset. You can connect to various source systems in Azure, on-premises, and in SaaS services. These systems can be used as a source, sink, or both, depending on the type of activity.

More Info **Azure Data Factory Connectors**

For supported data stores, see *https://docs.microsoft.com/en-us/azure/data-factory/connector-overview*.

2. **Transform and enrich** After data is collected, it is transformed and enriched using the data flow activity that is executed on Spark internally without any knowledge of the Spark cluster and its programming. If you would like to code the transformation, you can use external activities for the execution of transformation on compute services such as Data Lake Analytics, HDInsight, Spark, and machine learning.
3. **Publish** After data is transformed or enriched, it can be published to target systems such as Azure SQL Database, Azure Cosmos DB, and so on.

Azure Data Factory provides the **Monitor & Manage** tile on the **Data Factory** blade, where you can monitor pipeline runs. You can also monitor the pipeline programmatically using SDK (.NET and Python), REST API, and PowerShell. The **Azure Monitor** and **Health** panels in the Azure Portal are additional ways to monitor the pipeline. You can view active pipeline executions as well as the executions history.

Azure Data Factory is useful when you need to ingest data from a multicloud and on-premises environment. ADF is a highly scalable service to handle gigabytes and petabytes of data.

Recommend a solution for data analysis

Once data is available in a data store, the next step is data analysis. Microsoft Azure offers following services for data analysis:

- Azure Databricks
- Azure Data Lake

Azure Databricks

Azure Databricks is a fully managed, fast, and easy analytics platform that is based on Apache Spark on Azure. It provides flexibility for one-click setup and offers streamlined workflows and shared collaborative and interactive workspaces. These workspaces enable data science teams consisting of data engineers, data scientists, and business analysts to collaborate and build data products.

Azure Databricks is natively integrated with Azure services such as Blob Storage, Azure Data Lake Storage, Cosmos DB, Azure Synapse Analytics, and the like. It supports popular BI tools, such as Alteryx, Looker, Power BI, Tableau, and so on, to connect Azure Databricks clusters to query data.

Azure Databricks supports the following sources, either directly in the Databricks runtime or by using small shell commands to enable access:

- Avro files
- Binary files
- CSV files
- Hive tables
- Image files
- JSON files
- LZO compressed files
- MLflow experiment files
- Parquet files
- Zip files
- XML files

Let’s look at key components of Azure Databricks:

- **Databricks workspace** The workspace is an environment for accessing all Azure Databricks assets. A workspace folder contains:
  - **Notebook** A web-based user interface to document runnable code, narrative text, and visualizations.
  - **Dashboard** A user interface that provides organized access to visualizations.

    - **Library** A collection of code available to the notebook or to jobs running on a cluster. Databricks provides many ready-made libraries, and you can add your own.
    - **Experiment** A collection of MLflow runs for training a machine learning model.
- **Data management** The following objects hold data and are used to perform analytics as well as feed into the machine learning algorithm:
    - **Databricks File System (DBFS)** This is a file system abstraction layer over a blob store.
    - **Database** This is a systematic collection of information that can be easily accessed, managed, and updated.
    - **Table** This is structured data that can be queried using Apache Spark SQL and Apache Spark APIs.
    - **Metastore** This stores structured information from various tables and partitions.
- **Compute management** Following are the components that you must know to run a computation in Azure Databricks:
    - **Cluster** This is a computing resource to run notebooks and jobs. There are two types of clusters: all-purpose clusters and job clusters. An all-purpose cluster is created manually using UI, REST API, or CLI. A job cluster is created by Databricks when you trigger a job.
    - **Pool** This is a collection of ready-to-use idle instances that reduce cluster start and autoscaling times.
    - **Databricks runtime** This is a set of core components that run on the cluster.
    - **Job** This is an execution of a notebook or JAR at a scheduled time or on demand.

You can easily integrate and read data from Azure services such as Azure Blob Storage, Azure Data Lake Storage, Azure Synapse Analytics (formerly Azure SQL Data Warehouse), and so on. You can also connect to Kafka, Event Hub, or IoT Hub and stream millions of events per second to Azure Databricks. You can integrate with Azure Key Vault to store and manage secrets such as keys, tokens, and passwords. Azure Databricks integrates closely with Power BI for interactive visualization. You can create Build and Release pipeline for Azure Databricks with Azure DevOps for continuous integration (CI) and continuous deployment (CD).

The Azure Databricks runtime is a set of components that run on the Databricks cluster. Azure Databricks offers several runtime variants, such as runtime for ML, runtime for Genomics, and the like. These versions are updated and released regularly to improve the usability, performance, and security of big data analytics. It also offers a serverless option that helps data scientists iterate quickly.

Azure Databricks easily integrates with Azure Active Directory and provides role-based access control (RBAC) and fine-grained user permissions for notebooks, jobs, clusters, and data.

Azure Data Lake

Azure Data Lake is a fully managed, highly scalable data lake service on the Azure cloud platform. It provides an enormous amount of storage to store structured, semi-structured, and unstructured data and perform analytics to gain business insights quickly. Figure 2-5 shows that the Azure Data Lake platform primarily consists of Azure Data Lake Analytics, Azure Data Lake Store, and Azure HDInsight.

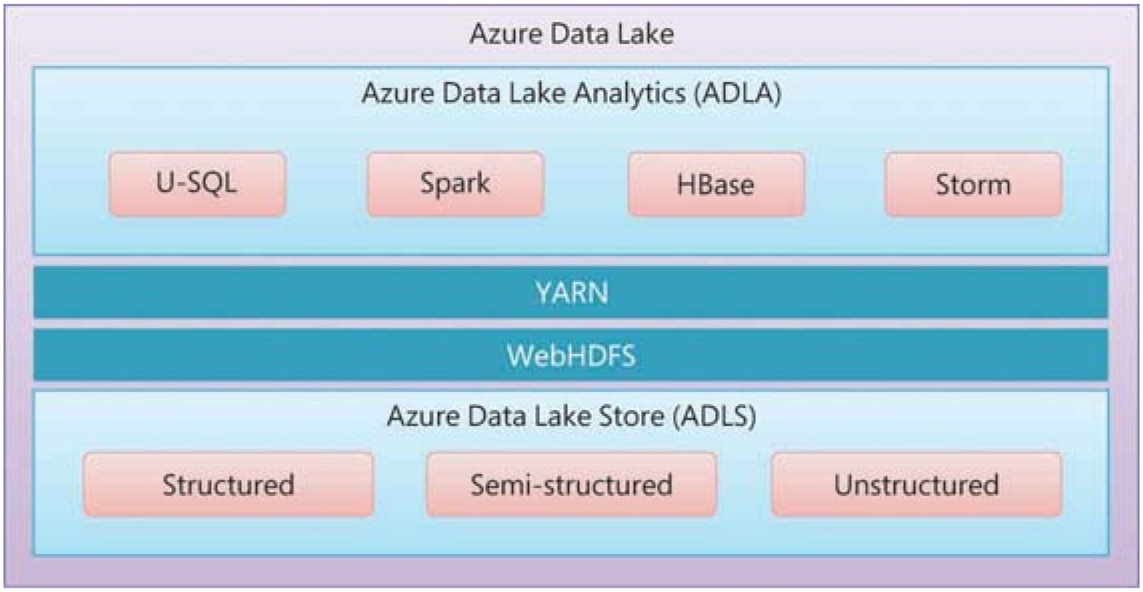


FIGURE 2-5 Azure Data Lake

Azure Data Lake includes three services:

- Azure Data Lake Storage
- Azure Data Lake Analytics
- Azure HDInsight

Azure Data Lake Storage (ADLS)

Azure Data Lake Storage (ADLS) is a fully managed, hyper-scale, redundant, and cost-effective data repository solution for big data analytics. This repository provides storage with no limits or restrictions on the file size or the type of data stored (structured, semi-structured, unstructured, or total data volumes). You can store trillions of files, and one file can be petabytes in size if needed. This allows you to run massively parallel analytics.

ADLS easily integrates with Azure services such as Azure Databricks and Azure Data Factory. To protect data, it uses Azure Active Directory for authentication and RBAC, and it uses Azure Storage Firewall to restrict access and encryption of data at rest.

ADLS comes in two variants:

- **ADLS Generation 1** ADLS Gen 1 uses a Hadoop file system that is compatible with Hadoop Distributed File System (HDFS). It also exposes a WebHDFS-compatible REST API that can be easily used by an existing HDInsight service. ADLS Gen 1 is accessible using the new AzureDataLakeFilesystem (adl://) file system. This file system provides performance optimization that is currently not available in WebHDFS. ADLS Gen 1 can be easily integrated with Azure services such as Azure Data Factory, Azure HDInsight, Azure Stream Analytics, Power BI, Azure Event Hubs, and the like.
- **ADLS Generation 2** ADLS Gen 2 is built on Azure Blob Storage. Azure Storage brings its power, such as geo-redundancy; hot, cold, and archive tiers; additional metadata; and regional availability. ADLS Gen 2 combines all the features of Gen 1 with the power of Azure Storage, which greatly enriches performance, management, and security. Gen 2

uses a hierarchical namespace (HNS) to Azure Blob Storage, which allows the collection of objects within an account to be arranged into a hierarchy of directories and subdirectories, like a file system on a desktop computer.

Azure Data Lake Analytics (ADLA)

Azure Data Lake Analytics (ADLA) is a fully managed and on-demand data analytics service for the Azure cloud platform. It is a real-time analytic service built on Apache’s Hadoop Yet Another Resource Negotiator (YARN). It allows the parallel processing of very large volumes of data (structured, semi-structured, and unstructured), which eliminates the need to provision the underlying infrastructure. ADLA easily integrates with ADLS and Azure Storage Blobs, Azure SQL Database, and Azure Synapse Analytics (formerly SQL Data Warehouse).

In ADLA, you can perform data transformation and processing tasks using a program developed in U-SQL, R, Python, and .NET. U-SQL is a new query langugae that blends SQL and C# to process both structured and unstructured data of any size. You can also use Visual Studio as your integrated development environment (IDE) to develop a U-SQL script.

Performing analytics is quite easy with ADLA. As a developer, you simply write a script using U-SQL or your language of choice and submit it as a job.

ADLA pricing is based on Azure Data Lake Analytics Units (ADLAUs), also known as analytics units (AUs). AU is a unit of compute resource (CPU cores and memory) provided to run your job. Currently, an AU is the equivalent of two cores and 6 GB of RAM. A job is executed in four phases: preparation, queuing, execution, and finalization. You must pay for the duration of the job’s execution and finalization phase.

Azure Hdinsight

Azure Data Lake brings integration with the existing Azure HDInsight service. It is a fully managed, open-source Hadoop-based analytics service on the Azure cloud platform. Azure HDInsight uses the Hortonworks Data Platform (HDP) Hadoop distribution. It is designed to process a massive amount of streaming and historical data. It enables you to build big data applications using open-source frameworks such as Apache Hadoop, Apache Spark, Apache Hive, Apache Kafka, and Apache Storm. You can also easily integrate Azure HDInsight with a range of Azure services, such as Azure Cosmos DB, Azure Data Factory, Azure Blob Storage, Azure Event Hubs, and so on.

Azure Synapse Analytics

Azure Synapse Analytics is an evolution of Azure SQL Data Warehouse that brings the SQL data warehouse and big data analytics into a single service. It provides a unified experience to ingest, prepare, manage, and serve data for business intelligence and machine-learning needs.

Azure Synapse Analytics provides end-to-end analytic solutions that combine the power of a data warehouse, Azure Data Lake, and machine learning at an immense scale on the Azure cloud platform.

AZURE SYNAPSE STUDIO As shown in Figure 2-6, Azure Synapse Analytics includes a component called Azure Synapse Studio. This is a web-based interface that provides an end-to-end development experience. Using Azure Synapse Studio, you can interact with various services of Azure Synapse.

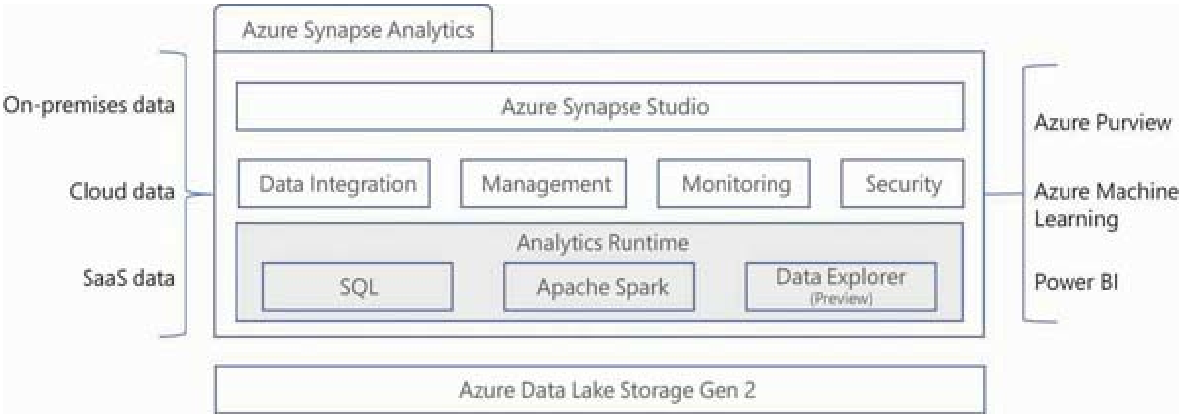


FIGURE 2-6 Azure Synapse Analytics

AZURE SYNAPSE SQL Azure Synapse Analytics also supports the use of Azure Synapse SQL. Azure Synapse SQL uses a node-based architecture that separates compute and storage. This separation enables you to scale compute independently of the data. You can pause the service to free up compute resources. You will be charged only for storage when you pause the service. The data remains intact in storage during this pause period.

Azure Synapse Consumption Models

Azure Synapse Analytics provides two consumption models:

- **Dedicated SQL pool** Dedicated SQL pool (formerly SQL Data Warehouse) is a collection of provisioned analytic resources. You can scale up, scale down, or pause dedicated SQL pools during non-operational hours. The size of the dedicated pool is measured in Data Warehousing Units (DWUs). In dedicated SQL pools, queries are distributed in parallel across computer nodes using a massively parallel processing (MPP) engine. Figure 2-7 illustrates the Azure Synapse dedicated SQL pool architecture.

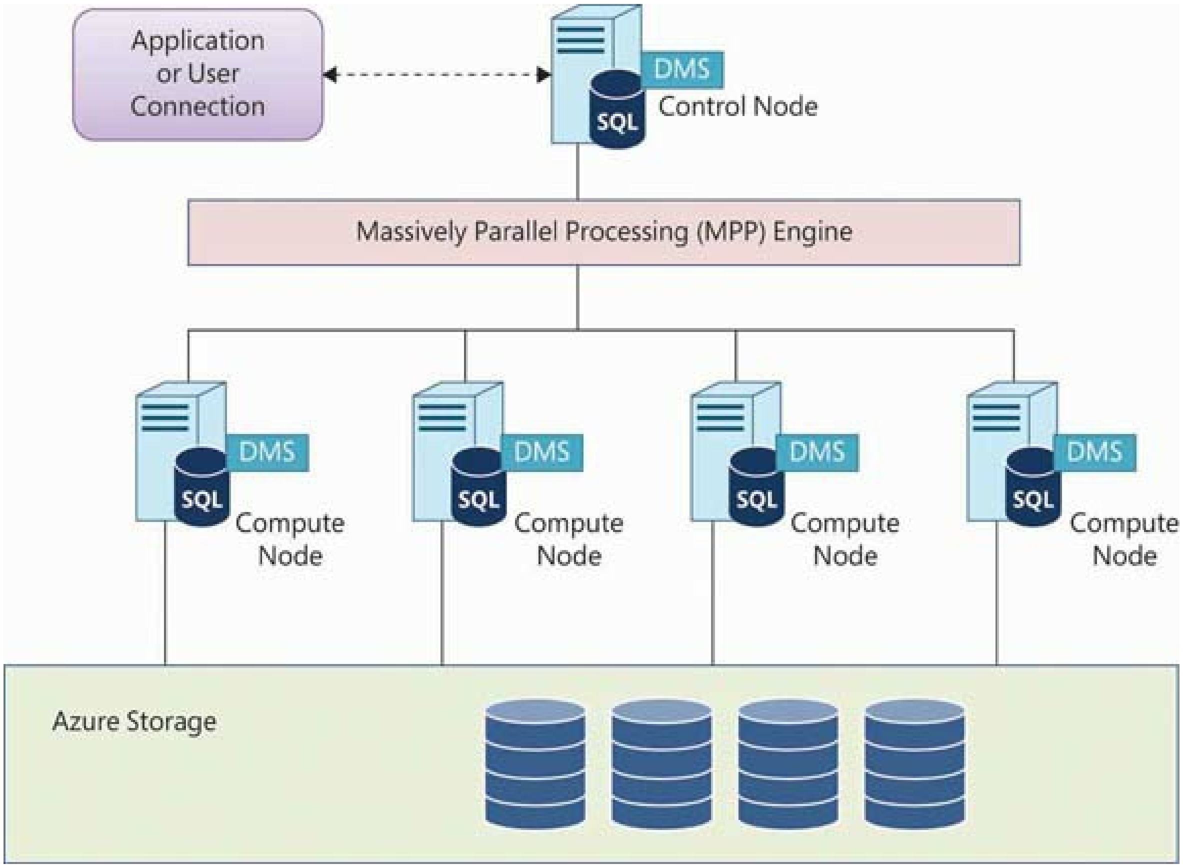


FIGURE 2-7 Azure Synapse dedicated SQL pool

- **Serverless SQL pool** As the name implies, with serverless SQL pools, you need not provision any infrastructure. It is scaled automatically to meet the query resource requirement. Once you provision an Azure Synapse workspace, you get a default serverless SQL pool endpoint. You can start querying data using the serverless SQL pool and will be charged based on the data process by each query run. Figure 2-8 illustrates the Azure Synapse serverless SQL pool architecture.

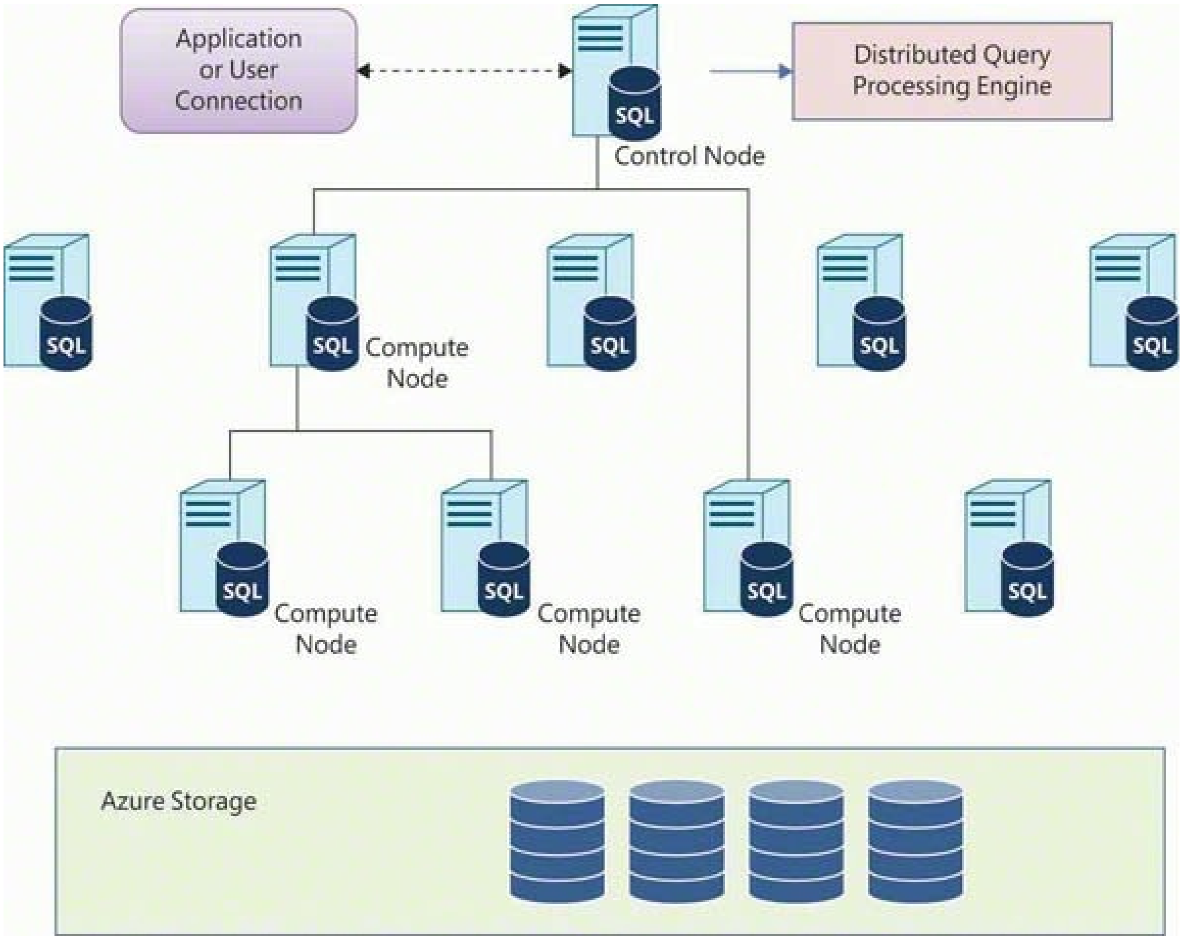


FIGURE 2-8 Azure Synapse serverless SQL pool

As shown in Figure 2-8, an Azure Synapse serverless SQL pool consists of the following components:

- **Control nodes** A user or an application connects to control nodes and gives T-SQL commands to the control node for execution. A control node optimizes queries using the MPP engine and then distributes it to multiple compute nodes to run in parallel.
- **Compute nodes** The Azure Synapse serverless SQL pool distributes processing across multiple compute nodes. It can use a maximum of 60 compute nodes for processing, which is determined by the service level for Azure Synapse SQL. (Again, DWU is the unit of compute power.) All the compute nodes run queries in parallel. The data movement service (DMS) manages data movement across compute nodes to run queries in parallel.
- **Azure Storage** The Azure Synapse serverless SQL pool uses Azure Storage to store data. Data is horizontally partitioned and stored in a shard to optimize the performance of the system. In this sharding process, data is split across 60 distributions. There are three methods of distribution, which determine how rows in the table are split across nodes:
  - **Round robin** This is the default method of distribution. In this method, data is distributed evenly across the nodes. It is quick and straightforward to create, but it is not optimized for query performance.
  - **Replicated** In this method, a complete table is replicated across nodes. This method is suitable for small tables and provides faster query performance.
  - **Hash** In this method, a hash function is used to distribute data. One of the columns in the table is used as a distribution key column. Azure Synapse SQL automatically spreads the rows across all 60 distributions based on distribution key column value.

APACHE SPARK POOL Azure Synapse Analytics also provides a serverless Apache Spark pool, which is a fully managed Microsoft implementation of Apache Spark. An Apache Spark pool uses the Apache Spark core engine, which is a distributed execution engine. An Apache Spark cluster is managed by the YARN, yet another resource negotiator. YARN ensures proper use of the distributed engine to process the Spark queries and jobs.

Apache Spark pools support in-memory cluster computing, which is much faster than disk-based data processing. An Apache Spark pool is compatible with ADLS Gen 2 and Azure Storage, which helps it to process data stored in Azure. Apache Spark pools have multilanguage support for languages like Scala, C#, Spark SQL, Pyspark, Java, and so on.

DATA INTEGRATION Azure Synapse Analytics provides the same data integration engine that is available in Azure Data Factory. Thus, the experience of creating data pipelines is the same as that of Azure Data Factory. This allows for rich data transformation capabilities within Azure Synapse Analytics itself.

Security

Azure Synapse Analytics provides an array of security features:

- Data encryption for data in transit and data at rest
- Support for Azure AD and multifactor authentication
- Object-level, row-level, and column-level security
- Dynamic data masking
- Support for network-level security with virtual networks and Azure Firewall

Skill 2.3: Recommend a data storage solution

In today’s world, data is generated at an unprecedented rate. Organizations are looking for cheaper, faster, and better ways to store, protect, and manage data. Microsoft Azure helps organizations address these challenges with services provided on the Azure cloud platform. In this skill, you will learn how to design data storage solutions for relational, semi-structured, and nonrelational data.

This section covers how to:

- Recommend a solution for storing relational data

- Recommend a solution for storing semi-structured data
- Recommend a solution for storing nonrelational data

Recommend a solution for storing relational data

This section examines the various relational database deployments available in Azure. First, though, you must understand the following vital requirements for selecting a relational data store:

- **Manageability** Are you ready to completely manage your database or do you want to offload manageability to the Microsoft Azure platform?
- **Encryption** What different encryption methods are required for securing your data?
- **Data volume** How much data do you need to store?
- **Ease of migration** How quickly can you migrate databases from on-premises to Azure?
- **Feature set** Does the platform support reporting, such as with SQL Server Reporting Services (SSRS), and analytics, like SQL Server Analysis Services (SSAS)? Does it support extract, transform, load (ETL) operations, such as with SQL Server Integration Services (SSIS)?
- **Database backup service** Do you need to make an explicit backup of the database?
- **Cost** How cost-effective is your database solution?
- **Security** Is the database in a cloud-exposed public endpoint or is it completely deployed in a private network?
- **Scalability** Is the database scalable to support your growing demands? Does it also support horizontal scaling?
- **High availability** Is your database highly available? How much availability will you get?
- **Read-intensive or transactional data** Does the database need to support read-intensive or transactional data?

Now let’s consider the options available to store relational data on the Azure cloud platform. Table 2-4 compares the capabilities of each of the Azure services discussed in the following sections.

TABLE 2-4 Azure relational database services

	Azure SQL Database	**Azure SQL Database Managed Instance**	**SQL Server on VM**
DATABASE SIZE	Up to 4 TB in DTU and vCore model 100 TB in hyperscale	Max 8 TB	SQL Server limit: 524,272 Database files on VM disk: maximum size of the disk supported by the VM Database files on Azure Storage: Azure Storage size limit
SCALABILITY	Vertical	Vertical	Vertical
AVAILABILITY	99.99 percent	99.99 percent	99.99 percent
DATA TYPE	Relational database that supports nonrelational data, such as graphs, JSON, spatial, and XML	Relational database that supports nonrelational data, such as graphs, JSON, spatial, and XML	Relational database that supports nonrelational data, such as graphs, and JSON, spatial, and XML
FEATURES (SSRS, SSIS, SSAS)	No	No	Yes
ENCRYPTION	Transparent data encryption (TDE) Always Encrypted for data in motion, data at rest, and data in use	Transparent data encryption (TDE) Always Encrypted for data in motion, data at rest, and data in use	Transparent data encryption (TDE) Always Encrypted for data in motion, data at rest, and data in use
DISAST	Active-geo replication	Auto-failover group	Always On availability

ER RECOVERY SOLUTION	Auto-failover group		groups Database mirroring Log shipping

Azure SQL Database

Azure SQL Database is a fully managed, scalable, and highly available relational database service on the Azure PaaS. It is a multimodel database that enables you to store relational data, graphs, JSON documents, key–value pairs, and XML data. The maximum database size supported by the Azure SQL Database is 4 TB. Microsoft has introduced a new Hyperscale storage and compute tier, which is highly scalable. This tier supports 100 TB of data.

By default, all new databases deployed in Azure SQL Database are encrypted at rest using transparent data encryption (TDE). (TDE must be manually enabled for Azure SQL databases created before May 2017.) The database, its backup file, and its transaction logs are encrypted and decrypted in real time by the Azure platform without requiring any application changes. Microsoft also offers the Always Encrypted feature to protect data at rest, data in transit, and data in use. This feature uses column granularity to encrypt data inside client applications and does not reveal encryption keys to the database engine. Thus, it provides separation between those who own the data and those who manage it, and it keeps data confidential from administrators and cloud operators.

Following are some scenarios in which you would use Azure SQL Database:

- To store relational data when the database schema is known before actual implementation of the application
- When you need high availability and quick disaster recovery of the database
- When you want to offload management tasks to the Microsoft Azure platform
- For artificial Intelligence (AI)–based database tuning

- When you need a database with Always Encrypted functionality to protect sensitive data

Azure SQL Managed Instance

Azure SQL Managed Instance is a fully managed and scalable database instance, and is nearly 100 percent compatible with the latest SQL Server (Enterprise Edition) database service. Azure SQL Managed Instance supports only the vCore-based purchasing model. You can manually scale from 4 to 80 cores.

Azure SQL Managed Instance is completely isolated and is deployed to VMs in a dedicated subnet. It provides 99.99 percent uptime and ensures that committed data is never lost due to failures. Similarly to Azure SQL Database, Azure SQL Managed Instance supports TDE encryption and the Always Encrypted feature.

> ***Note*** SQL Managed Instance databases created before February 2019 are not encrypted by default, so you would need to manually enable encryption.

Azure SQL Managed Instance eases the migration from an on-premises Azure SQL Server. Azure SQL Server Database backups from on-premises SQL Servers can be restored on the managed instance without the use of any other tools. Azure SQL Managed Instance supports the auto-failover group for data replication. It also provides the following standard features of SQL Server Database Engines:

- SQL Server Agent
- Database Mail
- Native database backup and restore
- Linked servers
- Cross-database transactions
- SQL CLR modules

- Row-level security
- SQL Audit
- Service Broker
- In-memory optimization
- DBCC statement
- SQL Server Analysis Services (SSAS)
- SQL Server Integration Services (SSIS)
- SQL Server Reporting Services (SSRS)

Following are some scenarios in which you would use Azure SQL Managed Instance:

- When you want to easily migrate a database from on-premises to Azure
- When you want to migrate an on-premises database to Azure with minimal downtime
- When an application uses lots of cross-database queries
- When an application requires a scheduled job to be executed inside the database using an SQL agent
- When you want to store relational data and you know the database schema before the actual implementation of the application
- When you need high availability and quick disaster recovery of the database
- When you want to offload management tasks to the Microsoft Azure platform
- When you have a database that requires Always Encrypted functionality to protect sensitive data

SQL Server on Azure Virtual Machines

SQL Server on Azure Virtual Machines (VMs) is Microsoft's SQL Server database engine for Azure VMs. You can vertically scale SQL Server on a VM to the maximum VM size supported by the Azure platform. The maximum size of the database depends on the maximum disk size supported by the Azure VM.

When you host a database on an Azure VM, you become responsible for managing and implementing the high-availability and disaster-recovery features of that database solution, even though Microsoft is typically responsible for the high availability of Azure VMs.

Following are some scenarios in which you would use SQL Server on Azure Virtual Machines:

- When an application requires full compatibility with the SQL Server (Enterprise Edition) database engine
- When an application needs top features of SQL Server, such as SQL Server Reporting Services (SSRS); analytics, such as SQL Server Analytics Services (SSAS); and ETL, such as SQL Server Integration Services (SSIS)
- When you want to migrate an on-premises database to Azure with minimal changes
- When complete isolation is required at the infrastructure level

Recommend a solution for storing semi-structured data

Semi-structured data has some structure but does not properly fit into a relational format consisting of rows, columns, and tables, and does not have fixed schema. This type of data contains tags and elements, which are used to describe how data is stored.

With semi-structured data, you can store any data in any structure without modifying database schema or coding. Semi-structured data is more flexible and simpler to scale than structured data. However, although semi-structured data does provide flexibility, it also adds a few challenges to storing, querying, and indexing data.

A few common formats of semi-structured data are XML, JSON, and YAML. Semi-structured data is also referred to as *nonrelational* or *NoSQL* data. The most recommended storage option for semi-structured data is Azure Cosmos DB, followed by Azure Table Storage. You will look into these services in the next section.

Recommend a solution for storing nonrelational data

A nonrelational database does not use tabular schema or rows and columns like in a relational database. Nonrelational databases are also known as *NoSQL* databases.

Nonrelational data is typically a collection of semi-structured and unstructured data of the following types:

- Key–value
- Document
- Column-family
- Graph
- Time-series
- Object

The following sections examine these types of nonrelational data and the recommended Azure services to store them.

Key–value data store

Key–value data consists of a unique index key and its associated value. It is one of the least complex forms of NoSQL data. This data is stored in a large hash table, which is useful for quickly searching for a value based on an index key. Sometimes this data is also searched using a range of keys. This type of data store is not suitable for querying data based on values, and generally does not have such capabilities. Table 2-5 shows a sample key–value data store:

TABLE 2-5 Sample key–value data store

Key	Value
AA	900-124-1254
AB	512-658-0000
AC	678-254-1245

The following Azure services support this type of data store:

- Azure Table Storage
- Azure Cosmos DB
- Azure Cache for Redis

Document data store

A document data store is a key–value data store in which the values are documents. A “document” in this context is a collection of named fields and values. A document database has flexible schema that can be different for each document.

A document data store uses key–value data to store and access document data. The key is a unique identifier for the document that is used to retrieve the document, and the value is the actual document. This is a complex data structure stored mostly in JSON format, although other formats could include XML, YAML, and so on. Table 2-6 shows a sample document data store.

> ***Note*** The JSON format is the most widely used format for document data stores.

TABLE 2-6 Sample document data store

Key	Value

100	{
	"employeeId": 100,
	"firstName": "Joseph",
	"lastName": "Taylor",
	"emailid": "joseph@contoso.com"
	}

	{

101	"employeeId": 101, "firstName": "John", "lastName": "Taylor", "emailid": "john@contoso.com" }

The following Azure service supports this type of data store:

- Azure Cosmos DB

Column-family data stores

A column-family data store arranges data into columns and rows. In a column-family data store, columns are divided into groups called *column families*. A column can contain null values or data with different data types. This type of store consists of multiple rows, and each row can contain different numbers of columns compared to other rows. A column-family data store is also known as a *columnar data store*, a *wide column store*, and a *column-oriented database*.

Column-family databases use something called a *keyspace*. A keyspace is similar to a schema in a relational database. The keyspace contains all the column families, each column family contains multiple rows, and each row contains multiple columns. Table 2-7 shows a sample employee column-family data store.

TABLE 2-7 Sample column-family data store

Employee ID	Column Family: Employee
1000	First Name: Joseph Last Name: Smith Email: joseph@contoso.com

	Department: HR
1001	First Name: John Last Name: Taylor Age: 45
1002	First Name: Lisa Last Name: Johnson Gender: Female

The following Azure service supports this type of data store:

- Azure Cosmos DB Cassandra API

Graph data stores

A graph data store stores relationships between entities. Graph databases use nodes to store data entities, edges to store relationships between entities, and properties to describe information associated with nodes. In some cases, edges are also described using properties. Edges can also have a direction, which represents the nature of the relationship. Relationships between nodes are not calculated at runtime, but are persisted in the database. This type of data store helps applications and users efficiently perform queries that traverse networks of nodes and edges. Figure 2-9 shows a sample graph data store.

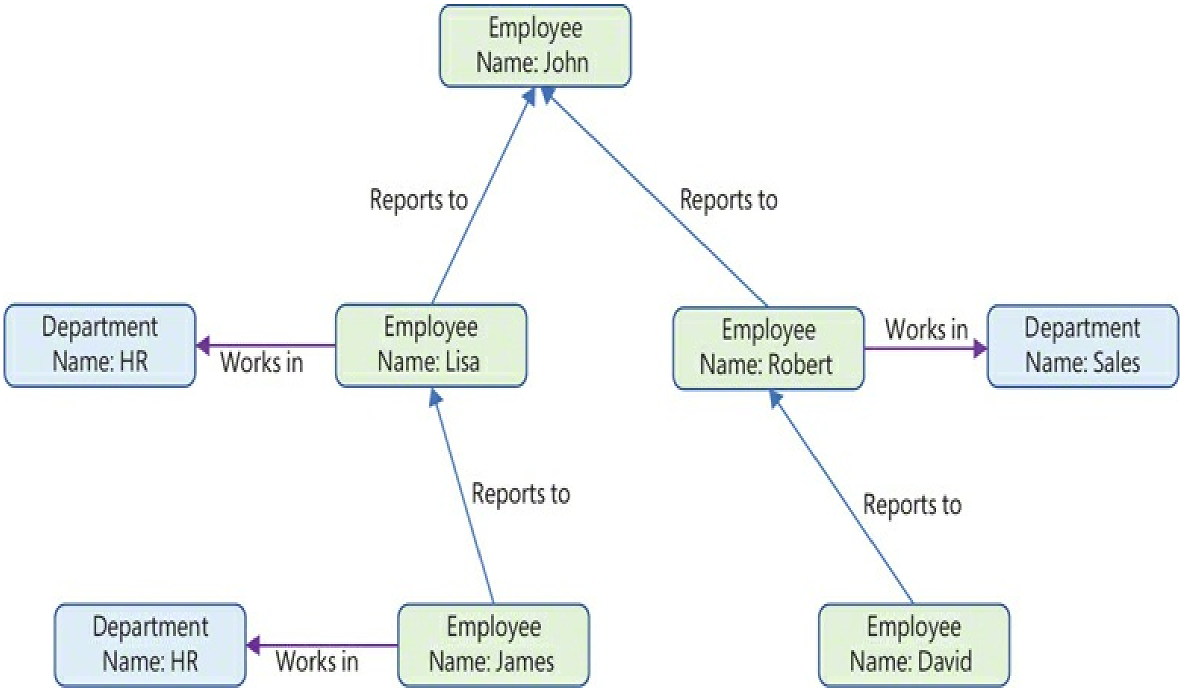


FIGURE 2-9 Sample graph data store

The following Azure service supports this type of data store:

- Azure Cosmos DB Graph API

Time-series data stores

Time-series data stores are designed to store and retrieve data records that are collected in the time-intervals sequence. Time-series data stores are optimized to allow for fast insertion and retrieval of timestamp data to support complex data analysis. Some examples of time-series data include data from sensors in IoT devices, financial market data, data from application performance-monitoring tools, network data, and so on. In time-series data, update operations are rare, and delete operations are often done in bulk. Table 2-8 shows a sample time-series data store.

TABLE 2-8 Sample time-series data store

Timestamp	Device ID	Value
2022-01-19T11:15:21.321	1023	54.89
2022-01-19T11:15:21.534	1023	125.75

The following Azure services support this type of data store:

- Azure Time Series Insights
- OpenTSDB with HBase on HDInsight

Object data stores

This type of data store is designed and optimized for storing and retrieving large binary objects (unstructured data). An object consists of a unique ID, data, and its metadata.

Some object stores replicate data into multiple locations or regions for faster and parallel reads. Object data stores provide high scalability and offer a cost-effective solution for storing unstructured data. Unlike hierarchical file systems (which store data in folders and subfolders), object data stores store all objects in a flat address space. Table 2-9 shows a sample object data store.

TABLE 2-9 Sample object data store

Path	Blog	Metadata
App/images/welcome.jpge	0XAABBSSPPVVXCS..	{created 2022-01-19T11:15:20.123}
App/images/banner.jpge	0XAADDGGEFAAGFF..	{created 2022-01-19T11:15:21.321}
App/images/logo.jpge	0XAABBFDECAGFF..	{created 2022-01-19T11:15:21.534}

The following Azure services support this type of data store:

- Azure Blob Storage
- Azure Data Lake Store
- Azure File Storage

Skill 2.4: Design a data storage solution for nonrelational data

A nonrelational database, also called a NoSQL database, is a database that does not store data in a tabular schema of rows and columns. The design of a data-storage solution for a nonrelational data store depends on the specific business requirement and the type of nonrelational data. When designing a data store, one must consider requirements such as cost, compliance, data sensitivity, data isolation, location, and how the data store allows rapid changes and rapid replication. In this skill, we look at designing a data storage solution for nonrelational data.

This section covers how to:

- Recommend access control solutions to data storage
- Recommend a data storage solution to balance features, performance, and cost
- Design a data solution for protection and durability

Recommend access control solutions to data storage

There are various ways to access Azure Storage accounts based on the type of Azure Storage service and your needs. This section covers the various aspects of controlling access to Azure Storage and ADLS Gen 2, such as accessing storage with different authorization methods, securing Azure Storage using Azure Storage Firewall for limited access, and so on.

Authorize Azure Storage access

When you provision an Azure Storage account, it creates the following endpoints for each service:

- **Blob** Endpoint URL format

  http://<<YourStorageAccountName>>.blob.core.windows.net
- **File** Endpoint URL format

  http://<<YourStorageAccountName>>.file.core.windows.net
- **Table** Endpoint URL format

  http://<<YourStorageAccountName>>.table.core.windows.net

- **Queue** Endpoint URL format

  http://<<YourStorageAccountName>>.queue.core.windows.net

These endpoints can be accessed using the following authorization options:

- **Account key (primary/secondary)** The account key is also referred to as a *shared key*. When you provision a storage account, two 512-bit storage account access keys are automatically generated—a primary key and a secondary key. A client attempting to access this storage can pass one of these keys in the authorization header to gain access to the storage and its content.
- **Shared access signature (SAS)** SAS is a granular method of providing access to resources in a storage account. With SAS, you can grant limited access to containers and blobs in a storage account. SAS is a URI that contains an SAS token and grants restricted access rights to the Azure Storage resource. This access includes specific permissions and a time period.
- **Azure Active Directory integration** With this option, a client is authenticated using his or her AD credentials—like a user, group, or application service principal—and is given the appropriate Azure RBAC access. Clients must authenticate against Azure AD, obtain a security token, and pass that token to access the Azure Blob or Queue service. Azure Table Storage does not support Azure Active Directory–based authorization. Table 2-10 lists the RBAC roles and their associated permissions.

TABLE 2-10 RBAC roles and their associated permissions

RBAC Role	Storage Access Permission
Storage blob data owner	This RBAC role gets full permission to Azure Storage blob containers and data.
Storage blob data contributor	Read, write, and delete access to blob storage containers and blobs.
Storage blob data reader	Read and list blob storage containers and blobs.

- **Azure Active Directory Domain Services (Azure AD DS)**

authentication This authorization option is applicable only for the Azure Files service using the Server Message Block (SMB) protocol. This option supports identity-based access to Azure file shares over SMB.

- **On-premises Active Directory Domain Services** Again, this authorization option is for the Azure Files service only. This is an identity-based authorization method that uses AD DS that has been set up on an Azure virtual machine or that goes through an on-premises server.
- **Anonymous public read access** When the **Allow Blob Public Access** setting on the Azure Storage **Configuration** blade is set to **Enabled**, then all Azure Blob and containers can be accessed anonymously without authorization. Anonymous public access should be avoided and should be allowed only when there is an absolute need.

Secure Azure Storage access

Using any of the authorization options discussed in the previous section, you can access Azure Storage endpoints publicly from the internet. You can restrict this public access by configuring Azure Storage Firewall. When you turn on a firewall rule, incoming requests to a storage account are blocked. Only selected VMs and public IP addresses can access the Azure Storage account.

Access Azure Storage from an Azure virtual network using a service endpoint

The service endpoint method enables you to connect securely and directly to Azure Storage. In this method, routes to Azure Storage are optimized as traffic passes through the Microsoft Azure backbone to reach the Azure Storage service endpoint. Azure Storage can be accessed using a public endpoint, in which case traffic passes from the vNET to access Azure Storage. With a service endpoints configuration, Azure Storage service can identify when traffic is coming from an Azure vNET and see the private IP address of the vNET. You can approve the private IP address of a vNET in Azure Storage Firewall to allow connections from a vNET. With service endpoints, you cannot connect storage from peered vNETs and on-premises networks.

Access Azure Storage from Azure virtual networks using a private endpoint

A private endpoint offers a way to securely and privately access Azure Storage from an Azure vNET. With a private endpoint, traffic between clients inside the vNET and Azure Storage travels via the Microsoft Azure backbone instead of traversing the public internet. The private endpoint uses a dedicated IP address assigned from the vNET IP address range. With a private endpoint, a client can use the same connection string and authorization method. No changes are required in either the connection string or authorization methods.

Table 2-11 compares using service endpoints and private endpoints to access Azure Storage from an Azure vNET.

TABLE 2-11 Service endpoints versus private endpoints

	Service Endpoint	Private Endpoint
SERVICE DESCRIPTION	Extends a vNET to Azure Storage and allows Azure Storage to see whether the request is coming from the client's (such as a VM's) private IP address.	Enables you to access an Azure Storage service inside a vNET. It is assigned a private IP address to Azure Storage to connect to it.
CONNECTION METHOD	Connects to an Azure Storage public IP using optimized routes.	Connects to a private IP address assigned to Azure Storage.
CONNECTION FROM PEERED VNETS AND ON-PREMISES	Cannot connect from peered vNETs and on-premises networks using service endpoints.	Enables connectivity from regionally and globally peered vNETs and from on-premises using VPN or ExpressRoute.
CONNECTIVITY TO AZURE BLOBS, FILES, TABLES, DATA LAKE STORAGE GEN 2, AND STATIC WEBSITES	Does not require you to set up a separate service endpoint for each service. Once enabled on a subnet, you can access all Azure Storage services.	Requires a separate private endpoint for each type of service.

PRIVATE IP ADDRESS NEEDED	No.	Yes, per the Azure Storage service (Blob, Files, Table, and so on) and per storage account.
AZURE STORAGE FIREWALL CONFIGURATION	Required because Azure Storage Firewall controls access through a public endpoint, and service endpoints connect to the public endpoint.	No specific Azure Storage Firewall configuration is required on an Azure Storage account.
NSG	Because the destination is still a public IP, NSG needs to be opened.	No additional NSG rule is required because traffic is within the vNET only.
EXFILTRATION	Needs to be managed.	Built in.
IMPLEMENTATION	Simple.	More complex than with a service endpoint.
COST	No cost.	Yes, at an additional cost.

Access ADLS Gen 2 using access control lists (ACLs)

ADLS Gen 2 includes access control lists (ACLs) to provide fine-grained control to directories and files. ACLs provide a POSIX-style set of permissions to files and directories. ACLs grants read, write, and execute permission on files and directories. You can apply ACLs on security principals to access files and directories. You can have a maximum of 32 ACL entries per file and per directory. When a security principal (user, group, managed identity, or service principal) tries to perform some operation on a directory or file, an ACL check is performed to verify that the appropriate permissions are in place to do so.

Recommend a data storage solution to balance features, performance, and cost

These days, nonrelational data is a crucial asset for any organization. Choosing the right data storage solution is important, and requires the right balance of features, performances, and cost. Microsoft Azure provides a number of solutions to store nonrelational data. This section examines these services one by one.

Azure Storage

Let’s first take a look at the Azure Storage service as a whole. Then, you will explore Azure Blob Storage, Azure Table Storage, Azure File Share, ADLS, and Azure Cosmos DB in detail.

Azure Storage is Microsoft’s storage service on the Azure cloud platform. The Azure Storage service is highly available, scalable, durable, secure, fully managed, and widely accessible. You can access Azure Storage over HTTP and HTTPS. It is also accessible using client libraries that are available for various languages, including .NET, PHP, Node.js, Java, Python, and Ruby.

As an Azure Solutions Architect, it is imperative to use storage capacity optimally, meeting the performance requirements of the workload and keeping costs low. Azure Storage provides the following three storage access tiers:

- **Hot** This storage access tier is designed for frequently accessed data. The cost for data storage is higher than for the Cool and Archive tiers, but the cost for access is lower. This storage access tier provides 99.99 percent SLA for RA-GRS storage and 99.9 percent for other redundant storage accounts. You use this tier for production workloads or any other workloads in which data is accessed frequently.
- **Cool** This storage access tier is designed for data that is accessed less often and will remain stored for at least 30 days. It is optimally designed for data that will be accessed less frequently but that needs to be available instantly. The cost for data storage on this tier is lower than with the Hot tier, but the cost for access is higher. This tier has a slightly lower SLA than the Hot tier. It provides 99.9 percent SLA for RA-GRS storage and 99 percent for other redundant storage accounts. You use this tier for older backups that still need to be accessed quickly. You also use this tier for old media and documents that are not accessed frequently but that still need to be available instantly when required.
- **Archive** This storage access tier is designed for data that is accessed very rarely and for data that should remain stored for at least 180 days. The cost for data storage on this tier is the lowest, but the cost for access is the highest. Data stored in this tier is offline and cannot be

read or modified directly. Before reading, updating, or downloading data from this tier, the data must first be brought online. This process is called **blob rehydration** The metadata of blobs stored in this tier always remains online, and you can obtain this metadata without rehydrating. Data stored in this tier takes several hours to retrieve, depending on the priority of rehydration. You use this tier for old data that is rarely accessed, such as old backups, raw data, and so forth. You also use this tier for data that must be maintained for compliance purposes but that is rarely accessed.

- Table 2-12 compares the Hot, Cool, and Archive access tiers.

TABLE 2-12 Azure Storage Hot, Cool, and Archive tiers

	Hot access tier	Cool access tier	Archive access tier
STORAGE COST	High	Lower	Lowest
ACCESS COST	Low	Higher	Highest
EARLY DELETION PERIOD	NA	30 days	180 days
EARLY DELETION FEE	No	Yes	Yes
SLA	99.99 percent for RA-GRS and 99.9 percent for others.	99.9 percent for RA-GRS and 99 percent for others.	Offline. Data must be moved to an online tier (Hot or Cool) before read/write.

EXAM TIP

Data in archive storage is not readily available for immediate read. You would need to copy or change the tier to Hot or Cool for instant read. To learn more, visit the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-rehydration?tabs=azure-portal*.

Along with these access tiers, Azure Storage accounts are also classified into two performance tiers: Standard and Premium. The Standard tier was the first one introduced by Microsoft. Later, Microsoft introduced the Premium tier, which is used for storing blobs, files, tables, queues, and unmanaged and managed VM disks.

- **Standard** In the Standard tier, an unmanaged disk is charged based on the amount of storage consumed. For example, if you attach 128 GB as a standard unmanaged disk (page blob) and you consume only 50 GB, then you are charged for only 50 GB. The Hot, Cool, and Archive access tiers are available only in the Standard performance tier and for General Purpose V2 storage account types. The Standard performance tier supports the following kinds of storage:
  - Locally redundant (LRS)
  - Geographically redundant storage (GRS)
  - Zone redundant storage (ZRS)
  - Read access geographically redundant storage (RA-GRS)
  - Zone-redundant storage (ZRS)
  - Geo-zone-redundant storage (GZRS/RA-GZRS) redundancy
- **Premium** This is a high-performance, low-latency tier. It stores data in a solid-state drive, so its performance is better than that of the Standard tier. This performance tier is available in General-Purpose V1, General-Purpose V2, File Storage, and Block Blob Storage account types. It supports the following kinds of storage:
  - Locally redundant (LRS)
  - Zone-redundant storage (ZRS) (in Block Blob Storage)

Table 2-13 compares the Standard and Premium performance tiers.

TABLE 2-13 Azure Storage Standard and Premium performance tiers

	Standard performance tier	**Premium performance tier**
ACCOUNT KIND	General-Purpose V1 General-Purpose V2 Blob Storage	General-Purpose V1 General-Purpose V2 File Storage

		Block Blob Storage
UNDERLYING HARDWARE	HDD	SSD
COST	Low	High
READ-WRITE LATENCY	Relatively high	Low
THROUGHPUT	Relatively low	High
REDUNDANCY	LRS, ZRS, GRS, RA-GRS, ZRS, and RA-GZRS	LRS, ZRS (in Block Blob Storage)
RECOMMENDED FOR	All non-latency and throughput workloads	For all critical applications that require low latency and high throughput
CORE STORAGE SERVICES	Blob, File, Queue, Table	Blob

Azure Blob Storage

Azure Blob Storage is Microsoft’s solution for object storage. It is optimized for storing enormous amounts of unstructured data. It provides highly scalable, available, secure, and durable storage. *Blob* stands for binary large object, which includes objects such as audio, video, text, images, and training documents. You can access Azure Blob Storage over HTTP and HTTPS. It is also accessible using client libraries that are available for various languages, including .NET, PHP, Node.js, Java, Python, and Ruby.

Following are some scenarios in which you would use Azure Blob Storage:

- To store an application’s images, audio, and videos, for direct access through the browser
- To store backup and archive data
- For streaming audio and video
- As a cost-effective data storage solution

Azure Table Storage

Azure Table Storage is a NoSQL data store where you can store key–value pairs with a schema-less design. Data stored in Azure Table Storage is in the form of entities, which are like rows. An entity can have a maximum of 252 properties; additionally, entities have system properties that specify a partition key, row key, and timestamp. Each entity can be a maximum of 1 MB in size.

In Azure Table Storage, you can store terabytes of semi-structured data. Data stored in Azure Table Storage is highly available. This is because Azure internally maintains three replicas in the primary region, and if the storage account is geo-redundant, then the data is replicated in the secondary region too.

Azure Table Storage is highly scalable; there is no manual need to shard a dataset. A table can span up to the maximum size of the storage account. (Sharding was covered earlier in this chapter.) By default, all data stored in Azure Table Storage (data at rest) is encrypted.

Following are some scenarios in which you would use Azure Table Storage:

- To store and query a huge set of nonrelational, schema-less data
- For nonrelational data that does not require complex joins or foreign keys
- For faster retrieval of data using the key (partition key)
- As a cost-effective data storage solution

Azure File Share

Azure File Share is Microsoft’s fully managed file share solution on the cloud. You can access it using Server Message Block (SMB) protocol or Network File System (NFS) protocol on Windows, Linux, and macOS operating systems. You can mount Azure File Share on the cloud as well as on an on-premises server. You can easily migrate your on-premises Windows file share to Azure File Share. You can also access file share over the internet using URL (Azure File Share endpoint) and shared access signature (SAS). Azure File Share can also be accessed using REST API and Azure Storage client libraries.

Following is one scenario in which you would use Azure File Storage:

- For enterprise-grade and secure file shares solutions on Microsoft's Azure cloud platform

Azure Data Lake Storage (ADLS)

Azure Data Lake Storage (ADLS) is a fully managed, massively scalable, highly available, durable, and secure data lake for high-performance big-data analytics workloads. ADLS is built upon Azure Storage as its foundation.

There are two types of ADLS:

- **Azure Data Lake Storage Generation 1 (ADLS Gen 1)** ADLS Gen 1 is a hyperscale, fully managed repository for big-data analytic workloads to store data of any size and type. ADLS Gen 1 is accessible from Hadoop using the WebHDFS-compatible REST APIs.
- It offers enterprise-grade capabilities such as availability, scalability, security, manageability, and reliability.

> ***Note*** ADLS Gen 1 is set to be retired in February 2024. For this reason, Microsoft recommends you use ADLS Gen 2, discussed next.

- **Azure Data Lake Storage Generation 2 (ADLS Gen 2)** ADLS Gen 2 has all the key features of ADLS Gen 1. It offers a hierarchical file system with low-cost tiered storage, strong consistency, and disaster recovery capabilities. The hierarchical file system allows for the organization of data within directories and files and significantly improves the performance of analytics jobs. ADLS Gen 2 also provides ACLs for fine-grained control of directories and files. ACLs provide a POSIX-style set of permissions to files and directories. ADLS Gen 2 supports storage and transactions at a low cost. Its Azure Blob Storage Lifecycle Management feature also helps reduce costs as data transitions through its lifecycle.

The primary scenario in which you would use ADLS is as a massively scalable storage solution for high-performance big-data analytics workloads.

Azure Cosmos DB

Azure Cosmos DB is a NoSQL, multimodel, fully managed, globally distributed, high-through-put database. Cosmos DB supports key–value pair–based, column family–based, document-based, and graph-based databases. It provides 99.999 percent availability for multiregion accounts and 99.99 percent availability for single-region accounts.

Azure Cosmos DB guarantees less than 10-millisecond latencies for both reads (indexed) and writes at the 99th percentile. Azure Cosmos DB supports multiple APIs, such as MongoDB API, Graph API, Cassandra API, Gremlin API, SQL API, and Cosmos DB Table API. It also provides SDKs for multiple programming languages, including Python, .NET, Java, Node.js, JavaScript, and the like.

Following are some scenarios in which you would use Azure Cosmos DB:

- For nonrelational data that is in key–value pair–based, document-based, graph-based, or column family–based form
- For business-critical applications that require near-real response times in milliseconds and high throughput
- For applications that require a massive and global scale with high availability and disaster recovery
- For a social media application or IoT and telematics application that requires enormous data ingestion or unpredictable loads

Design a data solution for protection and durability

As data continues to grow exponentially, it is important to protect it. This involves a set of processes and strategies to safeguard an organization’s data from loss, corruption, and compromise. Microsoft Azure provides several features in Azure Storage and ADLS Gen 2 for data protection, including the following:

- Azure Resource Manager locks
- Blob versioning
- Soft delete

- Immutable storage policies
- Point-in-time restore
- Blob snapshots
- Deleted storage account recovery
- Data encryption

Azure Resource Manager locks

Microsoft Azure includes a great feature called Resource Manager locks. This feature is not just limited to Azure Storage accounts. You can apply locks to subscriptions, resources groups, and resources. For example, by applying a Delete lock, you can prevent the accidental deletion of Azure Storage accounts. This feature is also applicable to ADLS. However, it does not protect blobs or containers stored in Azure Storage accounts.

There are two types of locks:

- **CannotDelete** This prevents users from deleting Azure Storage accounts, but they can still read and modify the account configuration.
- **ReadOnly** This prevents users from deleting and modifying Azure Storage accounts, but they can still read its configuration.

It is recommended to lock all your storage accounts.

Blob versioning

Blob versioning enables you to maintain previous versions of an object. When blob versioning is enabled, Azure Storage accounts automatically create a new version with a unique ID and maintain the previous version of the object.

Each version is identified by version ID. When you create a new blob, it has only one version: the current version. When you modify the blob, it creates a new version, which becomes the current version. When you delete a blob without specifying the version, it deletes the current version; then, the previous version becomes the current version.

You can read or delete older versions by providing the version ID. The blob versioning feature cannot help you recover data in the event of the deletion of the Azure Storage account or container.

Soft delete

The soft delete feature enables you to protect data from accidental deletion by maintaining a copy of deleted data for a set retention period. During this retention period, you can restore the data, which is said to be *soft deleted*, to its state at the time of deletion. You can specify a retention period of from 1 to 365 days. When the retention period is over, the soft-deleted data is permanently deleted. The soft delete feature does not protect data in the event of the deletion of the Azure Storage account. Soft delete can be enabled on a blob, snapshot, version, or container.

Immutable storage policies

Immutable storage stores data in a write once, read many (WORM) state. When you enable immutable storage, you cannot modify or delete data for a specified interval or until you clear the policy.

There are two types of immutable storage policies:

- **Time-based retention policies** With this type of policy, data is immutable until the retention period expires. You cannot modify or delete the data during the retention period. When the retention period is over, you can delete the data, but you cannot overwrite it.
- **Legal hold policies** With this type of policy, data is immutable until the legal hold is explicitly cleared. When you enable a legal hold, data can be created and read but not modified or deleted.

You can set immutable storage policies scoped at the blob version level or at the container level. For blob version scope, you must enable version-level immutability on the Azure Storage account or container. You cannot set immutable storage policies on ADLS.

These policies can help you meet legal or regulatory compliance requirements, hence it is recommended to enable them on Azure Storage accounts that store business-critical data.

Point-in-time restore

The point-in-time restore feature allows to restore one or more sets of block blobs to a previous state. Before you enable this feature, you must make sure that soft delete, change feed, and blob versioning are already enabled.

This feature is supported for General-Purpose v2 storage accounts in the Standard performance tier only. It is not supported for ADLS. Also, you can recover data (using point-in-time restore) stored in the Hot and Cool access tiers only. This feature is applicable to block blobs only. It does not support page blobs, append blobs, or operations on containers (such as delete container).

Blob snapshots

A blob snapshot is a copy of the blob along with its system properties created at a set point in time. When you create a snapshot, you can read, copy, and delete it, but you cannot modify it. You can have any number of blob snapshots. However, blob snapshots are supported only in the Hot and Cool tiers. The Archive tier does not support blob snapshots. When you create a blob snapshot, it will persist until it is explicitly deleted either individually or as a part of a Delete Blob operation. You cannot acquire a lease on a snapshot. Snapshots are billed at the same rate as the original blob.

Deleted storage account recovery

Microsoft provides support to recover an Azure Storage account that is accidentally deleted. The prerequisites for storage account recovery are as follows:

- The storage account is of the Azure Resource Manager (ARM) deployment model type.
- The storage account was deleted within the past 14 days.
- A storage account with the same name does not exist.
- A resource group of the deleted storage account exists.
- The user recovering the storage account has Microsoft.Storage/storageAccounts/write permissions.

You can recover the deleted storage by doing one of the following:

- In the Azure Portal, navigate to any existing storage account and select **Recover Deleted Account** in the **Support + Troubleshooting** section. Then select the deleted storage account from the dropdown list and click the Recover button.
- Raise a support ticket.
- For a classic storage account, you must seek help from the support team.

The least privileges principle

A best practice to protect data is to follow the least privileges principle and grant limited permission to limited users. For example:

- Limit access to the storage account contributor, storage blob data contributor, and subscription owner RBAC roles.
- Assign roles to groups instead of to individual users.

For more details, refer to the section “Recommend access control solutions to data storage” earlier in this chapter.

Data encryption

Data encryption is an important aspect of data protection. Even if unauthorized users gain access to encrypted data storage, they can’t read encrypted data. Hence, it is recommended to encrypt data at rest, in transit, and in motion. For more details, refer to the section “Recommend a solution for encrypting data at rest, data in transmission, and data in use” earlier in this chapter.

Chapter summary

- *DTU* stands for database transaction unit, and it blends CPU, memory, and I/O usage.

- The vCore purchasing model provides flexibility to independently pick compute, memory, and storage based on your workload needs.
- The autopause feature of serverless databases helps save on cost. There is no compute cost when the database is in the paused state, but you do pay for storage costs.
- All databases deployed in elastic pools share DTUs.
- Horizontal scaling (scale-out) can be implemented by using the Elastic Database tools.
- Read scale-out is the best match for an analytics workload that only reads data.
- Encryption at rest is mandatory for an organization to be compliant with HIPAA, PCI, and FedRAMP standards.
- The Always Encrypt feature protects data at rest, in motion, and in use.
- SSL is the predecessor of TLS. Always try to use the latest version of TLS.
- By default, all new databases deployed in Azure SQL Database are encrypted at rest using transparent data encryption (TDE).
- Azure SQL Managed Instance is suitable when you need to efficiently migrate databases from on-premises to Azure and leverage the SQL Server database engine, including the SQL Agent, cross-database queries, and offloading management work to Microsoft.
- Nonrelational databases are also known as NoSQL databases. key-value, document, column family, graph, time series, and objects are a few examples of nonrelational databases.
- Data in archive storage is not readily available for immediate read. You would need to copy or change the tier to Hot or Cool for instant read.
- Azure Databricks is a fully managed, fast, and easy analytics platform that is based on Apache Spark on Azure. Azure Databricks is natively integrated with Azure services such as Blob Storage, Azure Data Lake Storage, Cosmos DB, Azure Synapse Analytics, and the like.
- It is recommended that you use both a partition key and a row key to query Azure Table Storage. When you use the partition key and the row key, data retrieval is very fast; if you don’t, a table scan operation

is performed to search for the data in Azure Table

- Storage. Select the partition key and the row key wisely so that all data retrieval queries include both.
- Cosmos DB can be deployed into multiple regions. This allows for quick disaster recovery and keeps data closer to the user to improve network latency.
- A modern data platform consists of data ingestion, data storage, data preparation, data modeling, and data serving.
- Azure Blob Storage Lifecycle Management can be used to achieve significant cost savings by applying a simple rule to move blobs to the Cool and Archive tiers and to delete blobs.
- Azure Storage can be accessed programmatically using .NET, PHP, Nose.js, Java, Python, Ruby, and so on, as well as using tools such as Azure Storage Explorer, AzCopy, and Visual Studio Cloud Explorer.
- It is recommended that you rotate Azure Storage keys either manually or automatically using Key Vault.
- Always try to use SAS tokens to delegate access to Azure Storage instead of sharing the account key.
- It is recommended to encrypt data at rest, in transit, and in motion.

Thought experiment

Now it is time to validate your skills and knowledge of the concepts you learned in this chapter. You can find answers to this thought experiment in the next section, “Thought experiment answers.”

As an Azure Solutions Architect working for Contoso, you are responsible for architecting and designing applications and making the correct technical decisions to meet Contoso’s business goals. Contoso has decided to adopt a cloud environment and to migrate all its applications from an on-premises environment to the Microsoft Azure cloud platform. Although you support this migration, some business stakeholders are against it. Contoso has permitted the migration of only one LOB application. As an Azure Solutions Architect, you need to successfully migrate this application and prove yourself by accomplishing all the business requirements, as well as showing all the benefits of the cloud adoption.

The details of the current application are as follows:

- Develop a web application using the Microsoft .NET technology stack and deploy it to the IIS server.
- Deploy the database to SQL Server 2012 Standard Edition. The current size of the database is 5 TB.
- Use SQL Server Integration Services (SSIS) packages to connect to the business partners' SFTP server and retrieve business data.
- Archive and maintain old unstructured data in the local storage system. This data needs to be stored for three years to meet the company's compliance needs. The total size of the old data is 70 TB.
- Store the application users' uploaded images and videos in a local file system. These videos and images are typically used very frequently in the first six months and are very rarely used after six months.

After consulting with business users and other stakeholders, you have identified the following business requirements:

- Data stored in a SQL Server database, as well as all videos and images uploaded by the user, should be encrypted at rest and in motion.
- Customer Social Security numbers (SSNs) stored in the database should be encrypted. Even database administrators and cloud security operation people should be unable to read these numbers. The keys required to encrypt this data should not be stored in the cloud.
- A database should provide 99.99 percent availability.
- Archived data needs to be stored in Azure. The data migration solution should not depend on the network bandwidth, because Contoso has low network bandwidth. The cost of storage should be lower. An easy solution is to require the purging of data older than three years.
- Your solution should be cost-effective.
- Your solution should leverage PaaS services as far as possible so that Contoso can offload management work.
- Migration should be smooth, with few code changes.

With this information in mind, answer the following questions:

1. What database tier will you use for the SQL Server Database?
2. How will you address encryption requirements?
3. What solution should you implement to collect data from the partners' SFTP server?
4. What solution should you implement to transfer archived data to Azure cloud storage?
5. What would you recommend for purging old data without any manual intervention?

Thought experiment answers

This section contains the answers to the thought experiment questions.

1. Because the size of the current database is 5 TB, the most suitable option is to deploy this database into the Serverless tier of the vCore purchasing model. You could also use SQL Server on a VM. However, Contoso would like to leverage PaaS services, so SQL Server on a VM is not recommended. This option also provides autoscaling with the autopause feature to save compute cost. The Serverless tier offers 99.99 percent availability, which meets another business requirement.
2. The data stored in the Serverless database tier will be encrypted at rest by enabling TDE. It also always enforces TLS/SSL connection, irrespective of the `Encrypt` or `Trust-ServerCertificate` setting in the connection string. The Always Encrypted feature can be used to encrypt data in use. The Azure Storage service, by default, encrypts data with Microsoft-managed keys. You can encrypt data using your own keys also. Data encryption in transmission can be enforced by enabling the **Secure Transfer Required** setting in the Azure Storage account configuration. Thus, you can fulfill all encryption-related business requirements.
3. Azure Data Factory can be used to collect data from business partners. The existing SSIS package can be executed into SSIS runtime in the

Azure Data Factory.

4. Because Contoso has low network bandwidth, it is advisable to use the Azure Data Box solution to ship data offline to the Microsoft Azure datacenter. The size of the archived data is 70 TB, which can easily fit into Azure Data Box. The Azure Import/Export service will require the creation of multiple jobs and investment to procure the required Azure Data Box disks.
5. Azure Blob Storage Lifecycle Management is the recommended solution for purging old data. A simple rule can be created to delete data after three years. Similarly, videos and images uploaded by customers should be kept in a Hot access tier for the first six months. An Azure Blob Storage Lifecycle Management policy rule can be created to move it to the Cool access tier and to automatically delete it after three years.

EXAM TIP

Be aware that use case–style exam questions often provide more information than is needed to answer the question.

Chapter 3

Design business continuity solutions

Cloud Solution Architects understand the importance and need to design a business continuity solution. Most enterprises have a well-established business continuity and disaster recovery plan, also known as a BC/DR plan. Typically, the best starting point when defining and choosing a business continuity solution is to perform a business criticality assessment. A criticality assessment helps you determine the criticality of systems and their impact on the business if an outage occurs. This assessment should guide you in developing the right business continuity strategy for the company. When you perform the criticality assessment and identify critical applications, the next step is to figure out your backup and disaster recovery strategy.

The AZ-305 certification exam expects you to demonstrate a solid understanding of designing a business continuity and disaster recovery plan. The Azure Solution Architect certification is an expert-level exam, so this exam expects you to have advanced-level knowledge of each domain objective.

Skills covered in this chapter:

- Skill 3.1: Design a solution for backup and disaster recovery
- Skill 3.2: Design for high availability

Skill 3.1: Design a solution for backup and disaster recovery

The success of any application, especially when it runs in the cloud, depends on how gracefully it handles failures and continues to deliver as much business value as possible. This approach is also known as designing for failure. When designing a solution for backup and recovery, you should first identify failure situations and their potential impacts on your organization. Then you should perform analysis and a criticality assessment, develop a business continuity strategy, and document your data protection requirements. Finally, you should develop backup and recovery plans to address the data protection requirements identified by your analysis.

Note Successful architects typically follow the same approach while designing backup and recovery solutions.

This section covers how to:

- Recommend a recovery solution for Azure, hybrid, and on-premises workloads that meet recovery objectives (recovery time objective [RTO], recovery level objective [RLO], recovery point objective [RPO])
- Understand the recovery solutions for containers
- Recommend a backup and recovery system for compute
- Recommend a backup and recovery solution for databases
- Recommend a backup and recovery solution for unstructured data

Recommend a recovery solution for Azure, hybrid, and on-premises workloads that meets recovery objectives (recovery time objective [RTO], recovery level objective [RLO], recovery point objective [RPO])

When your systems are unavailable, your company could directly or indirectly face some reputational harm. Large-scale outages or disasters can disrupt your business, staff, and users. Also, your company could face financial losses such as lost revenue or penalties for not meeting availability agreements for your services.

Business continuity and disaster recovery (BC/DR) plans are formal documents that organizations develop to cover the scope and steps to be taken during a disaster or large-scale outage. Each disruption is assessed on its merit.

For example, consider a scenario in which an earthquake has damaged your datacenter power and communication lines. This situation has rendered your corporate datacenter useless until power is restored and lines of communication are fixed. A fiasco of this magnitude could take your organization’s services down for hours or days, if not weeks. This is why you need a complete BC/DR plan: to get the services back online as quickly as possible.

RTOs, RPOs, and RLOs

As part of your BC/DR plan, you must identify your application’s recovery time objectives (RTOs) and recovery point objectives (RPOs).

Both objectives, at a minimum, help you determine a baseline approach with a clear commitment to a speed of recovery (recovery time objectives, or RTOs) and risk of data loss (recovery point objectives, or RPOs).

Before diving into the solutions, let us look at three widely used terms to define recovery objectives RPO, RTO, and RLO.

- **Recovery point objective (RPO)** The recovery point objective is used to determine the maximum time between the last available backup and a potential failure point. Also, the RPO helps determine the amount of data a business can afford to lose in a failure. For example, if your backup occurs every 24 hours at 4 a.m. and a disaster happens at 1 p.m. the following day, then 9 hours of data would be lost. If your company’s RPO is 12 hours, then no data would be lost because only 9 hours would have passed, and you would have a better recovery point backup available from which you could recover. However, if the RPO is 4 hours, then your backup strategy would not meet your RPO requirement, and damage would occur to the business.
- **Recovery time objective (RTO)** The recovery time objective is used to determine the maximum time a data recovery process can take. It is

defined by the amount of time the business can afford for the site or service to be unavailable. For example, let's say one of your applications has an RTO of 12 hours. This means your business can manage for 12 hours if this application is unavailable. However, if the downtime is longer than 12 hours, your business would be seriously harmed.

- **Recovery level objective (RLO)** The recovery level objective defines the granularity with which you must be able to recover data regardless of whether you must be able to recover the whole application stack.

Figure 3-1 explains the recovery point and recovery time concepts. The recovery time is the amount of time needed to recover the data, whereas the recovery point is the last point a successful backup was made.

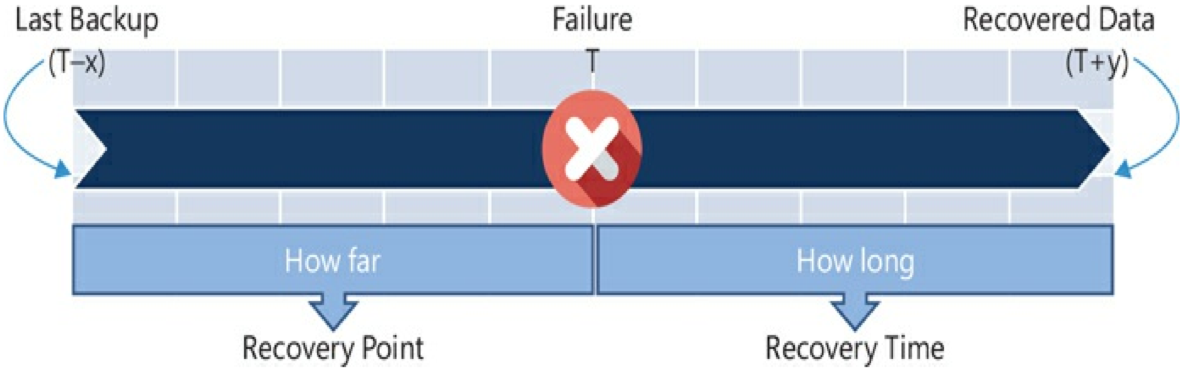


FIGURE 3-1 Recovery point objective and recovery time objective

Azure Site Recovery

To meet your business continuity and disaster recovery strategy, you should leverage Azure Site Recovery.

Azure Site Recovery supports applications running on Windows- or Linux-based physical servers, VMware, or Hyper-V. Using Azure Site Recovery, you can perform application-aware replication to Azure or to a secondary site. You can use Azure Site Recovery to manage replication, perform a DR drill, and run failover and failback.

Azure Site Recovery (ASR) is recommended for application-level protection and recovery:

- ASR can be used to replicate workloads running on a supported machine.
- ASR offers near-real-time replication with RPOs as low as 30 seconds. Typically, this meets the needs of most critical business apps.
- ASR can take app-consistent snapshots for single- or multi-tier applications.
- ASR also integrates with SQL Server AlwaysOn and other application-level replication technologies such as Active Directory replication and Exchange database availability groups (DAGs).
- ASR recovery plans are very flexible and enable you to recover the entire application stack with a single click and include external scripts and manual actions in the plan.
- ASR offers advanced network management capabilities to simplify app network requirements, such as the ability to reserve IP addresses, configure load-balancing, and integrate with Azure Traffic Manager for low RTO network switchovers.
- A rich automation library is available, which provides production-ready, application-specific scripts that can be downloaded and integrated with recovery plans.

> *Important* **Frequently Asked Questions About ASR**
>
> Microsoft documentation has a very comprehensive list of FAQs for ASR that cover various workload types and disaster recovery scenarios. To learn more, visit the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-faq*.

Azure Backup service

The Azure Backup service provides a secure and cost-effective solution to back up your data and keep it safe and recoverable in case of service disruption, accidental deletion, or data corruption. ASR and Azure Backup complement each other, helping organizations design end-to-end BC/DR plans.

Azure Backup helps you back up files, folders, machine states, and other workloads running on on-premises and Azure virtual machines (VMs). You can use Azure Backup to protect the following workload types:

- **Azure VMs** Use the Microsoft Azure Recovery Services agent (MARS) to back up both Windows and Linux VMs.
- **Azure Managed Disks** Back up Azure Managed Disks using Backup vault.
- **Azure File shares** Back up Azure File shares using the Recovery Service vault.
- **SQL Server in Azure VMs** Back up SQL Server Databases running on Azure VMs.
- **SAP HANA databases in Azure VMs** Back up SAP HANA databases running on Azure VMs.
- **Azure Database for PostgreSQL servers** Back up Azure Database for PostgreSQL servers with long-term retention.
- **Azure Blobs** Azure Backup helps you protect blobs in the storage account and enhance the data protection at scale.
- **On-premises machines** Use Microsoft Azure Recovery Services (MARS) agent to back up Windows Servers, or use System Center Data Protection Manager (DPM) or Azure Backup Server (MABS) agent to protect VMs (Hyper-V, VMware).

Azure Backup stores backed-up data in vaults: Recovery Services vault and Backup vault. A vault is a storage entity in Azure that holds data, such as backup copies, recovery points, and backup policies.

Consider the following recommendations when you create storage vaults:

- Use separate vaults for Azure Backup and Azure Site Recovery.
- Use role-based access control (RBAC) to protect and manage access to storage vaults.
- Design for redundancy. This means specifying how data in vaults is replicated. Azure offers the following three options to replicate data:
    - **Locally redundant storage (LRS)** To protect data from server rack and drive failures, use LRS. LRS replicates data three times

within a single datacenter in the primary region and provides at least 99.999999999 percent (11 nines) annual uptime.

- **Geo-redundant storage (GRS)** To protect data from region-wide outages, use GRS. GRS replicates data to a secondary region. The Recovery Services Vault uses GRS by default.
- **Zone-redundant storage (ZRS)** ZRS replicates data in availability zones, guaranteeing data residency and resiliency in the same region.

Understand the recovery solutions for containers

Many organizations' cloud adoption strategies use containers to focus heavily on modern application development. Containerization is an approach used in software development in which an application or service and its dependencies are packaged together as a container image. Containerized applications help organizations accelerate time to market, reduce operating overhead, make workloads more portable, and modernize legacy workloads.

The Azure Kubernetes Service (AKS) is the most popular service used by organizations to deploy and manage containerized applications in Azure. Although AKS is a fully managed service that provides built-in high availability (HA) by using multiple nodes in a virtual machine scale set (VMSS), the built-in HA within the region does not protect your system from failure.

Consider the following best practices and recommendations to maximize uptime and faster recovery of solutions in case of regional disruption:

- Deploy AKS clusters in multiple regions. Choose Azure-paired regions, which are designed explicitly for disaster-recovery scenarios.
- Use Azure Container Registry to store container images and geo-replicate the registry to each AKS region. You need a Premium SKU to use geo-replicated instances of Azure Container Registry.
- Back up AKS clusters using Velero and Azure Blob storage. Velero is an open-source community standard tool you can use to back up and restore Kubernetes cluster objects and persistent volumes.

EXAM TIP

Velero is an open-source community tool you can use to back up and restore AKS cluster persistent volumes and other additional cluster objects.

Recommend a backup and recovery solution for compute

As you learned earlier in this chapter, you can use Azure Backup to back up supported compute resources such as Azure virtual machines and restore them seamlessly when needed. Azure Backup consists of two tiers:

- **Snapshot** In this tier, the backups are stored locally for five days. The restore process from the snapshot tier is much faster because there is no wait time for snapshots to copy to the vault before triggering the restore.
- **Recovery Services vault** After the snapshots are created, Azure Backup transfers the data to the Recovery Services vault for additional security and longer retention.

Consider the following recommendations for Azure virtual machine backup and recovery:

- **Backup schedule policies** Create separate backup policies for critical and noncritical virtual machines. Consider scheduled start times for different backup policies at different times of the day and ensure the time slots do not overlap.
- **Backup retention policies** Implement both short-term (daily) and long-term (weekly) backups.
- **Cross-Region Restore (CRR)** Using CRR, you can also restore Azure VMs in a secondary region. This option lets you conduct drills to meet audit or compliance requirements.
- **Optimize restore time** During the restore process from a single vault, it is recommended that you use a general-purpose v2 Azure Storage account for each VM to avoid transient errors. For example, if five

VMs are restored, use five different storage accounts.

- **Monitoring** Use Azure Monitor alerts for Azure Backup to receive notifications when a backup or restore fails.

EXAM TIP

In a VM replication scenario, create a Recovery Services vault in any region except the source region you want to replicate from. In a VM backup scenario to protect data sources, create a Recovery Services vault in the same region as the data source.

Recommend a backup and recovery solution for databases

You learned earlier in this chapter that Azure Backup is the service you should use to back up and recover SQL Servers running on virtual machines and SAP HANA databases running on Azure virtual machines.

This section covers recommendations for the backup and recovery of the Azure SQL Database.

The Azure SQL Database and Azure SQL Database Managed Instance have a built-in automatic backup system, also known as a point-in-time restore (PITR). The PITR retains backups for 7 to 35 days, depending on your database service tiers. The PITR allows you to restore a database from these backups to any time in the past within the retention period. You incur an additional cost only if you use the restore capability to create a new database.

The automated database creates full weekly backups, differential backups every 12 to 24 hours, and transaction log backups every 5 to 10 minutes.

You might wonder, what if you need to keep backups for longer than 35 days for audit or compliance reasons? In this case, you can use the long-term retention (LTR) feature. With LTR, you can store Azure SQL Database backups in read-access geo-redundant storage (RA-GRS) blobs for up to 10 years. If you need access to any backup in LTR, you can restore it as a new database using either the Azure Portal or PowerShell.

More Info **Long-Term Retention (LTR) Backup Data Using Azure Backup Archive Tier**

You can also consider the Azure Backup Archive tier for long-term data retention to meet data compliance requirements. You can find the list of supported workloads and Azure regions for the Azure Backup Archive tier in the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/backup/archive-tier-support*.

Recommend a backup and recovery solution for unstructured data

Azure Blob storage is a storage solution for unstructured data. Unstructured data doesn't adhere to a particular data model or definition. Examples of unstructured data include text and binary data.

Azure Storage account has a built-in local data protection solution called operational backup for Blobs. The operational backup solution protects the block Blobs from various data loss scenarios such as container deletion, Blob deletion, or accidental storage account deletion. The data is stored locally within the storage account and can be recovered when needed to a selected point in time within a maximum retention period of 360 days.

Consider the following recommendations to enhance data protection and recovery for Azure Blob storage:

- **Soft delete** You can enable soft delete at the container level or for Blobs. When soft delete is enabled, you can restore a container and its Blob at the time of deletion.
- **Versioning** Blob versioning automatically maintains previous versions of a Blob. When Blob versioning is enabled, you can restore an earlier version of a Blob when needed.
  - **Resource locks** Soft delete does not protect you against the deletion of the storage account. You must use resource locks to prevent the accidental deletion of the storage account. You can use the following lock types:

- **CanNotDelete** Authorized people can read and modify a resource but can’t delete it.
- **ReadOnly** Authorized people read but cannot modify or delete a resource.

Skill 3.2: Design for high availability

Resiliency, fault tolerance, and high availability are essential attributes for mission-critical systems so that they can recover from failures and continue to function. You should design cloud applications keeping in mind the fact that failures do happen, so you should be able to minimize the effects of failing components on business operations. Every system has particular failure modes, which you must consider when designing and implementing your application.

High availability (HA) is the capability of any computing system to provide desired and consistent uptime, even in the event of an underlying infrastructure failure. This requirement is vital for mission-critical systems that will not tolerate an interruption in the service availability. HA is also imperative for any system for which any downtime would cause damage or monetary loss.

HA systems guarantee a percentage of uptime. The number of nines in the percentage is usually used to specify the degree of high availability offered. For example, “five nines” indicates a system that is up 99.999 percent of the time. A system with 99.9 percent uptime can be down only 0.1 percent of the time, so in a year, to meet 99.9 percent SLA, you can only have 8.77 hours of downtime.

Designing apps for high availability and resiliency usually means running them in a healthy state without significant downtime. This design begins with gathering requirements and asking the right questions. For example:

- How much downtime is acceptable?
- What does this potential downtime cost your business?
- What are your customer’s availability requirements?
- How much can you invest in making your application highly available?

- How much risk versus the cost can you tolerate?

Following are three essential characteristics of a highly available system:

- **Redundancy** This means ensuring that any elements crucial to the system operations have additional redundant components that can take control in the event of failure.
- **Monitoring** This means gathering data from a running system and identifying when a component fails or fails to respond.
- **Failover** This refers to a mechanism that could automatically switch from the currently active component to a redundant component if monitoring shows a breakdown of the active component.

Microsoft Azure services are designed and built at every layer to deliver the highest levels of redundancy and resilience. Azure infrastructure is composed of geographies, regions, and availability zones, limiting the failure and the potential impact on customer applications and data.

Microsoft defines its SLA for each Azure service. If you need to have a higher SLA than what Azure offers, you can set up redundant components with failover.

This section covers how to:

- Identify the availability requirements of Azure resources
- Recommend a high-availability solution for compute
- Recommend a high-availability solution for non-relational data storage
- Recommend a high-availability solution for relational databases

Identify the availability requirements of Azure resources

As you learned in the previous section regarding high availability (HA) and different service-level agreements (SLAs), depending on the SLA, your cloud workload can provide a continuous user experience with no apparent downtime, even when things go wrong.

Highly available workloads have the following quality attributes:

- They do not have a single point of failure.
- They can scale on demand to meet performance needs when load increases.
- They can detect and respond to failure gracefully.

Consider the following recommendations when defining the requirements to design resilient and highly available Azure applications:

- **Identify workload types and usage patterns** The SLA in Azure defines Microsoft's commitment to the uptime of the Azure services. Different services have different SLAs. For example, App Services have an SLA of 99.95 percent, and an Azure SQL Database has an SLA of 99.99 percent. Both services together provide a composite SLA of 99.94 percent. Understanding your overall SLA expectation for the application is vital to designing the application architecture appropriately to meet the business SLA need.
- **Cost and complexity** As you move toward more nines, the cost and complexity grow. The higher the SLA, the less frequently the service can go down, and the quicker the service must recover. To achieve four nines (99.99 percent), you can't rely on manual intervention to recover from failures. The application must be self-diagnosing and self-healing.
- **Start with failure mode analysis (FMA)** FMA is a process for building resiliency into a system by identifying possible failure points in that system. Create end-to-end dependency mapping in the application architecture and identify dependencies. Pay particular attention to dependencies that might be a single point of failure or cause bottlenecks to scale. If a workload requires 99.99 percent uptime but depends on a service with a 99.9 percent SLA, that service can't be a single point of failure in the system.
- **Understand availability metrics** Following are two measures you should use to plan for redundancy and determine SLAs:
  - **Mean time to recovery (MTTR)** The average time it takes to

restore a component after a failure

- **Mean time between failures (MTBF)** How long a component can reasonably expect to last between outages

Recommend a high-availability solution for compute

Microsoft Azure global datacenters and underlying infrastructure are designed to deliver the highest redundancy and resiliency for an application running on Azure services. However, failures do happen. Therefore, the key to designing a reliable application in the cloud is to design applications to handle failures and minimize business disruptions gracefully.

In this section, you'll learn the recommendations to increase the availability of Azure VMs:

- **Single VM** Single VMs have an SLA offered by Azure. If you use premium storage for all operating system disks and data disks, you can get only 99.9 percent SLA.
- **Availability sets** These can help you increase the level of SLA from 99.9 percent to 99.95 percent. Availability sets protect a set of VMs from localized hardware failures, such as a disk or network switch, ensuring not all VMs are deployed on the same underlying hardware. Each virtual machine in the availability set is assigned an update domain and a fault domain by default. Each availability set can be configured with up to three fault domains and 20 update domains. Update domains indicate groups of virtual machines that can be rebooted simultaneously. For example, if you deploy 10 virtual machines in an availability set with three update domains, you have at least six VMs always available during planned maintenance.
- **Availability zones** These are unique physical locations within an Azure region. Every single zone in Azure is composed of one or more datacenters with independent power, cooling, and networking. The physical separation of availability zones within a region limits the impact on applications and data from zone failures such as large-scale flooding or other natural disasters that could disrupt the entire datacenter and the availability of resources. Availability zones help you increase SLA levels from 99.95 percent to industry best 99.99 percent

uptime.

More Info **Availability Zones Supported Regions**

Not every Azure region supports availability zones. You can find the list of supported Azure regions for availability zones in the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/availability-zones/az-region#azure-regions-with-availability-zones*.

- **Proximity placement groups (PPGs)** A proximity placement group is a logical grouping that ensures Azure compute resources are physically located in close proximity for low network latency between VMs. You can use PPGs with both availability sets and availability zones.
- **Virtual machine scale sets (VMSS)** To achieve redundancy, high availability, and improved performance, applications are distributed across multiple instances. Azure VMSS are used to create and manage a group of load-balanced VMs. The number of virtual machine instances can automatically scale (increase or decrease) on demand or per defined time schedules.

EXAM TIP

Virtual machine scale sets can be deployed in multiple availability zones to achieve resiliency and fault tolerance against regional failures.

EXAM TIP

Always place VMs in one availability set. A single availability set with two or more VMs helps to provide redundancy so that one VM is always up and running if a failure occurs.

Recommend a high-availability solution for non-relational data storage

Azure Storage provides several redundancy options to help ensure your data is available. Azure stores multiple copies of your data in Azure Storage to prevent unplanned disruptions. Redundancy ensures that your storage account fulfills the SLA for Azure Storage.

While deciding which redundancy option is best, you should consider the trade-offs between cost and durability. The factors that help determine which storage type you should choose include the following:

- How do you replicate your data on the primary site?
- If your data needs to be replicated to a second site, is it geographically distant from the primary site to protect against regional disasters?
- Does your application need read access to the replicated data in the secondary region if the primary region is no longer available?

As noted, Azure maintains multiple copies of your data stored in Azure Storage. Azure offers two options for Azure Storage, based on how data will be replicated throughout the primary region:

- **Locally redundant storage (LRS)** With LRS, data is replicated synchronously three times within a single physical location in the primary region. Because LRS provides local redundancy, it is the least expensive option, but it is not recommended for mission-critical applications that require better availability.
- **Zone-redundant storage (ZRS)** With ZRS, data is replicated synchronously across three Azure availability zones in the primary region. It is recommended that you use ZRS in the primary region for applications requiring high availability, and you should also replicate it in a secondary region.

For mission-critical applications requiring the best availability, you can also replicate data in your Azure Storage account to another region that is hundreds of miles away from the primary region. Your data is more durable when your Azure Storage account is replicated to a secondary region. You are covered even in the case of a complete regional outage or a disaster, even if the primary region is not recoverable.

Microsoft offers two options for Azure Storage that offer redundancy for your data to another region:

- **Geo-redundant storage (GRS)** With GRS, data is replicated synchronously three times within a single physical location in the primary region using LRS. Azure then moves an additional three copies of data asynchronously to a single physical location in the secondary region. You get enhanced redundancy with a total of six copies of data.
- **Geo-zone-redundant storage (GZRS)** With GZRS, data is replicated synchronously across three Azure availability zones in the primary region using ZRS. Azure then moves an additional three copies of data asynchronously to a single physical location in a secondary region. You get enhanced redundancy with a total of six copies of data.

If you compare GRS and GZRS, you will find the only difference is how data is copied in the primary region. There is no difference in replication to the secondary region. For both options, data is always replicated in the secondary region three times using LRS. This LRS redundancy in the secondary region protects the data against hardware failures.

For both GRS and GZRS, the secondary location data will not be available for read or write access unless you do a failover to the secondary region. If you need read access to data in the secondary location, you should go for read-access geo-redundant storage (RA-GRS). If you also need zone redundancy, go for read-access geo-zone-redundant storage (RA-GZRS).

When the primary region is unavailable, you can failover to the secondary region. Once the failover is completed, the secondary region will become a new primary region, and you will again be allowed to read and write data.

More Info **Failing Over to The Secondary Region**

For more information on failing over to the secondary region, see the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/storage/common/storage-disaster-recovery-guidance*.

Table 3-1 describes critical parameters for each redundancy option.

TABLE 3-1 **Redundancy parameters**

	LRS	ZRS	GRS/RA-GRS	GZRS/RA-GZRS
Percent durability of objects over a given year	At least 99.999999999 percent (11 9s)	At least 99.9999999999 percent (12 9s)	At least 99.99999999999999 percent (16 9s)	At least 99.99999999999999 percent (16 9s)
Availability SLA for read requests	At least 99.9 percent (99 percent for Cool access tier)	At least 99.9 percent (99 percent for Cool access tier)	At least 99.9 percent (99 percent for Cool access tier) for GRS	At least 99.99 percent (99.9 percent for Cool access tier) for RA-GRS
At least 99.99 percent (99.9 percent for Cool access tier) for RA-GZRS	Availability SLA for write requests	At least 99.9 percent (99 percent for Cool access tier)	At least 99.9 percent (99 percent for Cool access tier)	At least 99.9 percent (99 percent for Cool access tier)

More Info **Azure Storage Guarantees**

For more information about Azure Storage guarantees for durability and availability, see *https://azure.microsoft.com/support/legal/sla/storage/*.

Table 3-2 depicts the durability and availability of data in various scenarios, depending on which type of redundancy is in effect for your storage account.

TABLE 3-2 Durability and availability of data

Outage scenario	LRS	ZRS	GRS/RA-GRS	GZRS/RA-GZRS
A node within a datacenter becomes unavailable.	Yes	Yes	Yes	Yes
An entire datacenter (zonal or non-zonal) becomes unavailable.	No	Yes	Yes	Yes
A region-wide outage occurs in the primary region.	No	No	Yes	Yes
Read access to the secondary region is available if the	N	N	Yes (with	Yes (with

primary region becomes unavailable.	o	o	RA-GRS)	RA-GZRS)

Note **Account Failover**

Account failover is required to restore write availability if the primary region becomes unavailable. For more information, see *https://docs.microsoft.com/en-us/azure/storage/common/storage-disaster-recovery-guidance*.

Recommend a high-availability solution for relational databases

All applications need databases to store business data for the functionalities and features they provide to end-users. It's important that these apps, and their respective databases, be highly available and recoverable.

Following are the four major potential disruption scenarios that could affect the database's availability and the application:

- **Local hardware or software failures affect the database node** An example of such a scenario is disk-drive failure.
- **Data corruption or deletion caused by an application bug or human error** Such failures are application-specific and typically cannot be detected by the database service.
- **Datacenter-wide outage, possibly caused by a natural disaster** This scenario requires some level of geo-redundancy with application failover to an alternate datacenter.
- **Upgrade or maintenance errors** Unanticipated issues during planned infrastructure maintenance or upgrades might require rapid rollback to a previous database state.

Azure SQL Database from the Azure SQL product family provides several business continuity features that you can use to mitigate various unplanned scenarios. For example:

- Temporal tables allow you to restore row versions from any point in time.

- Built-in automated backups and Point-in-Time Restore enable you to restore a complete database within the configured retention period of up to 35 days in the past.
- You can restore a deleted database to the point at which it was deleted if the server has not been deleted.
- Long-term backup retention allows you to keep backups for up to 10 years. This is in limited public preview for SQL Managed Instance.
- Active geo-replication is another out-of-the-box feature that helps you create readable replicas and allows you to manually failover to any replica in case of a datacenter outage or application upgrade.
- An auto-failover group allows for the recovery of a group of databases in a secondary region if a regional disaster occurs or if there is a full or partial loss of an Azure SQL database or Azure SQL Managed Instance.

Chapter summary

- As part of your BC/DR plan, identify the RTOs, RPOs, and RLOs for your applications.
- ASR gives you the flexibility to failover to Azure if a disaster occurs and fails back to on-premises machines after the event is over.
- AKS is the most popular tool for deploying container workloads. To maximize uptime for AKS, plan for AKS clusters in multiple regions, and use geo-replication for container image registries.
- Azure Backup provides simple, secure, cost-effective solutions to back up your compute, databases, and unstructured data.
- Availability zones are distinctive physical locations within an Azure region made up of one or more datacenters, along with independent power, cooling, and networking. The physical separation of availability zones within a region limits the impact on applications and data from zone failures.
- Autoscaling is a process of dynamically allocating computing

resources to match performance requirements.

- Azure stores multiple copies of your Azure Storage data to protect against planned and unplanned incidents, including transient hardware failures, network or power outages, and substantial natural disasters.
- Azure Storage offers a durable platform and multiple geo-redundant storage options to ensure high availability. Storage account options with geo-redundant replication such as GRS and GZRS first synchronously replicate data in the primary region and then asynchronously replicate data to a secondary region at least a few hundred miles away.
- GZRS/RA-GZRS will provide you with a maximum availability and durability solution (but it is more expensive).

Thought experiment

Now it is time to validate your skills and knowledge of the concepts you learned in this chapter. You can find answers to this thought experiment in the next section, “Thought experiment answers.”

You have been hired to work as a Cloud Solution Architect for Contoso. You must design disaster recovery and high-availability strategies for your internally hosted applications, databases, and storage. Your company has a primary office in Seattle and branch offices in New York, Chicago, and Dallas. As part of this project, you plan to move to the cloud three on-premises applications that belong to different departments. Each application has a different requirement for business continuity:

- **Sales department** The application must be able to failover to a secondary datacenter.
- **HR department** The application data needs to be retained for three years. From a disaster recovery perspective, the application needs to run from a different Azure region with an RTO of 15 minutes.
- **Supply-chain department** The application must be able to restore data at a granular level. The RTO requirement is six hours.

You must recommend which services should be used by each department. While there could be multiple answers, choose the options that help minimize cost.

1. Which of the following would you use for the sales department?

   A. Azure Backup only

   B. ASR only

   C. ASR and Azure Migrate

   D. ASR and Azure Backup

2. Which of the following services would you recommend for the HR department?

   A. Azure Backup only

   B. ASR only

   C. ASR and Azure Migrate

   D. ASR and Azure Backup

3. Which of the following services would you recommend for the supply-chain department?

   A. Azure Backup only

   B. ASR only

   C. ASR and Azure Migrate

   D. ASR and Azure Backup

Thought experiment answers

This section contains the answers to the “Thought experiment” questions.

1. Which of the following would you use for the sales department?

   Answer B: ASR only

   Explanation You can use the ASR service to ensure that you can failover your application to a secondary region. The other options are incorrect because you need ASR to address the sales department’s requirement for the failover. You don’t need Azure Migrate because it should be used when you want to migrate VMs from VMWare VMs to

Azure VMs.

2. Which of the following services would you recommend for the HR department?

   Answer D: ASR and Azure Backup

   Explanation As stated in the requirements, you need to retain backups for three years, so you must use Azure Backup. You also need the ASR service to ensure that the application can run in another datacenter in case of a disaster. You need both Azure Backup and ASR. The other options are not adequate to meet the stated requirements.

3. Which of the following services would you recommend for the supply-chain department?

   Answer A: Azure Backup only

   Explanation As stated in the requirements, you need to be able to restore from any point in time in the past. So Azure Backup is what you use. Azure Backup automatically creates recovery points when subsequent backups are taken so that you run the restore operations from any point in time.

Chapter 4

Design infrastructure solutions

Azure provides a wide range of infrastructure services, such as compute, network, and application services. These infrastructure services are among the most consumed services by Azure customers around the globe. AZ-305 is an advanced-level exam, and you must thoroughly understand Microsoft’s infrastructure services so that you can use your skills and experience to design solutions on the Azure platform.

This chapter looks at various ways to design solutions on the Azure platform using compute, application, migration, and network services.

Skills covered in this chapter:

- Skill 4.1: Design a compute solution
- Skill 4.2: Design an application architecture
- Skill 4.3: Design migrations
- Skill 4.4: Design network solutions

Skill 4.1: Design a compute solution

A compute service is a hosting model to host and run your application on the cloud. This type of service provides processing power, memory, and local storage.

Compute is one of the fundamental building blocks of your workload. Microsoft Azure offers various compute services, such as virtual machines (VMs), Azure App Service, function apps, Service Fabric, and so forth, to cater to your needs.

As an Azure solutions architect, you must be mindful of choosing the right compute service to optimally balance your business need and your Azure spend. In this skill, you learn the various Azure compute offerings available to host your application and the differences between them so that you can make the right choice for your application scenario.

This section covers how to:

- Recommend a virtual machine–based compute solution
- Recommend an appropriately sized compute solution based on workload requirements
- Recommend a container-based compute solution
- Recommend a serverless-based compute solution

Recommend a virtual machine–based compute solution

There are various compute-based solutions available in Azure. These include Azure virtual machines (VMs), virtual machine scale sets (VMSS), and Azure Desktop services.

Azure virtual machines (VMs)

Azure VMs are an infrastructure as a service (IaaS) type of compute service. An Azure VM provides a virtual processor, memory, storage, and network interfaces, along with the operating system of your choice.

You can connect to a VM using the Remote Desktop Protocol (RDP) connection for Windows VMs or using SSH for Linux VMs. You can take full control of a VM to install all the required software and server configurations for your application. When you have full control of the VM, managing the VM is your responsibility, so you must handle backup and OS-patching activities.

You should use an Azure VM:

- When you must quickly migrate servers or applications from on-premises to Azure. This is called a *lift-and-shift*. When migrating a server from on-premises to Azure, it's also called a *rehost*.

- To migrate legacy applications that would be challenging to redesign, remediate, or deploy into Azure PaaS offerings.
- To deploy databases with features not supported in Azure PaaS—for example, SQL Server Database with the full database engine, SQL Server Integration Services (SSIS), SQL Server Reporting Services (SSRS), and SQL Server Analysis Services (SSAS).
- To deploy commercial off-the-shelf (COTS) applications that you cannot remediate and deploy into Azure PaaS services.
- When you need full control over the application server, including the operating system and services.
- When you need immediate access to a development or test environment for your applications. In this case, you can quickly provision an Azure VM and use its auto-shutdown feature to save costs. When your work is complete, you can delete any VMs you no longer need.

Virtual machine scale set (VMSS)

Azure offers another IaaS compute service that is a slight variation on standard VMs: virtual machine scale sets, which allow you to create and manage a group of identical, load-balanced VMs. This enables you to centrally manage, update, and configure the VMs simultaneously, in minutes, to provide highly available applications. The number of VM instances in a VMSS can automatically scale out or scale in (increase or decrease) in response to demand or based on a defined schedule.

Here's an example of how a VMSS works. Suppose you run a website that enables scientists to upload astronomy images for processing. If you were to duplicate the VM, you would normally need to configure an additional service to route requests between multiple instances of the website, but a VMSS could do that work for you.

Azure Virtual Desktop

Azure Virtual Desktop is a desktop and application virtualization service that runs on the cloud. Key features of Azure Virtual Desktop include the following:

- You can use it to set up a multisession Windows 11 or Windows 10 deployment that delivers a full Windows experience with scalability.
- It supports Microsoft 365 Apps for Enterprise and optimizes it to run in a multi-user virtual scenario.
- It provides Windows 7 virtual desktop with free extended security updates.
- You can access your existing Remote Desktop Services (RDS) and Windows Server desktop and apps from any computer.
- You can virtualize both desktop and apps.
- It provides a unified management experience for managing desktops and apps from different Windows and Windows Server operating systems.

Recommend an appropriately sized compute solution based on workload requirements

In Azure, VM sizing has a significant impact on your overall Azure spend, so you must be mindful of choosing the right size. When choosing your VM size and family, the general recommendation is to start small and evolve from there. Scaling a VM in the cloud is just a button-click away, so starting with a small, inexpensive VM makes sense. When you need more capacity, you can change the size and even the type of the VM to meet your scaling needs.

Right-sizing is one of the important steps of the capacity-planning process. You must match the VM instance size to your workload performance requirements at the lowest possible cost. You must also take actions to cut waste. This includes examining existing running VM instances and identifying opportunities to optimize or downsize without compromising performance requirements to lower costs.

In this section, you learn how to choose the right size of Azure compute solution (VMs) based on the workload you plan to deploy. Table 4-1 shows the available VM family types and sizes, along with recommendations for various workload types.

TABLE 4-1 Azure VM family types and sizes

Family Type	Size	Description	Sample workload type
General Purpose	A, B, Dsv3, Dv3, Dasv4, Dav4, DSv2, Dv2, Av2, DC, DCv2, Dv4, Dsv4, Ddv4, Ddsv4, Dv5, Dsv5, Ddv5, Ddsv5, Dasv5, Dadsv5	Balanced CPU-to-memory ratio	Ideal for dev/test environments, small to medium databases, and low- to medium-traffic web servers B (burstable) series: Best for small proof-of-concept applications that do not require consistent full CPU performance Ddsv5 series: Latest generation D family sizes recommended for general-purpose needs
Compute Optimized	F, Fs, Fsv2, FX	High CPU-to-memory ratio	Ideal for medium-traffic web servers, batch processing, web servers, analytics, and gaming
Memory Optimized	Esv3, Ev3, Easv4, Eav4, Ebdsv5, Ebsv5, Ev4, Esv4, Edv4, Edsv4, Ev5, Esv5, Edv5, Edsv5, Easv5, Eadsv5, Mv2, M, DSv2, Dv2	High memory-to-CPU ratio	Recommended for relational database servers, medium to large caches, and in-memory analytics M and Mv2 series: Best for SAP HANA databases Edsv5 series: Best for SQL Servers on VMs
Storage Optimized	Lsv2, Lsv3, Lasv3	High disk throughput and I/O	Ideal for big data, data warehousing, and large transactional databases
GPU	NC, NCv2, NCv3, NCasT4_v3, ND, NDv2, NV, NVv3, NVv4,	Specialized VMs	Used for heavy graphic rendering and video editing as well as for

	NDasrA100_v4, NDm_A100_v4		model training and inferencing (ND) with deep learning
High-Performance Compute	HB, HBv2, HBv3, HC, H	The fastest and most powerful CPU VMs	Ideal for special use cases such as electronic design automation, rendering, Spark, weather modeling, quantum simulation, and computational chemistry

More Info **VM Sizing Guideline**

For information about other factors to consider when sizing your Azure VM, see *https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/virtual-machine-recs*.

Recommend a container-based compute solution

Over the past few years, containerization has gained much traction. It has completely changed the IT industry, especially with organizations moving to the cloud with a multicloud strategy. With that in mind, the Azure platform has made it incredibly simple to leverage industry-leading container technologies to develop and deploy containerized applications.

In this section, you learn about the compute choices available in Azure to run containerized applications on Azure and when to choose one over the other. These include the following:

- **Azure Container Apps** This is a fully managed serverless container service to build and deploy cloud-native containerized applications. Examples of workload types you can deploy into Azure Container Apps include microservices, long-running background tasks, and event-driven applications.
- **Azure Container Instances (ACI)** This service enables you to spin up containers on demand without worrying about existing infrastructure such as Azure VMs. Azure manages all the underlying infrastructure mechanics transparently, allowing you to focus on building applications and deploying them in a readily available

containerized environment. ACI is best suited for apps that can operate in isolated containers and do not need orchestration. You can use ACI to deploy and run small event-driven applications, simple web apps, and small-batch-processing jobs, and pay only for those containers. ACI is a managed service, which means infrastructure management and operational overhead—such as upgrading and patching the underlying operating system or Azure VMs—are not your concern.

- **Azure Kubernetes Services (AKS)** Using this fully managed service, you can deploy and manage containerized applications with full-fledged container-orchestration capabilities. AKS eliminates the operational and maintenance overhead involved with managing your own Kubernetes deployments. AKS also handles critical Kubernetes tasks such as monitoring the health of underlying infrastructure and maintains the desired state and lifecycle of containerized applications, including autoscaling, monitoring the health of individual services, auto-discovery for interservice communication, and load balancing. The best part is that AKS is free. You pay only for the agent nodes within your clusters; you do not pay for the masters that control the AKS cluster.

EXAM TIP

ACI is recommended for small-scale applications and automation tasks. For enterprise-grade applications and microservices, you must use AKS.

Recommend a serverless-based compute solution

Microsoft Azure offers several serverless technologies to deploy and run your application code at scale without managing servers. These serverless technologies abstract the underlying infrastructure, so you can think less about servers and focus more on developing application code.

When you use serverless technologies, you need not worry about scaling hardware to meet increased demand, paying for hardware when it is not in use, or managing the availability of servers for planned maintenance. The cloud vendor takes care of all these tasks for any cloud-based infrastructure.

The most popular serverless compute options in Azure are as follows:

- **Azure Functions** This event-driven serverless compute service enables developers to run event-triggered code on-demand without provisioning the underlying VMs. Examples of events include HTTP requests, messages in the queue, and scheduled jobs. Azure Functions supports many popular programming languages—including C#, F#, Java, Python, JavaScript, and PowerShell—so you can build applications in the language of your choice. Azure Functions consumption plans are priced based on the number and duration of executions run.
- **Azure Logic Apps** This is a designer-first integration service that allows for a low-code/no-code approach to creating workflows to automate business processes and orchestrating tasks to integrate line of business (LOB) applications. Integration solutions include app integration, data integration, system integration, enterprise application integration (EAI), and business-to-business (B2B) integration. Pricing of Azure Logic Apps is based on the number of executions run and the types of connectors used.

> ***Note*** In a nutshell, Azure Functions is a serverless compute service, whereas Azure Logic Apps is a serverless orchestration service.

In some situations, you could use either of these services to solve a given business problem—although one might still be preferable to the other. For example, you could use Azure Functions to orchestrate business processes and workflows, but this would be complex and time-consuming. A better idea would be to employ Azure Logic Apps, which has a growing gallery of more than 200 built-in connectors to help you develop enterprise integration solutions in just a few clicks.

Table 4-2 lists various use cases and whether they call for the use of Azure Functions or Azure Logic Apps.

TABLE 4-2 Azure serverless compute options

Use case	**Soluti**

	on
Implementing an enterprise integration solution for B2B scenarios using built-in connectors (for example, to send email alerts using an Office 365 account when some specific event occurs)	Azure Logic Apps
Developing a net-new application or migrating existing, event-driven short-lived processes in a variety of languages such as C#, Python, Java, and so on	Azure Functi ons
Performing complex data-lookup and data-parsing operations from relational or No-SQL databases	Azure Functi ons
IT pros or business analysts who want to develop workflows graphically or using a visual designer	Azure Logic Apps

EXAM TIP

Azure Functions is stateless by default and does not maintain any states of data upon execution. If you need to maintain or store data between workflow processes and Function executions, you should use Azure Durable Functions.

Skill 4.2: Design an application architecture

This domain objective focuses on Azure services for designing modern and cloud-native applications. The AZ-305 exam expects you to have a good understanding of these services.

This section covers how to:

- Recommend a caching solution for applications
- Recommend a messaging architecture
- Recommend an event-driven architecture
- Recommend an application configuration management solution
- Recommend an automated deployment solution for your application

- Recommend a solution for API integration

Recommend a caching solution for applications

The cache is a data storage layer that stores a subset of frequently accessed data, typically for read-only workloads. That way, future requests for that data can be served up more quickly, with higher throughput and lower latency, than it would if it were accessed from the original data source. Caching is an important design consideration for any cloud-based or distributed application to improve performance, scalability, and availability for a better user experience.

Microsoft Azure platform offers two caching services:

- **Azure Cache for Redis** This is a fully managed cache service. It provides an in-memory data store and a low-latency, high-throughput data storage solution for modern applications.
- **Azure Content Delivery Network (Azure CDN)** Azure CDN is designed to deliver static content directly from Microsoft's own global CDN network. Global organizations use Azure CDN to increase user satisfaction by enhancing the performance and responsiveness of their cloud-hosted websites, mobile apps, and streaming media.

Table 4-3 lists various use cases and whether they call for the use of Azure Cache for Redis or Azure CDN.

TABLE 4-3 Azure caching service options

Use case	Solution
Accelerating the retrieval of static content for faster page loads and response times	Azure CDN
Distributing static web content for users around the world from the closest edge location	Azure CDN
Reducing latency associated with database query requests and keeping the data closer to the front-end web app	Azure Cache for Redis
Efficiently storing session data, such as user cookies, to handle demand with fewer back-end requests	Azure Cache for Redis

More Info **Caching Guidance**

Caching data effectively and efficiently is imperative. For more information, see the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/architecture/best-practices/caching*.

Recommend a messaging architecture

Cloud is all about speed and agility. These two advantages often require organizations to move away from the traditional N-tier monolith approach to re-architect their applications and adapt to a modern cloud-native design.

One of the modern cloud-native designs is microservices-based architecture. A microservices architecture yields benefits such as cost-efficient scaling, improved agility, independent deployment, and faster time to market compared to traditional N-tier architectures. But these benefits do bring increased complexity, as well as some challenges.

One imminent problem with the microservices architecture is inter-services communication. To address this problem, you need to incorporate a messaging-based architecture that orchestrates seamless communication among loosely coupled, distributed services without direct service-to-service integration.

Messaging-based architectures provide the following key benefits:

- Loosely coupled, distributed services can communicate with each other without violating the microservices fundamental of data sovereignty.
- Services can scale independently.
- They support several types of communication protocols that cater to a variety of business use cases, such as one-to-one, one-to-many, and many-to-many.
- Advanced messaging capabilities allow you to design complex systems when the order of the workflow among services is critical and duplication is unavoidable.

A message can be classified into the following two categories:

- **Command** Commands are messages that tell the subscriber to perform a specific task. A command expects at-least-once delivery. If a command is lost, it affects the entire workflow of a business transaction. With a command, the producer generally expects a response from the subscriber to acknowledge that the message has been received and processed.
- **Event** An event is raised by the producer in response to some change. The subscriber or consumer of the message is not necessarily expected to confirm receipt of the event to the producer.

Based on the requirements of these two message types, Microsoft Azure offers different services to implement the messaging architecture. The following sections describe these in more detail and offer guidance on when to choose one or the other for a given application scenario.

Azure queue services

Azure offers two queue services:

- **Azure Queue Storage** This is a queueing service within the Azure Storage infrastructure. Using Azure Queue Storage, the producer can push messages to the queue and the consumer can then consume the messages through the polling mechanism. Azure Queue Storage is highly scalable, guarantees at-least-once delivery, and can store millions of messages (with a message size limit of 64 KB per message).
- **Service Bus Queues** This queueing service offers enterprise messaging capabilities. These include queueing, a publish/subscribe (pub/sub) model (which allows you to fan out messages to multiple subscribers—very useful in enterprise application scenarios), and advanced integration patterns. These patterns are for cloud-native applications that require advanced queueing capabilities such as first-in-first-out (FIFO), dead-lettering, and duplicate detection.

EXAM TIP

If you have a use case in which subscribers can consume messages without polling the queue, you need at-most-once delivery, or you need to accommodate messages larger than 64 KB, then Azure Service Bus is the best fit.

Azure event services

Azure offers two event services:

- **Azure Event Grid** This is a managed event-routing service that relies on a pub/sub model to route information between systems. You can integrate other Azure services and custom third-party services with Azure Event Grid.
- **Azure Event Hubs** This is a big data pipeline solution designed for massive real-time streams of event data from various event producers —for example, IoT devices and GPS systems. Unlike Azure Event Grid, Event Hubs can receive millions of events per second and send them to consumers in real time.

Although these event services have a few similarities, each service is designed for particular scenarios. In some cases, however, Event Hubs and Event Grid can complement each other and can be used together.

More Info **Compare Event Hubs and Event Grid**

To learn more about the differences between Event Hubs and Event Grid, see the Microsoft documentation at *https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services*.

Recommend an event-driven architecture

An event-driven architecture is usually composed of two components:

- **Event producer** This component generates a stream of events.
- **Event consumer** This component listens for events.

Applications that are designed using an event-driven architecture generally use a pub/sub model. In this model, the event producers are decoupled from the event consumers.

Figure 4-1 shows an example of an event-driven architecture. In this architecture, users shop on an e-commerce website and place orders for their products. They can also use the site to return the purchased products if they do not like them and can track their refunds.

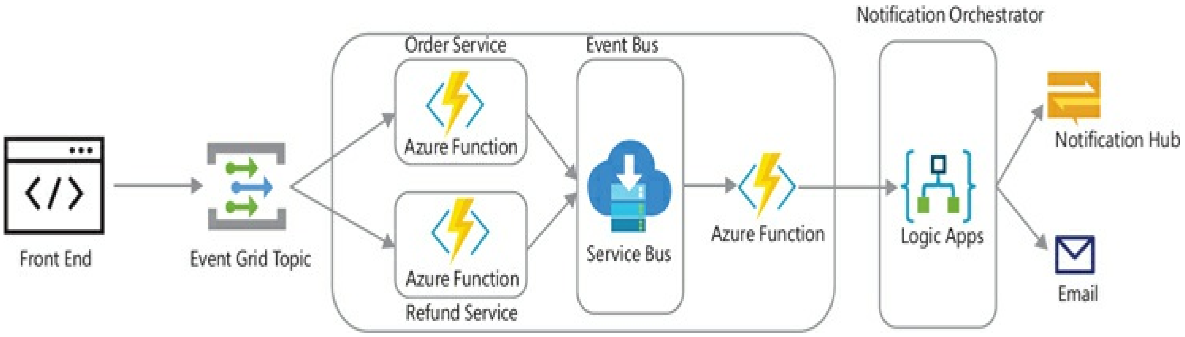


FIGURE 4-1 Sample e-commerce architecture using serverless service offerings

At a high level, this architecture consists of the following Azure services:

- **Event Grid** As you can see in Figure 4-1, users can place new orders or submit return requests for previously ordered products. These two separate events are then sent to the appropriate event consumers via Event Grid. In the sample architecture shown in Figure 4-1, order events are consumed by the order service and return request events are handled by the refund service.
- **Azure Functions** Azure Functions provides the event-driven, serverless compute capabilities in this architecture. It runs the business logic based on the event it receives from Event Grid.
- **Azure Service Bus** You use Azure Service Bus in this architecture for resiliency. You can queue messages in the Service Bus queue to handle faults and transient errors. You can also use Service Bus to handle other automated tasks, such as notifications.
- **Azure Logic Apps** You can use Azure Logic Apps to automate workflow tasks. Instead of writing code to implement these tasks,

however, you can do so using built-in connectors. Tasks can range from sending email notifications to integrating with external management applications. For example, in the architecture shown Figure 4-1, Azure Logic Apps could send order confirmation emails to users when new orders are placed. Logic Apps can also be used to create automated support tickets with ITSM systems such as ServiceNow when there are anomalies in the application code.

Recommend an application configuration management solution

Microservices-based applications pose a significant challenge to maintaining application- and environment-specific configurations in a distributed environment. Fortunately, there are several ways developers can deal with this challenge. For example, one best practice is to store configuration settings somewhere that is separate from the code runtime environment. Another is to use the Twelve-Factor app, which is a well-known methodology for building cloud-ready applications.

> *More Info* **Twelve-Factor App**
>
> The Twelve-Factor app is a well-known collection of patterns that relate to microservices architectures. It is also considered a requirement for cloud-native application architectures. To learn more, see *https://12factor.net/*.

Azure App Configuration can also help you address these challenges for cloud-based applications. The following use cases are good fits for using Azure App Configuration:

- **Containerized applications** These include AKS and Azure Service Fabric. It is recommended that you use Azure App Configuration to manage environment-specific deployments of these types of applications.
- **Serverless applications** Azure App Configuration can help you

instantly respond to changes in key-values, such as when a key-value is added or modified. These changes are less frequent; other changes in your scenario might require more immediate attention. In this case, Azure App Configuration events can be sent to Azure Event Grid to trigger event-based orchestration workflows such as updating the application configuration or triggering deployments.

Recommend an automated deployment solution for your application

The increasing adoption of container and serverless technology over the past few years has pushed organizations to adopt DevOps practices so they can capitalize on these technologies by delivering features faster than ever.

Using DevOps empowers you to automate infrastructure so you can build, test, deploy, and monitor applications without manual intervention. DevOps practices also enable organizations to achieve continuous delivery and continuous deployment in software development lifecycles.

The ability to ship features and functionalities repeatedly and more quickly makes it imperative to keep the infrastructure on which your code will run in a reliable state. This becomes even more important when you consider that in DevOps, your infrastructure becomes part of your iterative release cycles. So the operations and development teams must work together to manage infrastructure and application code through a unified process.

To enable this, and to get the best out of your cloud and DevOps practices, you need automation and orchestration solutions. Figure 4-2 shows a sample DevOps automation architecture that employs Azure DevOps and other Azure services to build an orchestration solution for automatic application deployment and maintenance.

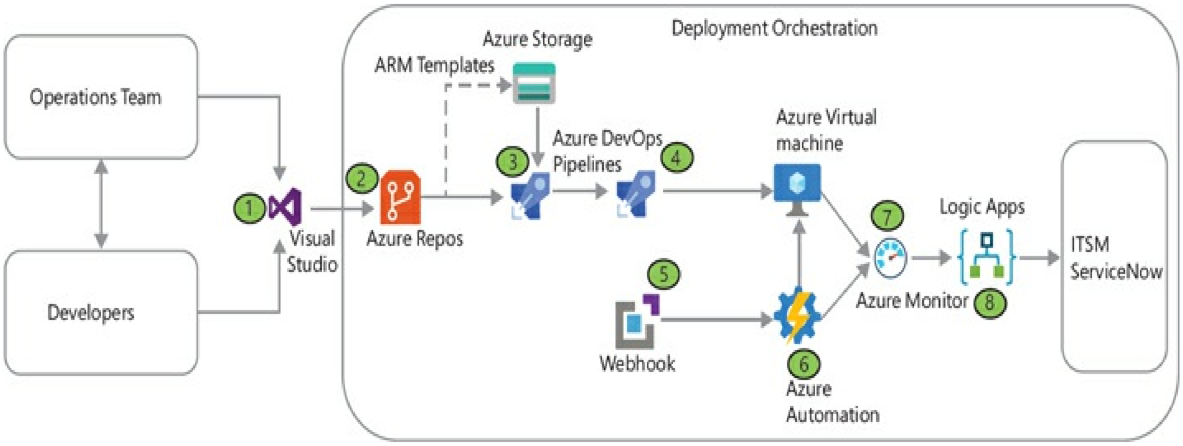


FIGURE 4-2 Automated orchestration solution for infrastructure and application deployment

At a high level, this architecture consists of the following Azure services:

- **Visual Studio** Developers and DevOps professionals use Visual Studio to develop application code and infrastructure as a code (IaC) templates such as Azure Resource Manager (ARM) templates and Bicep templates. They could also use Visual Studio IDE to commit their code to the appropriate Azure Repos tool.
- **Azure Repos** Azure Repos is a set of version control tools you can use to manage your application and infrastructure code.
- **Azure Pipelines** Continuous integration requires the use of a tool like Azure DevOps (AzDo) Pipelines, which helps you combine continuous integration (CI) and continuous delivery (CD) to test and deploy infrastructure and application code.
- **Webhooks** After Azure Pipelines provisions the infrastructure, it calls http webhooks to trigger an Azure automation runbook for the VM's desired state configuration.
- **Azure Automation** An Azure Automation runbook is triggered to run PowerShell scripts for the VM's desired state configuration.
- **Azure Monitor** Azure Monitor is a standard monitoring solution for infrastructure and applications to collect and analyze health, performance, and usage data.
- **Azure Logic Apps** You should use Azure Logic Apps with the built-

in ITSM connector to automate the operational task of notifying stakeholders of an anomaly in the application’s execution or infrastructure, and to automatically generate a service ticket using the organization’s IT service-management tool (for example, ServiceNow).

Regarding infrastructure deployment, in Azure the native deployment option for IaC uses ARM templates. You can develop ARM template files using JavaScript Object Notation (JSON) files or Bicep templates. Bicep is a new domain-specific language (DSL) designed to simplify the IaC authoring experience. Using Bicep templates is much easier than using JSON files.

Recommend a solution for API integration

With the emergence of cloud-based applications, microservices, and containerized applications, organizations must adopt an API-first approach to reach the cloud platform’s full potential. By using this approach, organizations become more agile—building applications quickly and delivering value to the business by exposing APIs to internal and external partners faster than ever. This approach also empowers developers to accelerate development by giving them full insight into the API’s internal implementation through API mocking and API documentation. Hence, it bridges the gap between the front-end and back-end teams.

Managing APIs is not easy when you adopt an API-first approach. You must secure the APIs, manage the various versions, and decide on a deployment methodology. You need a tool to act as a front-door gateway for all such capabilities.

Azure API Management (APIM) is one such tool. APIM is a front-door gateway service for publishing and managing REST APIs for enterprise-grade applications. Figure 4-3 shows an example of an API gateway strategy in which APIM is used to manage APIs securely and efficiently.

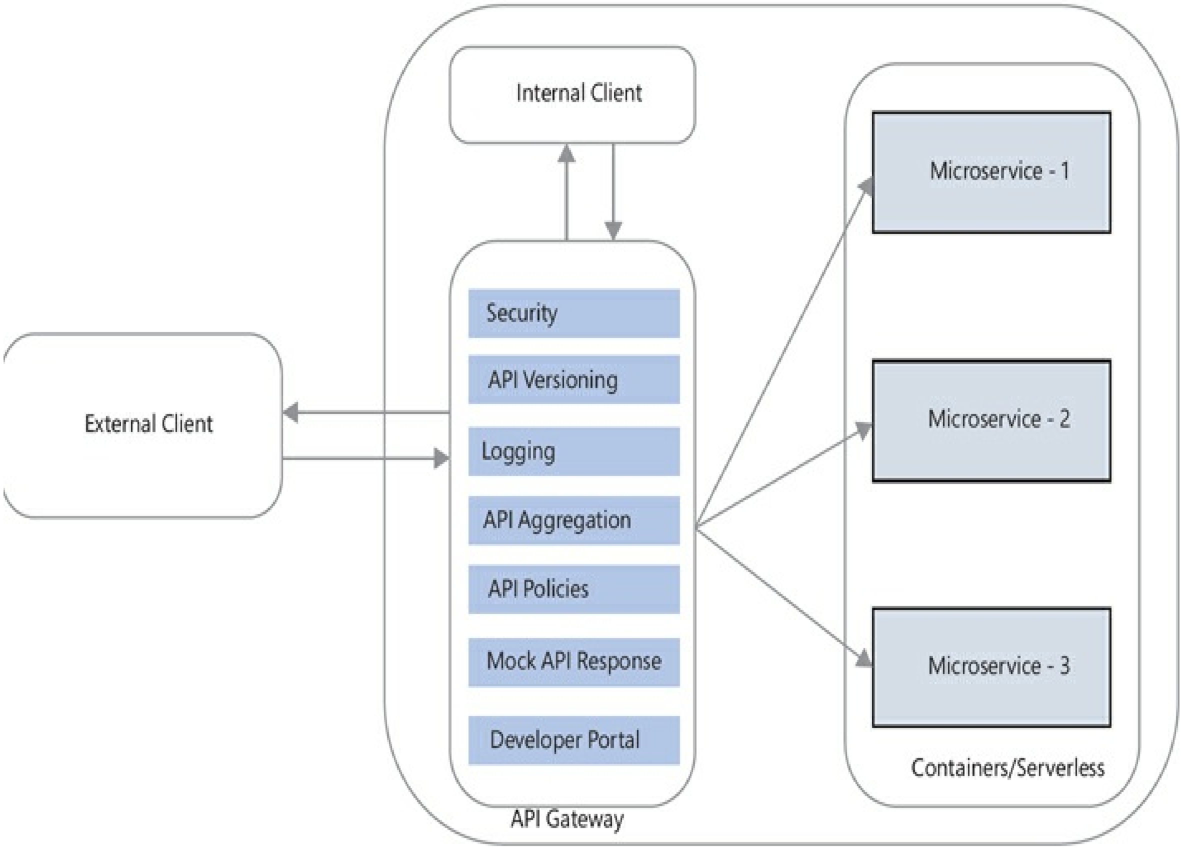


FIGURE 4-3 API gateway strategy for cloud-based applications

As you see in Figure 4-3, the APIM does not host the actual APIs. Rather, the APIs are hosted separately in the containers running inside services such as Azure Container Instance (ACI) and Azure Kubernetes Service (AKS). The APIM acts as a façade, or front door, for your APIs. In this way, APIM helps you to decouple your APIs, which in turn enables you to set API policies and other management options in Azure to securely manage and expose back-end systems.

Let us look at some out-of-the-box capabilities you can configure without much effort using APIM.

- **Security** It is imperative to secure access to your APIs to ensure that only authorized clients can access them. APIM supports the following mechanisms to secure published APIs without you needing to do the custom development:
  - Subscriptions keys

    - OAuth2.0
    - Client certificates
    - IP filtering (allow/deny)
- **API versioning** When you use APIM as a single gateway for all back-end APIs, you have the leeway to publish a new version of the same feature or functionality without affecting the existing clients. With APIM, you can deploy API versions to publish a new version of an API, safely test it, and deploy it to QA and production without affecting existing consumers.
- **Logging and monitoring** APIM has native integration with Azure Monitor. This allows for a unified experience when monitoring the health of your published APIs and the state of the API gateway. You can see how your APIs are being used by searching Azure Monitor activity logs for write operations (PUT, POST, and DELETE) performed on your API management services. Azure Monitor resource logs provide further insights into operations and errors for auditing and troubleshooting purposes.
- **API aggregation** You can use APIM to aggregate multiple individual requests into a single request to reduce chattiness between the client and the underlying APIs. This pattern mainly applies to microservices-based applications, where a single operation needs to call multiple microservices. APIM can dispatch calls to several back-end services, aggregate the results, and send them back to the client.
- **API policies** APIM offers robust built-in policies, which you can customize to change the behavior of a published API. Policies are automatically applied to the inbound request or outbound response of an API, so you have full control over how your APIs are exposed to internal and external customers.
- **Mock responses** This key APIM capability helps organizations accelerate development cycles. Essentially, you create a blank API and apply a policy for that API so that it returns a mocked response when called. This enables developers to implement and test the APIs, even if the back-end API is still being developed.
- **Developer Portal** APIM automatically generates a Developer Portal

and a fully customizable website with your API's documentation. It provides developers with the ability to discover APIs and learn how to use them.

EXAM TIP

You are very likely to see multiple-choice questions about APIM policies. To learn more about these, see *https://docs.microsoft.com/en-us/azure/api-management/api-management-policies*.

Skill 4.3: Design migrations

With the acceleration of cloud adoption, it is vital to understand how to migrate on-premises servers to Azure. As a Cloud Solution Architect, you are likely to face situations in which you need to plan and execute such migrations. This section provides an overview of the design options for migrating to Azure.

This section covers how to:

- Evaluate a migration solution that leverages the Cloud Adoption Framework for Azure
- Assess and interpret on-premises servers, data, and applications for migration
- Recommend a solution for migrating applications and VMs
- Recommend a solution for migration of databases
- Recommend a solution for migrating unstructured data

Evaluate a migration solution that leverages the Cloud Adoption Framework for Azure

Microsoft's Cloud Adoption Framework (CAF) offers proven guidance, best practices, tools, and detailed documentation to help cloud architects and IT stakeholders accelerate cloud adoption and achieve organizations' business goals. The CAF also provides guidance for migration.

The CAF's Migrate methodology recommends an iterative process, migrating one workload or group of small workloads per iteration. According to the CAF, this iteration consists of three phases:

1. **Assess** The first step is to assess your workloads to evaluate cost, modernization, and deployment tooling. This process focuses on validating or challenging assumptions made during earlier discovery and assessments by looking more closely at rationalization options. You should also assess workloads to ensure technical success after migration.
2. **Deploy** After your workloads are assessed, their existing functionality is replicated (or improved) in the cloud. Migration could involve a lift-and-shift approach or a rehosting in the cloud. Many assets that support these workloads will need to be modernized to capitalize on the benefits of the cloud.
3. **Release** After you replicate a workload's functionality, you can test, optimize, document, and release that workload for ongoing operations. As you do, it is critical to review the migrated workloads and hand them off as needed to governance, operations management, and security teams for support.

Microsoft has various tools and methods to handle different migration scenarios:

- **VMs** Windows Server, Linux Server, virtual desktops
- **Applications** ASP.NET, Java, PHP
- **Data** SQL Server, open-source databases
- **Hybrid** Azure Stack, VMware
- **Technology platforms** SAP, Kubernetes

More Info **Azure Migration Best Practices**

To learn more about the CAF's tools and templates for various migration scenarios and best practices, see Microsoft's documentation at *https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/resources/tools-templates* and *https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/migrate/azure-best-practices/*.

Microsoft's CAF team recommends the following steps to better prepare yourself for the migration journey:

1. **Migrate your first workload** Start with a small application that meets the following criteria to become familiar with the migration tools, processes, and best practices to streamline the remaining journey.
    - Non-mission-critical
    - Containing no sensitive data
    - Requiring a small number of servers
    - No dependency (or less dependency) on other applications
    - Requiring only one (or few) business unit's alignment for the migration
2. **Process improvement** The CAF's Migrate methodology recommends an iterative migration process, where after each iteration, you evaluate and mature various aspects of the process.

Assess and interpret on-premises servers, data, and applications for migration

As mentioned, migrating workloads such as servers, data, and applications generally spans three phases: assessing workloads, deploying workloads, and releasing workloads. Azure offers various tools for these different phases.

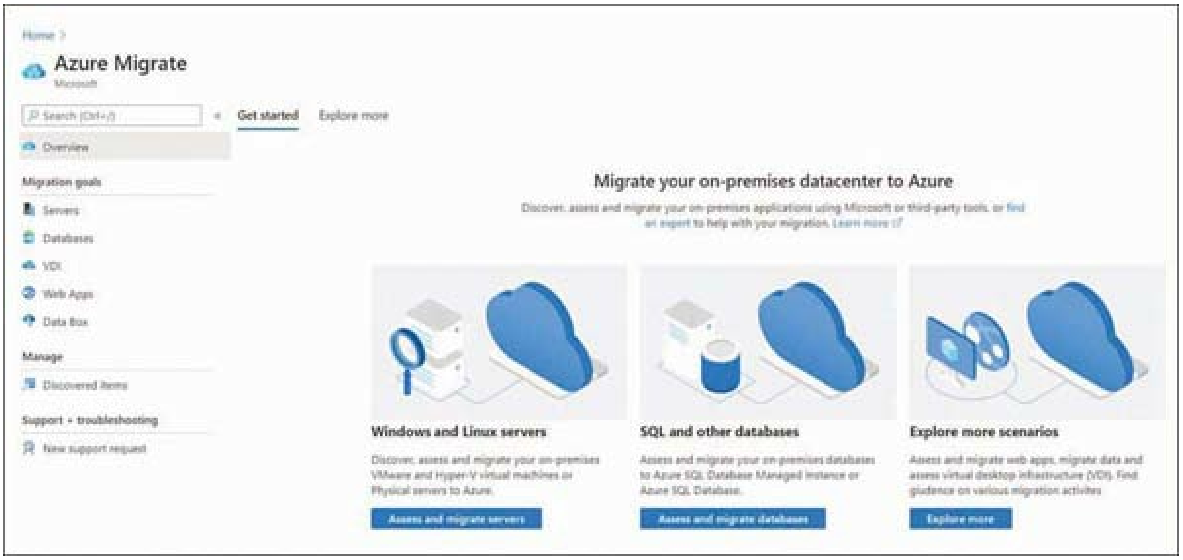
Azure Migrate

Azure Migrate is the native tool for assessing workloads and migrating to Azure. It assesses on-premises infrastructure, applications, and data prior to migration to Azure. It helps with the following tasks:

- **Discovery** You can use Azure Migrate for discovery on multiple vCenter servers by leveraging a VMware VM running the Azure Migrate Collector appliance. You can also use the same collector to discover VMs on different vCenter servers.
- **Assessing readiness** Azure Migrate allows you to perform a pre-migration assessment, regardless of whether your on-premises machines are suitable for running in Azure. In addition to performing feasibility analysis, assessing Azure readiness helps with:
- **Sizing recommendations** You can obtain sizing recommendations for Azure VMs based on the performance and utilization history of on-premises VMs.
- **Estimated monthly costs** You can generate an estimate of your Azure usage cost before migrating to Azure.
- **Identifying dependencies** Azure Migrate offers graphical features that enable you to visualize VM dependencies. This helps in creating optimal move groups for assessment and migration.

To access the Azure Migrate service, follow these steps:

1. Log in to the Azure Portal.
2. Enter **Azure Migrate** in the global search box and click **Azure Migrate** under the **Services** section in the list that appears to open Azure Migrate. (See Figure 4-4.)



Migrate your on-premises datacenter to Azure

Discover, assess and migrate your on-premises applications using Microsoft or third-party tools, or find an expert to help with your migration. Learn more

Discover, assess and migrate your on-premises VMware and Hyper-V virtual machines or Physical servers to Azure.

Assess and migrate your on-premises databases to Azure SQL Database Managed Instance or Azure SQL Database.

Assess and migrate web apps, migrate data and assess virtual desktop infrastructure (VDI). Find guidence on various migration activites

FIGURE 4-4 Azure Migrate

As a native tool, Azure Migrate offers a centralized hub for assessing and migrating various on-premises resources, including the following:

- **Servers** You can use Azure Migrate to assess on-premises servers and migrate them to Azure VMs or Azure VMware Solution (AVS).
- **Databases** Azure Migrate can assess on-premises databases and migrate them to Azure SQL Database or SQL Managed Instance.
- **Web applications** After assessing on-premises web applications, Azure Migrate can migrate them to Azure App Service using the Azure App Service Migration Assistant.
- **Virtual desktop** Azure Migrate can assess your on-premises virtual desktop infrastructure (VDI) and migrate it to Windows Virtual Desktop in Azure.
- **Data** Migrate massive amounts of data to Azure quickly and cost-effectively using a separate Azure service.

Azure Migrate Server Assessment tool

You can use the Azure Migrate Server Assessment tool to assess on-premises VMware VMs, Hyper-V VMs, and physical servers for Azure migration. This tool provides the following vital functions:

- **Azure readiness assessment** The Server Assessment tool checks whether on-premises machines are ready for migration to Azure.
- **Azure sizing estimation** The Server Assessment tool estimates sizing for Azure VMs sizing and the number of Azure nodes needed after migration.
- **Azure cost estimation** You can use the Server Assessment tool to obtain a cost estimate for Azure resources for existing on-premises workloads.
- **Dependency analysis** This identifies server-to-server dependencies and suggests optimization and grouping strategies for moving on-premises servers to Azure.

Using the Azure Migrate Server Assessment tool's dependency analysis feature can give you greater confidence when assessing VM groups to migrate. Dependency analysis also cross-checks various dependencies to help you avoid unexpected outages when you migrate to Azure. Behind the scenes, Azure Migrate leverages the Service Map solution in Azure Monitor to enable dependency analysis.

There are two ways to use the Azure Migrate tool to perform discovery:

- **Agent-based** An agent is used on all on-premises servers.
- **Agentless** No agent is used.

Table 4-4 summarizes the differences between agentless visualization and agent-based visualization.

TABLE 4-4 Agentless versus agent-based visualization

Requirement	Agentless	Agent-based
Agent	No agents are needed on machines you want to analyze.	Agents are required on each on-premises machine that you want to analyze.
Log	Not required.	Azure Migrate uses the Service Map solution in Azure Monitor for

Analytics		dependency visualization.
Process	A process captures TCP connection data on machines enabled for dependency visualization. After discovery, the process gathers data in 5-minute intervals.	Service Map agents installed on a machine gather data about TCP processes and inbound/outbound connections for each process.
Data	On the source machine, this data includes the server name, process, and application name. On the destination machine, this data includes the server name, process, application name, and port.	On the source machine, this data includes the server name, process, and application name. On the destination machine, this data includes the server name, process, application name, and port. Azure Migrate uses the Service Map solution in Azure Monitor logs for dependency analysis.
Visualization	You can view a dependency map of a single server for 30 days.	You can view a single server’s or a group of servers’ dependency maps for an hour’s worth of data.
Data export	You can download the last 30 days of data in a CSV format.	You can query data with Log Analytics.

Movere

Movere is a software as a service (SaaS) offering. Its agentless bots scan 1,000 servers per hour to capture everything in your IT environment and surface that information in a dynamic and customizable dashboard. The software also analyzes the data and highlights key insights to provide IT administrators with the visibility and control they need over their environments. Movere continuously learns your environment while eliminating duplicative data points to ensure users can access the most accurate, reliable, and actionable data.

> ***Note*** Microsoft acquired Movere and has made it available through the Microsoft Solution Assessment and Microsoft Cloud Economics Program.

Recommend a solution for migrating applications and VMs

After using Azure Migrate for your assessment, you can decide which of your workloads are good candidates for migration to Azure. Azure Migrate can also migrate VMWare VMs, Hyper-V VMs, and physical servers into Azure.

Agentless replication options are available for VMware VMs and Hyper-V VMs. These options orchestrate replication by integrating with the functionality supported by the virtualization provider.

Recommend a migration strategy

With the assessment complete, you must identify tools to move applications, data, and Azure infrastructure.

When you start planning for migration and perform migration assessment, you typically perform a migration strategy known as the *cloud rationalization* process, which evaluates workloads to determine the best way to migrate or modernize each workload in the cloud. The five R’s of migration dispositions are the most common options for cloud rationalization.

- **Rehost** Also known as a lift-and-shift migration, a rehost effort is a no-code option for migrating existing applications to Azure quickly and with minimal change to the overall architecture. With the rehost strategy, you can migrate an application as-is with some of the benefits of the cloud IaaS and without the risks or costs associated with code changes.

- **Refactor** Platform as a service (PaaS) options can reduce the operational costs associated with many applications. It is a good idea to refactor an application slightly to fit a PaaS-based model. Refactor essentially ties to the application development process of restructuring code to enable an application to deliver new business opportunities.
- **Rearchitect** In some cases, you might find a few aging applications that are not compatible with cloud providers because of some anti-patterns. In such cases, you might be better off rearchitecting before the transformation. In other cases, you could have a cloud-compatible application that is not cloud-native, and that might provide you with cost and operational efficiencies if you decide to rearchitect the solution into a cloud-native application. While rearchitecting, you can adopt resilient, independently deployable, highly scalable services in your architecture. Azure services can accelerate the process, scale applications with confidence, and manage applications with ease.
- **Rebuild** In some scenarios, the refactoring of an application can be too large to justify further investment. It is typical that an application has previously met a business’s needs but is now unsupported or misaligned with the current business processes. In this case, you must create a new codebase to eliminate technical debt and align it with the cloud-native approach.
- **Replace** Typically, solutions are implemented by using the best technology and approach available at that time. However, as time passes, you can find software SaaS alternatives to provide all the necessary functionality for the hosted application. With the “Replace” strategy, you replace legacy workloads with alternate solutions, effectively removing them from the transformation effort.

Recommend a migration tool

Azure Migrate is the native Azure service used for migration from within Azure and from on-premises sites to Azure. You can use Azure Migrate to orchestrate replication from an on-premises datacenter to Azure. When replication is set up and running, on-premises machines can be failed over to Azure, completing the migration.

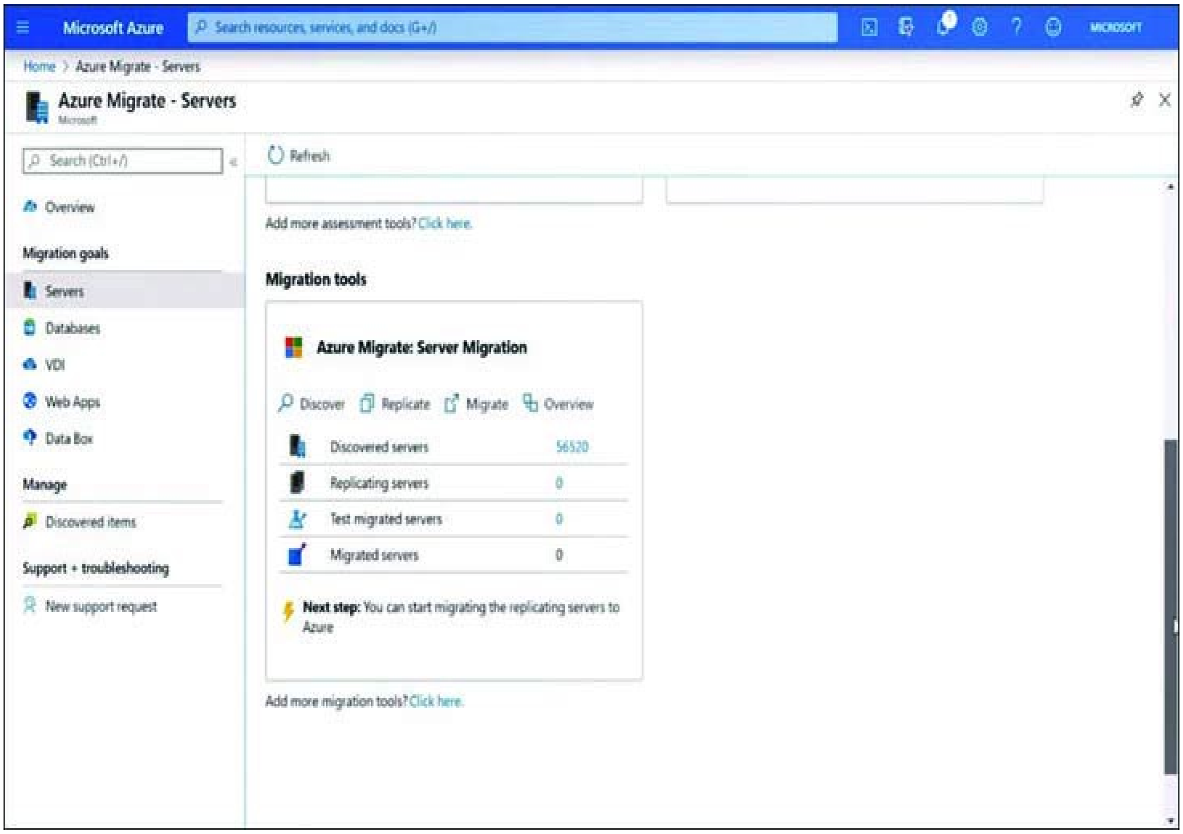
As mentioned, Azure Migrate provides a centralized hub to both assess and migrate to Azure from on-premises servers, infrastructure, applications, and data. The Azure Migrate hub includes the migration tools shown in Table 4-5 for migrating applications and VMs.

TABLE 4-5 Azure Migrate tools

Tool	Assess and migrate	Details
Azure Migrate Server Migration	Migrate servers.	Migrate VMware VMs, Hyper-V VMs, physical servers, other virtualized machines, and public cloud VMs to Azure.
Web App Migration Assistant	Assess on-premises web apps and migrate them to Azure.	Use the Azure App Service Migration Assistant to assess on-premises websites for migration to Azure App Service. Use Migration Assistant to migrate .NET and PHP web apps to Azure.

When you add the Azure Migrate Server Migration tool to your Azure Migrate dashboard—which carries over machines and insights from the assessment—you can initiate replication by clicking **Replicate** in the Azure Migrate: Server Migration window under Migration Tools, as shown in Figure 4-5.

Note The window in Figure 4-5 also shows discovered servers, replicating servers, test-migrated servers, and migrated servers.



Add more assessment tools? Click here.

Discovered servers	56520
Replicating servers	0
Test migrated servers	0
Migrated servers	0

Next step: You can start migrating the replicating servers to Azure

Add more migration tools? Click here.

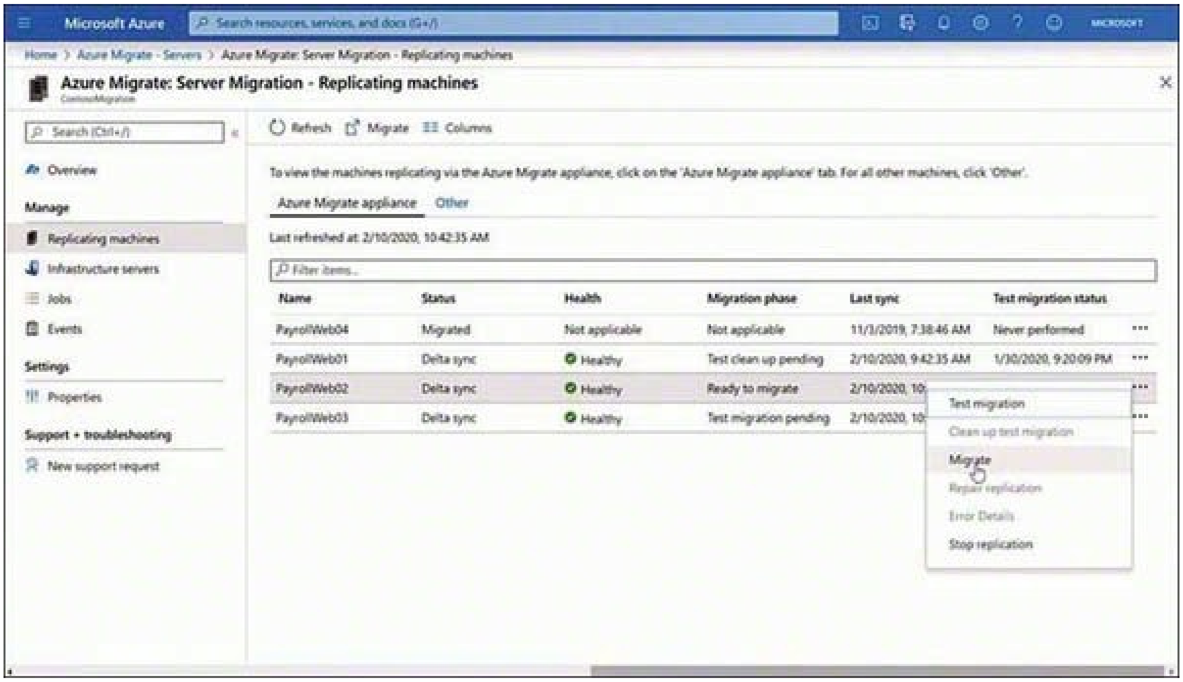
FIGURE 4-5 The main Azure Migrate screen

Azure Migrate replicates as many as 100 VMs simultaneously. If you need to replicate more, you can create multiple batches. Replication times vary based on the number and size of your VMs and on the connection speeds between your datacenter and Azure.

After all your targeted VMs are replicated to Azure, you can test them to ensure everything works before migrating them into production. This process involves running a prerequisite check, preparing for the test, creating a new test VM, and starting it.

When you are ready to migrate a VM to production, simply right-click the VM you want to migrate in the Replicating Machines screen in Azure Migrate and choose Migrate from the context menu that appears. (See Figure 4-6.) You’ll be prompted to shut down the VM to avoid data loss and to perform the final replication.

REAL WORLD We recommend you do this during off-peak business hours because the VM will go down for a few minutes.



Azure Migrate: Server Migration - Replicating machines

To view the machines replicating via the Azure Migrate appliance, click on the 'Azure Migrate appliance' tab. For all other machines, click 'Other'.

Last refreshed at: 2/10/2020, 10:42:35 AM

Name	Status	Health	Migration phase	Last sync	Test migration status
PayrollWeb04	Migrated	Not applicable	Not applicable	11/3/2019, 7:38:46 AM	Never performed
PayrollWeb01	Delta sync	Healthy	Test clean up pending	2/10/2020, 9:42:35 AM	1/30/2020, 9:20:09 PM
PayrollWeb02	Delta sync	Healthy	Ready to migrate	2/10/2020, 10:	
PayrollWeb03	Delta sync	Healthy	Test migration pending	2/10/2020, 10:	

FIGURE 4-6 Server Migration – Replicating Machines

In addition to starting a migration, you can use the Replicating Machines screen (refer to Figure 4-6) to perform a test migration (right-click the migration and choose **Test Migration**) and to stop replication (right-click the migration and choose **Stop Replication**). You can also use it to view all the servers being replicated and to check the status of the replication as it validates the prerequisites, prepares for migration, creates the Azure VM, and starts the Azure VM.

After the migration has taken place, you review the VM's security settings. It's recommended that you restrict network access for unused services by using network security groups (NSGs). You should also deploy Azure Disk Encryption to secure the disks from unauthorized access and data theft.

You should also consider improving the resilience of the migrated machines by doing the following:

- Adding a backup schedule that uses Azure Backup
- Replicating VMs to a secondary region using Azure Site Recovery
- Completing clean-up tasks for the remaining on-premises servers, including following:
    - Removing the servers from local backups
    - Removing the servers' raw disk files from the storage-area network (SAN) to free up space
    - Updating documentation related to the migrated servers to reflect their new IP addresses and locations in Azure

More Info **Azure Migration Best Practices**

Learn more about the best practices for migration to Azure here: *https://docs.microsoft.com/en-us/Azure/cloud-adoption-framework/migrate/Azure-best-practices/*.

Recommend a solution for migration of databases

Typically, any workload migration from on-premises to Azure involves one or more database migrations. Data is the heart of any application, and it is critical to migrate databases with minimal downtime and no data loss. As a Cloud Solution Architect, you must carefully choose a database-migration strategy and solution to migrate databases from on-premises to Azure.

Typically, the migration process involves the following three phases, which are discussed in the coming sections:

1. Pre-migration
2. Migration
3. Post-migration

Data migration tools

The following sections cover the following database-migration tools:

- Data Migration Assistant (DMA)
- Data Migration Service (DMS)
- SQL Server Migration Assistant (SSMA)

Note A fourth database-migration tool is Azure Migrate. This tool was discussed in detail in the section "Recommend a solution for migration of databases."

Data Migration Assistant (DMA)

Data Migration Assistant (DMA) is Microsoft's database assessment and migration tool. It is free to use. You can download, install, and execute it locally.

Table 4-6 lists the source and target databases supported by DMA.

TABLE 4-6 Source and target databases supported by DMA

Supported Database Sources	Supported Database Targets
SQL Server 2005	SQL Server 2012
SQL Server 2008	SQL Server 2014
SQL Server 2008 R2	SQL Server 2016
SQL Server 2012	SQL Server 2017 on Windows and Linux
SQL Server 2014	SQL Server 2019
SQL Server 2016	Azure SQL Database single database
SQL Server 2017 on Windows	Azure SQL Managed Instance
	SQL Server running on an Azure VM

DMA offers the following key capabilities:

- It detects compatibility issues—such as breaking changes, behavior changes, and deprecated features—that affect database functionality in Azure, and provides guidance on how to resolve them.
- It allows you to migrate database schema, users, server roles, SQL Servers, Windows logins, and data.
- You can use it to discover new features of the target database platform —such as those pertaining to performance, security, and storage—that will be beneficial after migration.

Data Migration Service (DMS)

Data Migration Service (DMS) is a fully managed service to help you easily migrate schema, data, and objects from multiple on-premises sources to Microsoft’s Azure platform.

The key capabilities of DMS are as follows:

- It migrates databases, including user objects, at scale with near-zero downtime.
- It makes the database-migration process simple and easy to understand and implement.
- It offers a standard pricing tier for small to medium business workloads (offline migration only). It also offers a premium pricing tier to support offline and online migrations (also called *continuous migration*) for business-critical workloads that require minimal downtime. The premium pricing tier is generally available.
- It is resilient and self-healing.
- You can use it to automate migration using PowerShell cmdlets.

SQL Server Migration Assistant (SSMA)

SQL Server Migration Assistant (SSMA) is Microsoft’s database migration tool for heterogeneous migration. It is freely available to download, install, and execute locally.

Table 4-7 lists the source and target databases supported by SSMA.

TABLE 4-7 Source and target databases supported by SSMA

Database Sources	Database Targets
Access	SQL Server 2012
DB2	SQL Server 2014
MySQL	SQL Server 2016
Oracle	SQL Server 2017 on Windows and Linux
SAP ASE	SQL Server 2019 on Windows and Linux
	Azure SQL Database
	Azure SQL Managed Instance
	Azure Synapse Analytics

SSMA's key capability is that it provides a simple and easy tool to automate the migration of databases from Oracle, MySQL, DB2, Microsoft Access, and SAP ASE to Azure.

EXAM TIP

Azure SQL databases block all inbound connections to SQL Servers and Databases using built-in firewalls. You must therefore configure the IP addresses of clients to connect to the server or databases.

Pre-migration

In the pre-migration phase, you collect the databases' inventory, assess these databases for potential incompatibilities, and plan for migration. If you plan to perform a heterogeneous migration—for example, migrating from Oracle to Azure SQL Database—you must convert the source database schema to match the target database.

The pre-migration phase has the following stages:

1. **Discover** This stage is required primarily if you plan to migrate databases in bulk, such as migrating all databases from an on-premises environment. In this stage, you scan your network and collect information about your on-premises databases, such as the server hostname, IP address, database version, and features in use. You can do this using tools like the Microsoft Assessment and Planning (MAP) Toolkit and Azure Migrate.
2. **Assess** To develop a migration plan and successfully migrate your databases, you must thoroughly assess the source database. A good tool for this is the Data Migration Assistant (DMA) tool. The idea is to identify gaps or incompatibilities between the source and target databases. Some objectives of the assessment are as follows:

    - Identifying migration blockers
    - Identifying breaking changes
    - Determining what efforts are required to fix migration blockers and breaking changes
    - Deciding whether to decommission unused databases
    - Deciding whether to consolidate databases
    - Analyzing the technical and business dependencies of the application/databases on other applications, databases, and services

> ***Note*** It is helpful to group databases with the same dependencies for a single wave of migration.

- Considering migration downtime

3. **Convert** When you perform a heterogeneous migration, you must convert the source schema to match the target database. For example,

when migrating from Oracle to SQL Server, you must convert the Oracle database schema to the SQL Server database schema. For this particular schema conversion, you can use SQL Server Migration Assistant (SSMA) for Oracle.

4. **Plan** Use the results of your assessment to plan the migration. When planning, you must make two important choices:

   - **Choose a target database** The target database you choose will be based on the attributes of the source database—things like database size, ease of management, scalability, availability, and features used (for example, SSIS, SSRS, or SSAS)—and on total cost of ownership (TCO).

Calculating TCO

Chapter 1, “Design identity, governance, and monitoring solutions,” covered TCO. Calculating the TCO provides several benefits:

- It gives you some idea of the costs and benefits of migrating your database to Azure.
- It enables you to compare the cost of using Azure and the cost of maintaining an on-premises datacenter.
- It allows you to gauge how much you will save by migrating to Azure (enabling you to make a better business case for the move when proposing it to stakeholders).
- It helps you choose the target database. For example, if you have budget constraints at your business unit level or organization level, you can select an appropriate target database based on application needs and budget.
- **Choose a migration method and tool** You can migrate databases online or offline. With the offline method, you accept some amount of downtime in the application during which you migrate the source databases to the target Azure databases. In contrast, the online method involves minimal downtime. Table 4-8 offers guidance on choosing a migration method based on acceptable downtime.

TABLE 4-8 Migration method versus acceptable downtime

Criticality	Acceptable Downtime	Migration Method	Migration Tool
High	Near-zero downtime	Transaction replication	SQL Server Management Studio (SSMS)
Medium	Small maintenance window	Online and offline migration	Azure Database Migration Service (DMS)
Low	Large maintenance window	BACPAC export/import	Azure Portal and SQL Server Management Studio (SSMS)

Migration

When you finish the pre-migration phase, you can start the migration phase. This phase has the following stages:

1. **Migrate schema** In the assessment stage of the pre-migration phase, you identified and rectified compatibility issues. So your database schema is ready to migrate to the target database. Before you migrate the schema to the target database, however, you must create that database. If the migration is homogeneous, you use the Data Migration Assistant (DMA) to migrate the schema; for heterogeneous migrations, you use SQL Server Migration Assistant.
2. **Migrate data** After migrating the schema, you can migrate your data. If your migration is homogeneous, you achieve this using DMA or DMS. For heterogeneous migrations, you use SQL Server Migration Assistant.
3. **Sync data** This step is needed if you have performed an online migration. When the migration is complete, you must sync incremental data. You do this using DMS.
4. **Complete the cutover** If you have performed an online migration, when you are finished with the full load and there are no pending changes for the incremental load, you use DMS to complete the cutover.

More Info **Migration Scenario and Tool**

For more information about migration scenarios and supported tools, see *https://docs.microsoft.com/en-us/Azure/dms/dms-tools-matrix*.

Post-migration

The post-migration phase involves the following steps:

1. **Remediate the application** When migration is complete, your application needs to connect to the target database. So you need to remediate the application to consume the target database. This includes changing the connection string to refer to the target database and changing the data access layer to use a target database–specific library.
2. **Test the application** With remediation complete, you're ready to test the application's functionality and performance using the new target database. To do this, obtain a copy of the source and target databases. Then perform functional validation tests and performance tests using both databases based on the defined scope, and compare the results.
3. **Optimize the database** In this stage, you fix any performance issues uncovered during the test stage. If you've migrated your database to Azure SQL Database, you should also fine-tune your database—for example, by restoring missing indexes. This will dramatically improve the performance of the application.

Recommend a solution for migrating unstructured data

Cloud adoption is gaining traction. Over the past decade, many organizations have moved to the cloud or are in the process of moving to the cloud.

One of the critical stages in cloud migration is the migration of data. Data migration is a three-step process:

1. Collect an inventory of servers and gather files and their configuration.

2. Transfer data.
3. Perform a cutover to the new servers.

Microsoft offers a variety of services to migrate data from on-premises to Azure. This section looks at data-migration solutions.

Storage Migration Service

Storage Migration Service is a graphical tool for migrating storage to Windows Server on Azure. This tool collects data from Windows and Linux servers and migrates it to a newer server or Azure VM. You can also use it to maintain the server's identity in the target environment so that apps and users can access it without changing their links or paths.

Key features of Storage Migration Service are as follows:

- It provides a user interface with a graphical workflow.
- It collects inventory of multiple servers and their data.
- It's scalable, consistent, and fast.
- It can manage multiple migrations using Windows Admin Center.

Azure Data Box

Azure Data Box is a family of products designed to transfer massive amounts of data. These products are as follows:

- Azure Data Box
- Azure Data Box Disk
- Azure Data Box Heavy
- Data Box Edge
- Data Box Gateway

The first three products in the preceding list—Azure Data Box, Azure Data Box Disk, and Azure Data Box Heavy—transfer data offline by shipping disks/appliances to Microsoft data-centers. These products are suitable for a one-time initial bulk transfer or periodic uploads. Table 4-9 outlines the main features of these products.

TABLE 4-9 **Azure Data Box products**

Azure Data Box	Azure Data Box Disk	Azure Data Box Heavy	
Total devices per order	1	Up to 5	1
Total capacity	100 TB	40 TB	1 PB
Usable capacity	80 TB	35 TB	800 TB
Supported Azure Storage services	Azure Block Blob, Page Blob, Azure Files, Managed Disk	Azure Block Blob, Page Blob, Azure Files, Managed Disk	Block Blob, Page Blob, Azure Files, Managed Disk
Interface	1x1/10 Gbps RJ45, 2x10 Gbps SFP+	USB/SATA II, III	4x1/10 Gbps RJ45, 4x40 Gbps QSFP+
Encryption	AES 256-bit	AES 128-bit	AES 256-bit

In contrast, Azure Data Box Edge and Azure Data Box Gateway are online data-transfer tools. Azure Data Box Edge is a hardware appliance provided by Microsoft to be placed on-premises. It acts as a cloud storage gateway that links the on-premises resources to Azure Storage. It caches data locally and then uploads it to Azure Storage.

Azure Data Box Gateway is a virtual appliance deployed in an on-premises virtualized environment. You can write data locally using the NFS and SMB protocols; this device then uploads the data to Azure Storage.

Azure File Sync–based migration to hybrid file server

A hybrid file server enables you to share files across multiple locations and securely store data in centralized cloud storage. You can use Azure File Sync to seamlessly synchronize your files between your local server and Azure Files. The migration process consists of the following phases:

1. Identify the required number of Azure file shares.
2. Provision an on-premises Windows Server.
3. Provision the Azure Storage Sync service.

4. Provision an Azure Storage account.
5. Install the Azure File Sync agent.
6. Configure Azure File Sync on the on-premises Windows server.
7. Use RoboCopy to copy the files.
8. Perform the cutover.

These steps are explained in more detail in the sections that follow.

Identify The Required Number of Azure File Shares

To synchronize your local files to Azure file share, you need a Windows server. A single Windows server (or cluster) can sync up to 30 Azure file shares. If you are planning a 1:1 mapping between the on-premises share to the Azure file share, you need a single Windows server. If you have more than 30 local shares, you need more than one Windows server. You can group your local shares and store the data into one Azure file share.

Azure File Sync supports up to 100 million items (files and folders) per Azure file share, but the best practice is to have 20–30 million in a single share. If your local share contains more than 30 million items, it is recommended that you split this data into multiple Azure file shares.

Azure file shares are provisioned within a storage account. Hence, the storage account is a scale target for IOPS and throughput. Also, there are additional IOPS and throughput limits on Azure file shares.

More Info **Azure Files and Storage Account Limits**

For Azure files and storage service limits (such as throughput and IOPS), see *https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/azure-subscriptionservice-limits*.

More Info **Microsoft-Provided Mapping Templates**

To arrive at the number of Azure file shares mapping with your local shares, consider the preceding points, and perform a mapping exercise using a Microsoft template at *https://download.microsoft.com/download/1/8/D/18DC8184-E7E2-45EF-823F-F8A36B9FF240/Azure%20File%20Sync%20-%20Namespace%20Mapping.xlsx*.

Provision an On-Premises Windows Server

Provision a Windows Server 2019 or Windows Server 2012 R2 on-premises based on the mapping completed in the previous step using the Microsoft-provided mapping template. You can also use a Windows Server failover cluster instead of a single server.

Provision the Azure Storage Sync Service

Provision the Azure Storage Sync service in the Azure region closest to your location. Also, use the same region for Azure Storage.

Provision an Azure Storage Account

Provision an Azure Storage account in the Azure region closest to your location. This should be the same region as the one used for the Storage Sync service. Refer to the mapping sheet referenced in the section “Identify the required number of file shares” to determine how many storage accounts to provision.

Install the Azure File Sync Agent

To install the Azure File Sync Agent, perform the following steps:

1. Disable Internet Explorer Enhanced Security Configuration.
2. Install the following PowerShell module:

   ```
   Install-Module -Name Az -AllowClobber
   Install-Module -Name Az.StorageSync
   ```

3. Install the Microsoft Sync Agent.

Configure Azure File Sync on the On-Premises Windows Server

Follow these steps to configure Azure File Sync on the Windows server:

1. In the Azure Storage Sync service, create a new sync group for each Azure file share. To do so, in Azure Storage Sync service, click **Sync Group**. Then enter the sync group name, subscription, storage account, and Azure file share.
2. Select the newly created sync group and click **Add Server Endpoint.**
3. Enter the required information—the registered server, path, cloud tiering, volume of free space, and initial download mode—to create the server endpoint.

Use Robocopy to Copy the Files

Using RoboCopy, copy files from your local shares, NAS appliance, and Linux server to the Windows server already configured with Azure File Sync.

Perform the Cutover

To perform the cutover, follow these steps:

1. After you run RoboCopy to copy your files, run it again to copy the new changeset that occurred after the last run.
2. Take your source file location offline or change ACLs so users cannot modify or add new files.
3. Create a share on the Windows Server folder and change the DFS-N deployment to point to it.

EXAM TIP

Azure CDN supports serving static files only from Azure Blob Storage. If you have a use case to transfer static files from network attached storage (NAS) file shares to be served from a CDN for performance improvement, you should use AzCopy to transfer files from NAS file shares to Azure Blob Storage.

Skill 4.4: Design network solutions

With a spaghetti of cables running through the datacenter, and with the massive amount of networking gear such as ports, connectors, plugs, routers, and switches to manage, understanding a traditional datacenter network can be daunting. Fortunately, the basic principles of cloud networking architecture are straightforward.

As an Azure Solution Architect taking the AZ-305 exam, you need to understand Azure networking services to set your foundation. It is the glue between most of the Azure resources you will use for your solutions. This section examines various Azure networking services and discusses how to design a network architecture so you can recommend the right solutions.

This section covers how to:

- Recommend a network architecture solution based on workload requirements
- Recommend a connectivity solution that connects Azure resources to the internet
- Recommend a connectivity solution that connects Azure resources to on-premises networks
- Recommend a solution to optimize network performance for applications
- Recommend a solution to optimize network security
- Recommend a solution for load balancing and traffic routing

Recommend a network solution based on workload requirements

Network topology is a critical element of enterprise-scale architecture because it defines how applications can communicate with each other. This section explores topology concepts for Azure enterprise deployments.

There are three key concepts:

- Azure virtual networks

- Hub-and-spoke network topologies
- Azure Virtual WAN topologies

Azure virtual networks

Azure virtual networks (VNets) are foundational building blocks for most Azure workloads. Azure VNets enable many Azure resources—such as VMs, VMSS, the App Service environment, App Service, and Azure Functions with VNet integration and Kubernetes clusters—to communicate with each other securely via on-premises networks and on the internet.

Key capabilities of Azure VNets include the following:

- They provide secure communication for Azure resources to communicate with each other.
- You can configure endpoints on VNets for services that require internet communication.
- A VNet provides logical isolation for your Azure subscription.
- You can implement multiple VNets within Azure regions in your subscriptions.
- VNets offer isolation from other VNets.
- You can use private and public IP addresses defined in RFC 1918 and expressed in CIDR notation.
- If you use your public IP address as the VNet's address space, this public IP is not routable from the internet and remains private from an accessibility standpoint.
- You can connect two VNets by using virtual network peering. When two VNets are peered, resources in one VNet can connect to resources in the other VNet.
- Peered VNets can be in the same region or in different regions.

By default, Azure learns routes from on-premises over ExpressRoute, routes for all peered VNets, and a default route to the internet. However, Azure also allows customers to override these system routes with user-defined routes (UDRs). You can assign UDRs at the subnet level.

Hub-and-spoke network topology

A hub-and-spoke network topology isolates workloads while sharing services, such as identity, connectivity, and security. The hub VNet, as its name suggests, is a central point of connectivity. Spoke VNets connect to the hub VNet using VNet peering or global VNet peering.

Typically, you would deploy network security gear, such as Azure Firewall or third-party firewall appliances, in the hub. Also, shared services are typically deployed in the hub or as separate spokes peered with the hub. In contrast, you would deploy individual production and non-production workloads as spoke VNets. You can provision an ExpressRoute gateway in the gateway subnet. If you do, however, you cannot deploy anything else in the gateway subnet.

In a hub-and-spoke topology, all the spoke-to-spoke communication transits through the hub. You must set the firewall (Azure Firewall or a network virtual appliance, or NVA) as the next hop in any UDRs attached to subnets in spoke VNets. The UDR overrides system routes that would otherwise send all traffic destined for an on-premises network through the gateway. With the UDR, you set your virtual appliance as a next-hop address.

Figure 4-7 shows the implementation of a hub-and-spoke network topology. The spoke VNets typically host a management subnet and at least one workload subnet each. The hub VNet hosts core networking and security solutions in subnets dedicated for gateway, management, firewalls, Active Directory, and so on. You should use VNet peering between hub and spoke VNets, and ExpressRoute circuit private peering connecting to an on-premises gateway and an ExpressRoute gateway in the hub VNet.

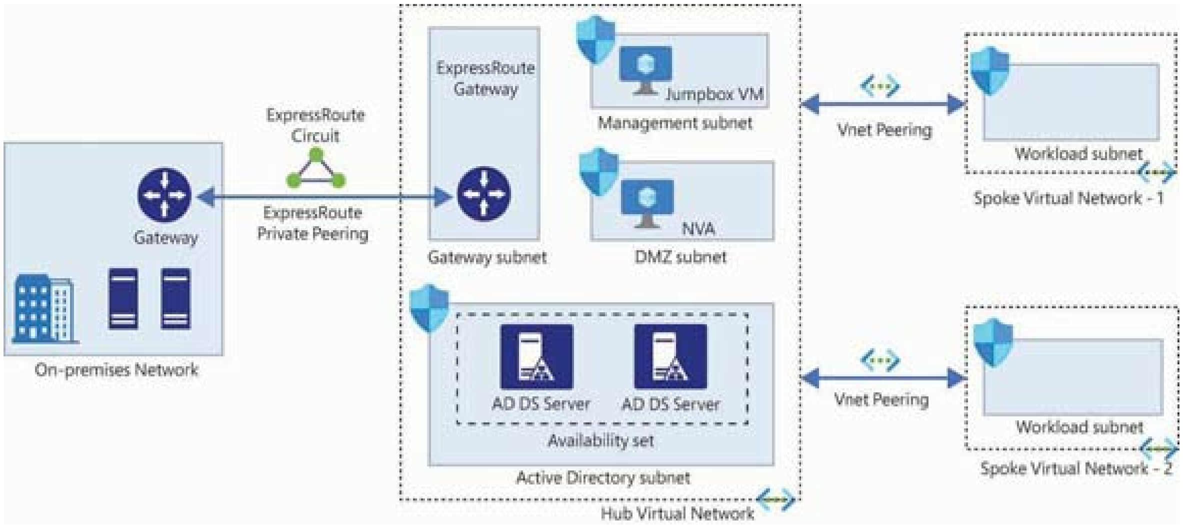


FIGURE 4-7 Hub-and-spoke topology

More Info **Hub-and-Spoke Topology**

To learn more about the hub-and-spoke topology, see *https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/hybrid-networking/hub-spoke*.

Hub-and-spoke topologies have the following design considerations:

- Implementing a hub-and-spoke topology in Azure centralizes standard services, including connections to on-premises networks and firewalls.
- The hub VNet acts as a central point of connectivity and hosts shared services used by workloads hosted in spoke VNets.
- Enterprises typically use a hub-and-spoke configuration.
- Spoke VNets isolate workloads. Spoke-to-spoke communication goes through the hub, and a centralized firewall has the visibility and can control traffic flow. Each workload can include multiple tiers.
- Azure lets you provision hub-and-spoke VNets in the same or different resource groups or subscriptions. You can also have spoke VNets in different subscriptions than the hub. Moreover, the subscriptions can be associated with the same or a different Azure Active Directory (Azure AD) tenant.

- The hub-and-spoke topology allows for decentralized management of each workload while sharing services maintained in the hub network.

Use a traditional Azure network topology if these are your requirements:

- You intend to deploy resources across multiple Azure regions.
- You have a small number of branch locations per region.
- You need fewer than 30 IPSec tunnels.
- You require full control.
- You need granularity for configuring your Azure network.

Azure Virtual WAN topology

You can use a virtual WAN to meet large-scale, multi-site interconnectivity requirements. One way to implement a virtual WAN is to use Azure Virtual WAN. Azure Virtual WAN is a Microsoft-managed networking solution that provides end-to-end global transit connectivity.

Azure Virtual WAN hubs eliminate the need to configure network connectivity manually. For example, with Azure Virtual WAN hubs, you are not required to configure UDRs or NVAs for hub-and-spoke connectivity. (You can use NVAs with Azure Virtual WAN if you require NVAs in your architecture, however.) Because Azure Virtual WAN is a Microsoft-managed service, it reduces overall network complexity and modernizes your organization’s network.

Following are the design considerations for Azure Virtual WAN:

- Azure Virtual WAN simplifies end-to-end network connectivity within Azure and cross-premises by creating a hub-and-spoke network architecture with a Microsoft-managed hub. The architecture can span multiple Azure regions and multiple on-premises locations (any-to-any connectivity) out of the box. Figure 4-8 shows the global transit network with Azure Virtual WAN.

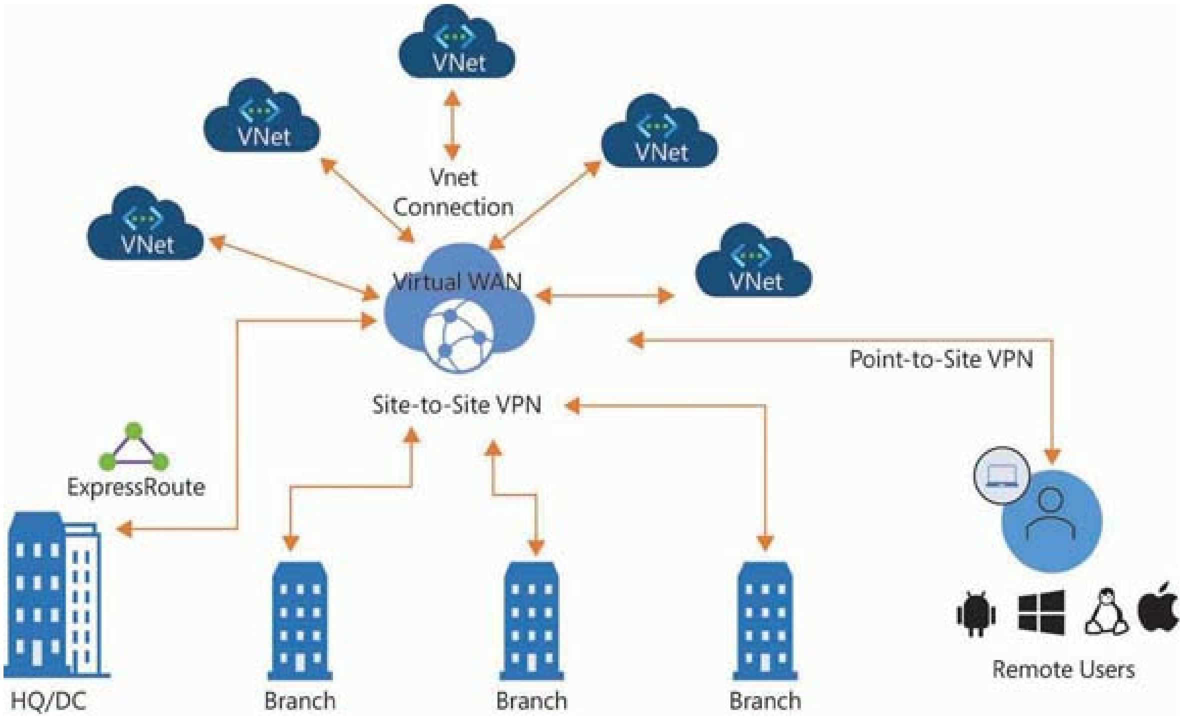


FIGURE 4-8 Global transit network with Azure Virtual WAN

- Azure Virtual WAN hub VNets are locked down. You cannot deploy any resources in the WAN hub VNet except VNet gateways (point-to-site VPN, site-to-site VPN, or Azure ExpressRoute), Azure Firewall through Firewall Manager, and route tables.

Azure Virtual WAN increases the number prefixes from Azure to on-premises via ExpressRoute private peering—from 200 to 10,000 prefixes per Azure Virtual WAN hub. The 10,000-prefix limit includes prefixes advertised over site-to-site VPNs and point-to-site VPNs.

Microsoft recently announced general availability (GA) for Azure Virtual WAN hub-to-hub connectivity and network-to-network transitive connectivity (within and across regions) features. Azure Virtual WAN transitive connectivity, made possible because there is a router in every virtual hub, supports the following:

- VNet to branch
- Branch to VNet
- Branch to branch

- VNet to VNet (same region and across regions)

Here are a few more key points about Azure Virtual WAN:

- Every virtual hub router supports up to 50 Gbps aggregate throughput.
- Azure Virtual WAN integrates with a variety of SD-WAN providers.
- You must use ExpressRoute circuits with the premium add-on, and they should be from an ExpressRoute Global Reach location.
- You can scale VPN gateways in Azure Virtual WAN up to 20 Gbps and 20,000 connections per virtual hub.
- Azure Firewall Manager allows the deployment of Azure Firewall in the Azure Virtual WAN hub.

Azure Virtual WAN is a recommended solution for new global network deployments in Azure when you need global transit connectivity across multiple Azure regions and various on-premises locations. Figure 4-9 shows an example of global deployment with datacenters spread across Europe and the United States and many branch offices across regions. The environment is connected globally via Azure Virtual WAN and ExpressRoute Global Reach.

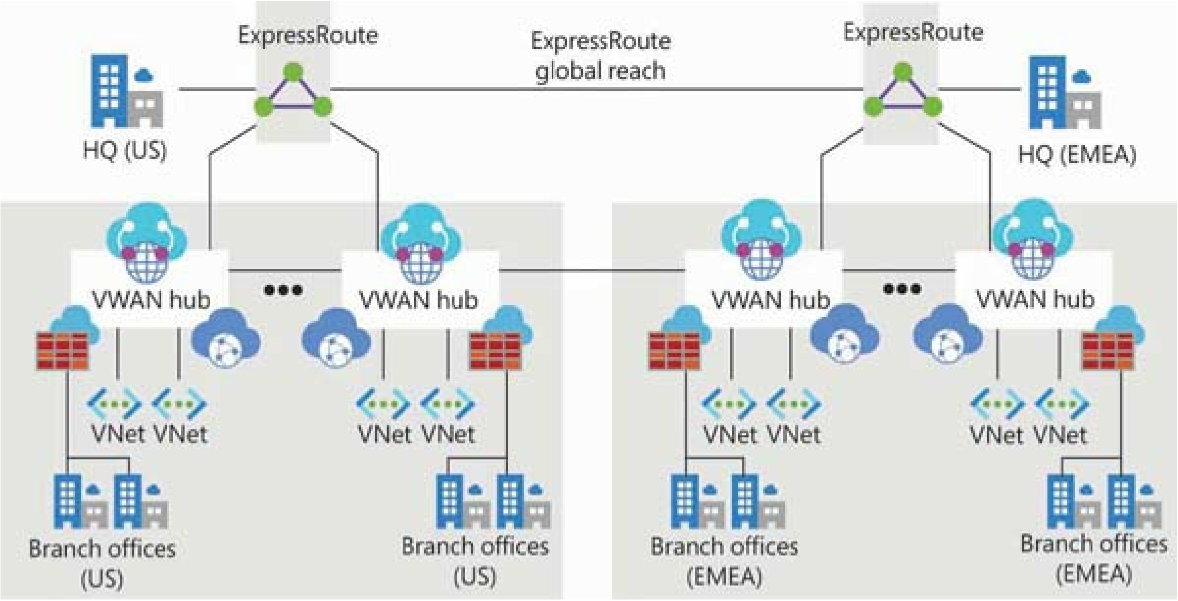


FIGURE 4-9 Global connectivity using Azure Virtual WAN and ExpressRoute global reach

Azure Virtual WAN is also recommended as a global connectivity resource. You can use one or many Azure Virtual WAN hubs per Azure region to connect multiple landing zones across Azure regions via local Azure Virtual WAN hubs.

Following are a few design recommendations for implementing an Azure Virtual WAN solution:

- Connect Azure Virtual WAN hubs with on-premises datacenters using ExpressRoute.
- Deploy required shared services such as DNS or Active Directory domain controllers in a dedicated landing zone. Be aware that you cannot deploy such shared resources in the Azure Virtual WAN hub VNet.
- You can connect branches and remote locations to the nearest Azure Virtual WAN hub using site-to-site VPN or branch connectivity to a virtual WAN through an SD-WAN partner solution.
- You can connect users to the Azure Virtual WAN hub through a point-to-site VPN.
- You should follow the “traffic within Azure should stay in Azure” principle. With this solution, communication between Azure resources across regions occurs over the Microsoft backbone network.
- Azure Firewall in an Azure Virtual WAN hub helps with east–west and south–north traffic protection.
- If you require third-party NVAs for east–west or south–north traffic protection and filtering, you can deploy the NVAs in a separate VNet, such as a shared VNet. You can then connect this shared VNet to the regional Azure Virtual WAN hub and the landing zones that need access to the NVAs.
- You need not build a transit network on top of a virtual WAN. The Azure Virtual WAN solution itself satisfies transitive network topology requirements. It would be redundant and increase complexity.
- Do not use existing on-premises networks such as multiprotocol label switching (MPLS) to connect Azure resources across Azure regions. Azure networking technologies support the interconnection of

resources across regions through the Microsoft backbone.

Comparing your options

Multiple products and services provide various networking capabilities in Azure. As part of your networking solution design, you should compare your workload requirements to the networking use cases in the Table 4-10.

TABLE 4-10 Azure networking use cases and solution options

Networking use case	Solution options
Networking infrastructure to connect everything	Azure Virtual Network
Inbound and outbound connections and requests to applications or services	Azure Load Balancer Application Gateway Azure Front Door
Securely use the internet to access Azure Virtual Network	High-performance VPN gateways
Ultra-fast DNS responses and ultra-high availability for all domain needs	Azure DNS
Accelerate delivery of high-bandwidth content to customers worldwide, from applications and stored content to streaming video	Azure Content Delivery Network (CDN)
Protect Azure Applications from DDoS attacks	Azure DDoS protection
Distribute traffic globally while providing high availability and responsiveness	Azure Traffic Manager Azure Front Door
Add private network connectivity to access Microsoft cloud services from corporate networks, as if they were on-premises	Azure ExpressRoute
Monitor and diagnose conditions at a network scenario level	Azure Network Watcher

Firewall capabilities with built-in high availability and zero maintenance	Azure Firewall
Connect to branch offices, retail locations, and sites securely	Azure Virtual WAN
Add a scalable, security-enhanced delivery point for global microservices-based web applications	Azure Front Door

Recommend a connectivity solution that connects Azure resources to the internet

Azure provides various native network services to connect Azure resources to the internet. These include VNets, Azure Bastion, Azure Network NAT Gateway, service endpoints, and Azure Private Link service. These are fully managed PaaS offerings.

VNets

As you’ve learned, VNets are a fundamental connectivity solution in Azure. You can use a VNet to:

- **Communicate between Azure resources** When you deploy VMs and Azure resources such as Azure App Service Environments, Azure Kubernetes Service (AKS), and Azure VM scale sets in an Azure VNet, these resources can communicate using a VNet connection.
- **Communicate between each other** When you connect two or more VNets using VNet peering, resources in either VNet can communicate with each other. If the two VNets you want to connect are in two different Azure regions, you can peer them using global VNet peering.
- **Communicate to the internet** By default, all resources in each VNet can communicate in an outbound direction to the internet. When you assign a public IP or add an available load balancer in front of your VMs, you can manage inbound communication too.
- **Communicate with on-premises networks** You can connect your on-premises network to an Azure VNet with VPN or ExpressRoute connections.

EXAM TIP

VNet peering is the most secure and cost-efficient way to establish low-latency connectivity between Azure VMs deployed in different VNets in the same region or different regions. Traffic between Azure VMs in the peered network remains in the Microsoft backbone infrastructure.

Azure Bastion

Azure Bastion is a native PaaS that provides secure RDP/SSH connectivity to your VMs. With Azure Bastion, you need not expose your VMs over the internet by attaching a public IP and opening ports for RDP/SSH access. Instead, users access Azure Bastion by using a web browser over the internet with the Secure Sockets Layer (SSL) protocol and can then perform a remote login securely. (See Figure 4-10.)

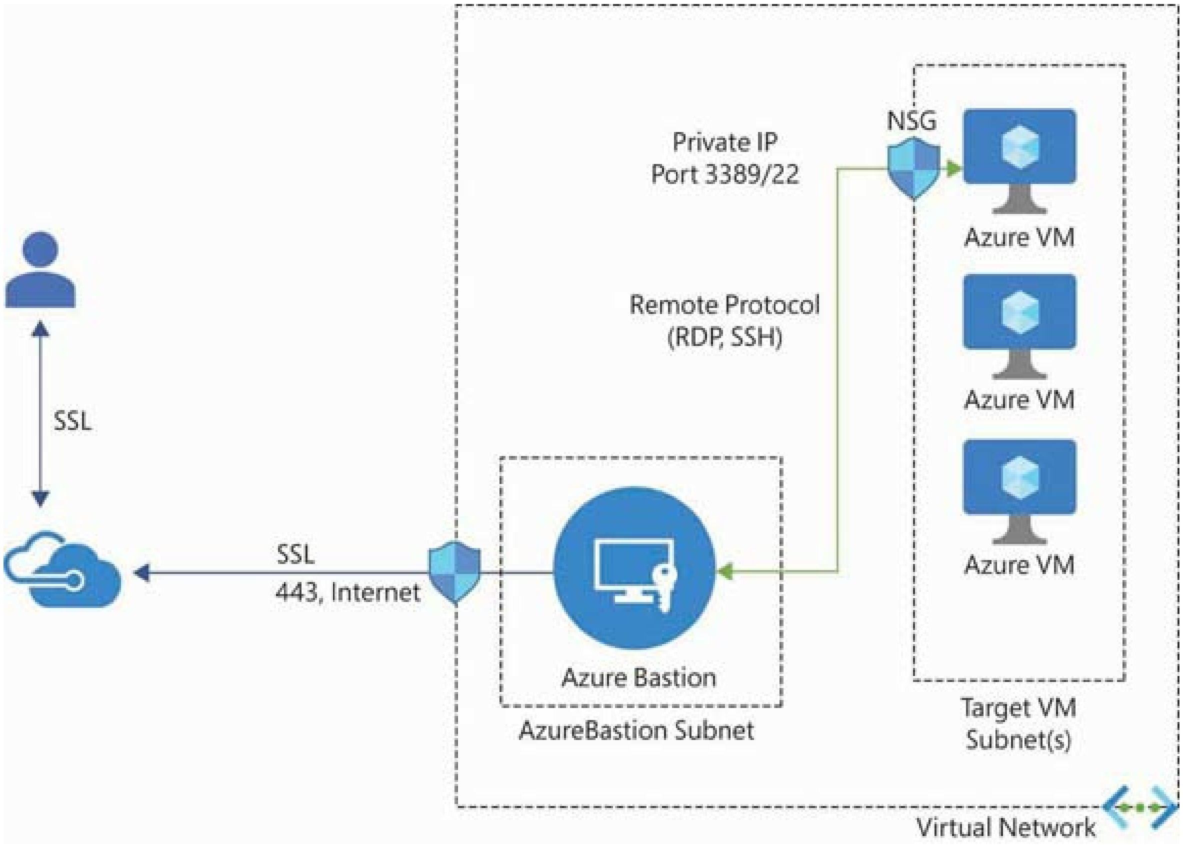


FIGURE 4-10 Azure Bastion

Virtual network NAT gateway

Virtual network NAT (network address translation) is a new service offered by Microsoft to simplify outbound-only internet connectivity for VNets. Virtual network NAT enables outbound connectivity even when you do not have a load balancer or public IP directly attached to your VMs. When you configure NAT on a subnet with this service, your partners see traffic from your specified static public IP address for your outbound connectivity. (See Figure 4-11.)

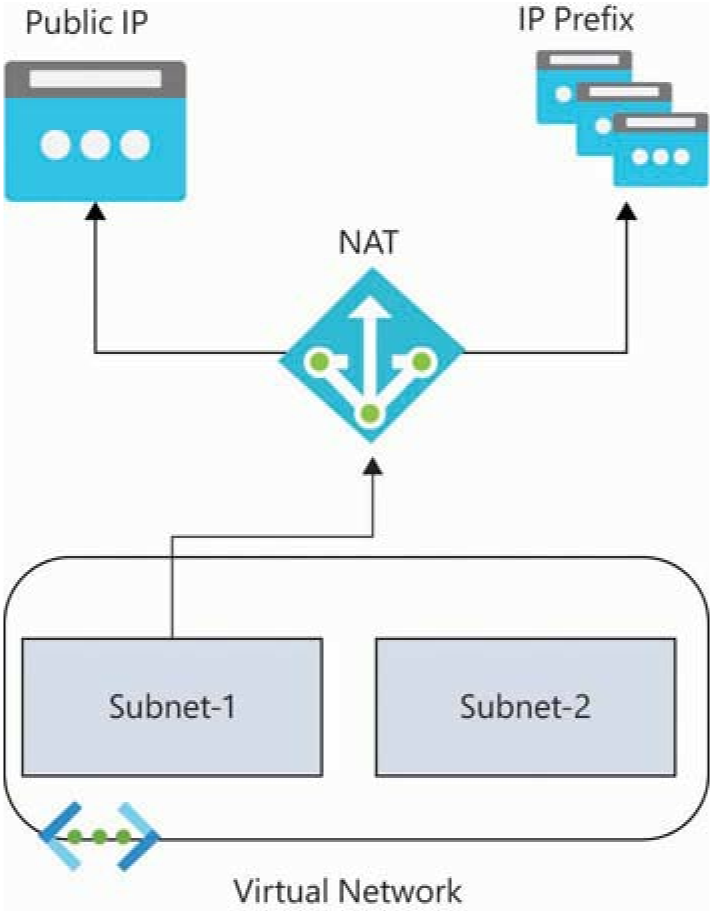


FIGURE 4-11 Virtual network NAT gateway

Service endpoints

Azure provides a unique feature called service endpoints. These allow traffic to be routed from a VNet to specific Azure PaaS services such that it remains on the Microsoft Azure backbone network. VNet service endpoints also extend your VNet's identity to Azure platform services, such as an Azure Storage account, over a direct connection.

When you use service endpoints, service traffic switches to use VNet private addresses as the source IP addresses when accessing the Azure service from a VNet. This switch allows you to access the services without the need for reserved, public IP addresses used in IP firewalls. Service endpoints can secure Azure service resources to your VNet by extending the VNet's identity to the service. After enabling service endpoints in a VNet, you simply add a VNet rule to secure the Azure service resources to your VNet. This improves security by fully removing public internet access to resources and allowing traffic only from your VNet.

Recommend a connectivity solution that connects Azure resources to on-premises networks

Azure provides various solutions to connect Azure resources to on-premises networks. These include ExpressRoute, Azure VPN Gateway, and Azure Virtual WAN.

ExpressRoute

Most enterprise customers have hybrid connectivity needs. The ExpressRoute service enables the extension of on-premises networks into Azure over a private connection facilitated by a connectivity provider. With ExpressRoute, you can expect better reliability and higher throughput—with lower and more consistent latencies—than with typical internet connections. (See Figure 4-12.)

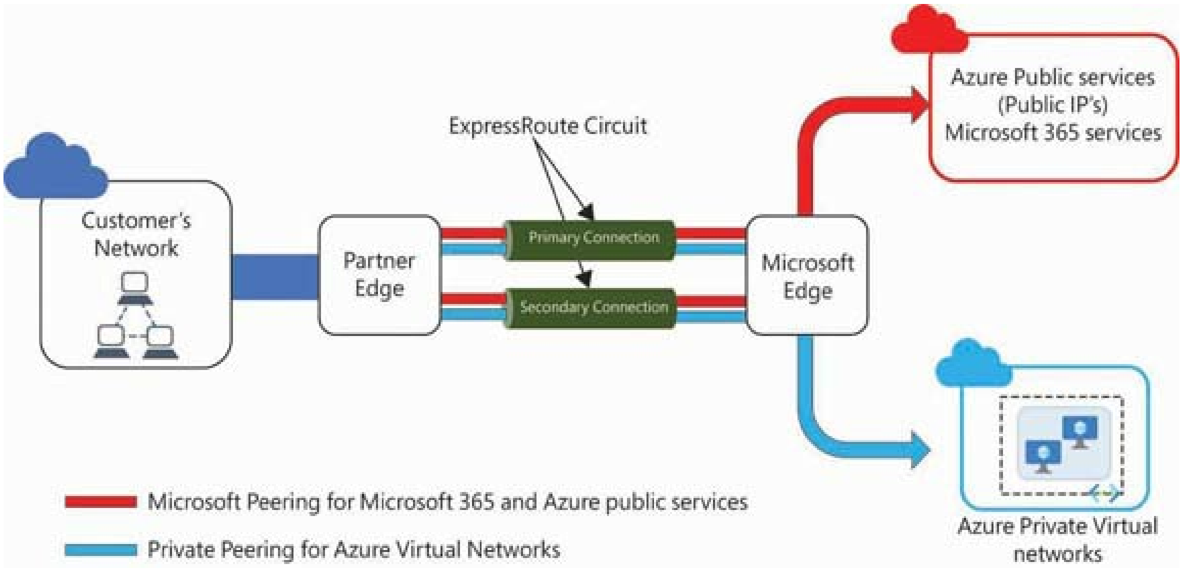


FIGURE 4-12 ExpressRoute

ExpressRoute offers two types of connectivity:

- **Private peering** This allows private connectivity between your Azure VNet and the on-premises network.
- **Microsoft peering** This enables access to Microsoft public endpoints from your on-premises network over a secure connection, not over the public internet.

> ***Note*** Leveraging your existing network provider that is already part of the ExpressRoute partner ecosystem can help reduce the time needed to obtain extensive bandwidth connections to Microsoft.

Microsoft also offers the ExpressRoute Direct service, which allows you to directly connect your on-premises network to the Microsoft backbone. ExpressRoute Direct offers two line-rate options: dual 10 Gbps or 100 Gbps.

Azure VPN Gateway

Azure VPN Gateway is a networking service that helps you create encrypted cross-premises connections from your VNet to on-premises locations and create encrypted connections between various VNets.

There are various Azure VPN Gateway connection options available, such as site-to-site, point-to-site, and VNet-to-VNet. Figure 4-13 shows two site-to-site VPN connections from on-premises sites to the same Azure VNet.

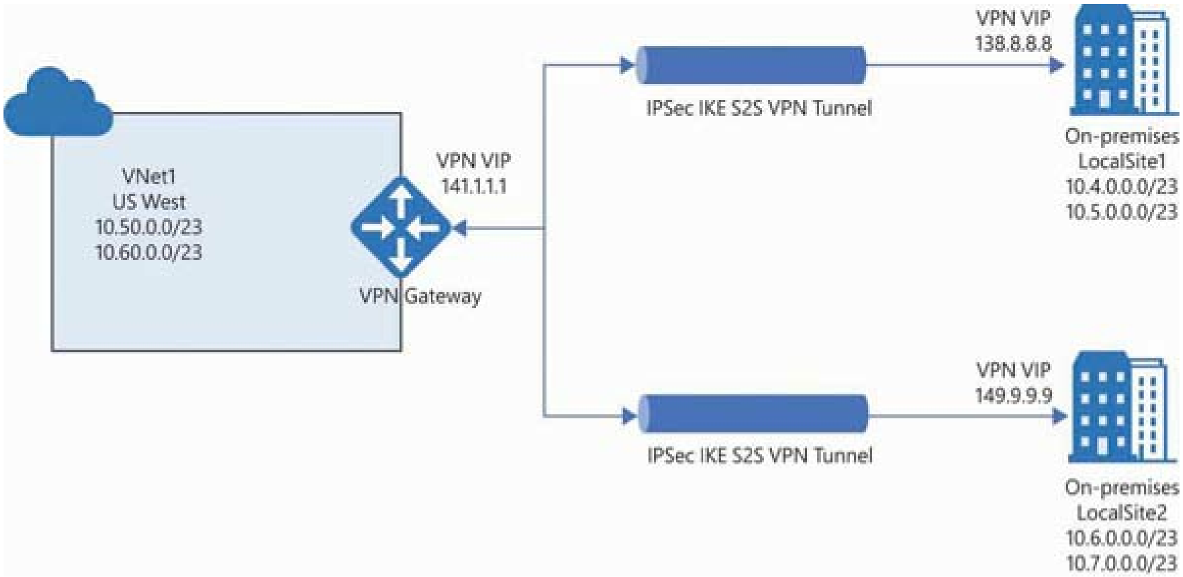


FIGURE 4-13 VPN Gateway

Azure Virtual WAN

As discussed, with Azure Virtual WAN, Azure regions act as hubs to which you can connect your network branches. You can leverage Microsoft's backbone to connect branches and to support branch-to-VM connectivity. Azure Virtual WAN consolidates many Azure Cloud connectivity solutions, such as site-to-site VPN, ExpressRoute, and point-to-site user VPN into one unified solution. You can establish connectivity to Azure VNets by using VNet connections.

Recommend a solution to optimize network performance for applications

Performance is key to the success of any application. Application performance can directly affect your ability to increase customer satisfaction and grow your business.

Many factors affect application performance. One factor is network latency. This is typically directly proportional to the physical distance between the VMs deployed. Azure provides various features to optimize network performance for applications, such as accelerated networking and proximity placement groups.

Accelerated networking

Accelerated networking is an enhancement that enables single root I/O virtualization (SR-IOV) to a VM, improving its networking performance. This high-performance path bypasses the host from the data path, reducing latency, jitter, and CPU utilization for the most demanding network workloads on supported VM types. Without accelerated networking, all networking traffic in and out of the VM must traverse the host and the virtual switch. The virtual switch provides all policy enforcement, such as NSGs, access control lists, isolation, and other network virtualized services, to network traffic.

When accelerated networking is enabled, network traffic first arrives at the VM's network interface (NIC). It is then forwarded to the VM. All network policies applied by the virtual switch are offloaded and applied in hardware. So the NIC can forward network traffic directly to the VM. The NIC bypasses the host and the virtual switch while maintaining all the policies it applied in the host.

The key benefits of accelerated networking are as follows:

- **Lower latency/more packets per second** Eliminating the virtual switch from the data path means the packets spend zero time in the host for policy processing. It also increases the number of packets that can be processed inside the VM.
- **Reduced jitter** Virtual switch policy processing depends on the amount of policy that needs to be applied, and on the workload of the CPU that is doing the processing. Offloading policy enforcement from the virtual switch to the hardware removes this variability by delivering packets directly to the VM. Offloading also removes the host-to-VM communication, all software interrupts, and all context switches.

- **Decreased CPU utilization** Bypassing the virtual switch in the host leads to less CPU utilization for processing network traffic.

The benefits of accelerated networking apply to the VMs it is enabled on. For the best results, you should enable this feature on at least two VMs connected to the same Azure VNet.

Proximity placement groups (PPGs)

Latency plays a particularly important role in application performance. To address this, Azure provides various deployment options:

- **Regions** By placing Microsoft Azure VMs in a single Azure region, you reduce the physical distance between them, which reduces (but does not eliminate) latency.
- **Availability zones** When you place Azure VMs within a single availability zone, they are deployed even closer to each other than when you place them in a single Azure region. Still, a single availability zone might span multiple physical datacenters, so users could still experience some lag.
- **Proximity placement groups (PPGs)** When you assign VMs to a PPG, those VMs are placed in the same physical datacenter. This results in lower and deterministic latency for your applications.

> *More Info* **Azure Network Performance Tuning**
>
> Microsoft has documented TCP/IP performance tuning techniques and various considerations when you use those techniques for Azure VMs. You can find them at *https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-tcpip-performance-tuning*.

Recommend a solution to optimize network security

You can implement network security solutions using appliances on-premises or using native offerings such as NVAs, Azure Firewall, Azure Private Link and private endpoints, Azure Application Gateway, and Azure Web

Application Firewall (WAF). These are fully managed PaaS offerings. You can also use third-party NVAs if your organization prefers them or if native services do not satisfy your organization's specific requirements.

Network virtual appliances (NVAs)

Network virtual appliances (NVAs) play a critical role in Azure, allowing you to use brands and solutions you already know and trust. Most third-party networking offerings are available as NVAs in the Azure Marketplace. These NVAs offer a diverse set of capabilities such as firewalls, WAN optimizers, application delivery controllers, routers, load balancers, proxies, and more. They also enable many hybrid solutions.

A VNet appliance is often a full Linux VM image consisting of a Linux kernel that includes user-level applications and services. Figure 4-14 shows an example of a reference architecture with a demilitarized zone (DMZ) that serves as a perimeter network between on-premises and Azure using NVAs.

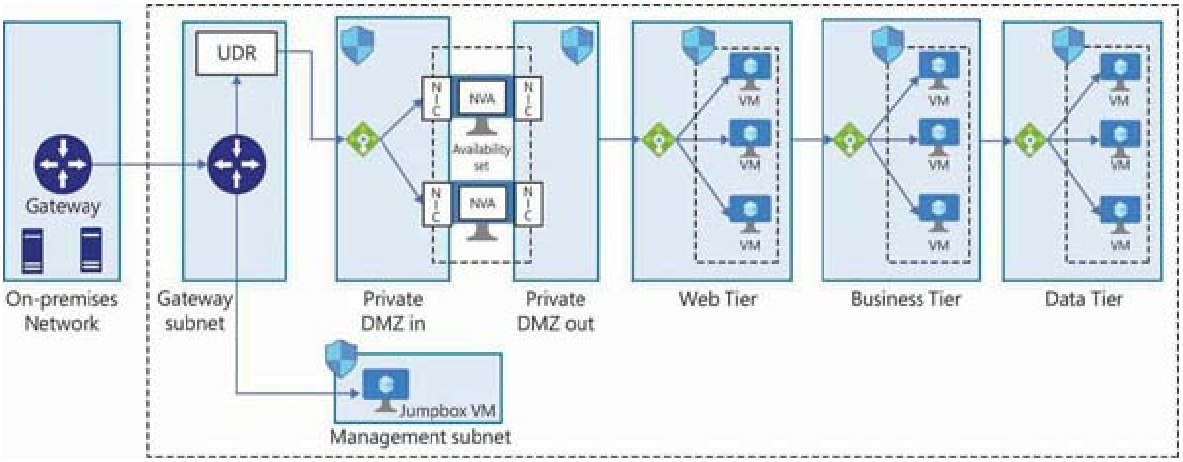


FIGURE 4-14 Reference architecture using NVAs as a demilitarized zone (DMZ)

Azure Firewall

Azure Firewall is a managed, stateful, cloud-native network security service that is highly available by design. This firewall as a service (FaaS) product offers unrestricted and automatic cloud scalability, and you pay as you use it. Microsoft provides a published SLA to support Azure Firewall. Azure

Firewall fits into the DevOps model for deployment and uses cloud-native monitoring tools. Figure 4-15 shows a typical Azure Firewall topology.

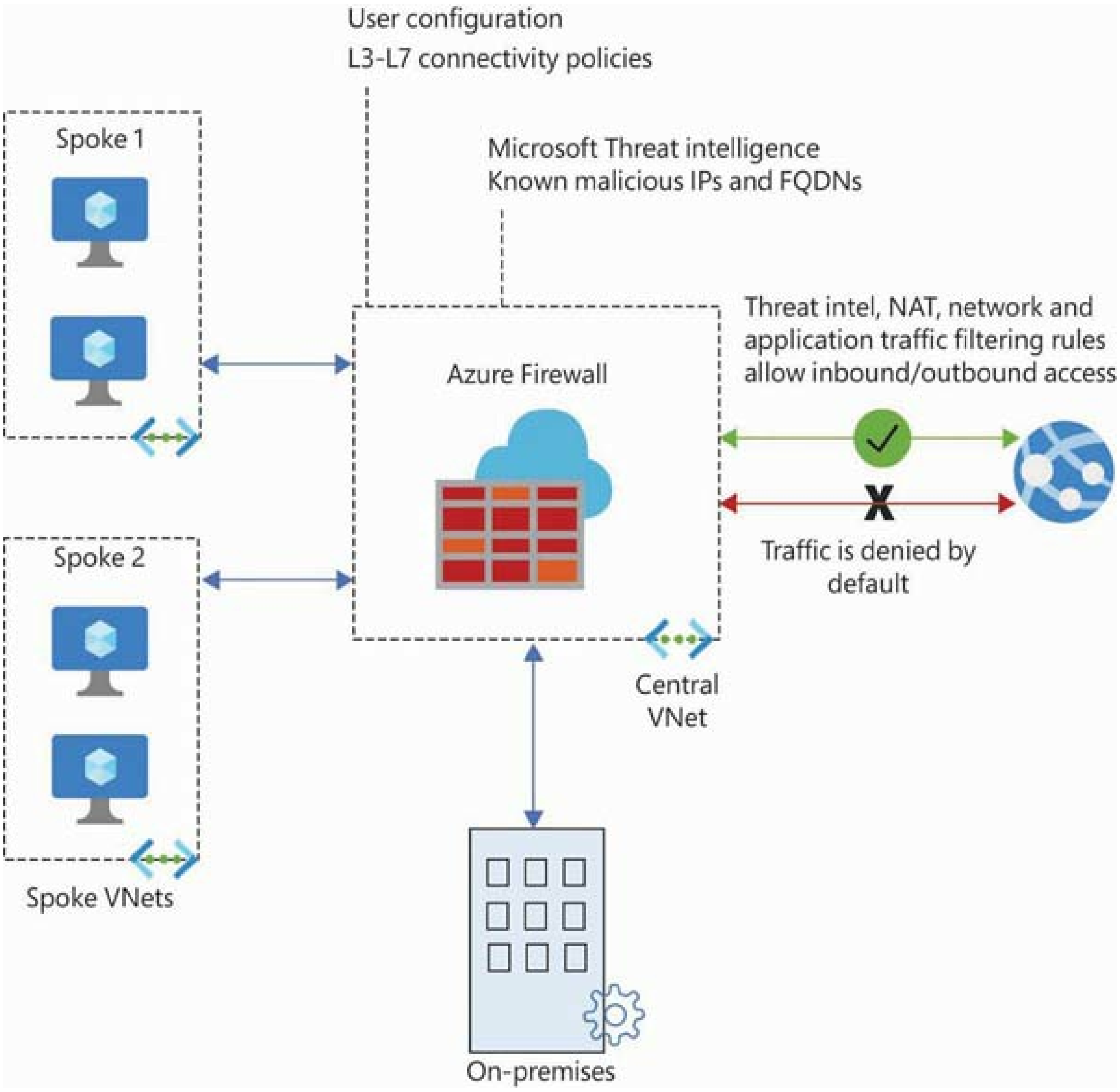

User configuration
L3-L7 connectivity policies
Microsoft Threat intelligence
Known malicious IPs and FQDNs
Spoke 1
Azure Firewall
Threat intel, NAT, network and application traffic filtering rules allow inbound/outbound access
Traffic is denied by default
Spoke 2
Central VNet
Spoke VNets
On-premises

FIGURE 4-15 Azure Firewall

Azure Firewall allows you to centrally create, enforce, and log network connectivity policies across Azure VNets. It uses a static outbound public IP address to identify traffic originating from your VNet. You can use Azure Monitor to generate firewall logs, metrics, and log analytics.

The Azure Firewall feature set has improved over time. As a cloud-native managed service, it provides the following benefits over NVAs:

- It supports easy DevOps integration and can be quickly deployed using IaC, PowerShell, CLI, and REST.
- It offers built-in high availability with cloud-scale.
- It has a zero-maintenance service model.
- It includes unique Azure specialization with features such as service tags and FQDN tags.
- It has a lower total cost of ownership (TCO).

Organizations have diverse security needs. As mentioned, third-party offerings often play a critical role in Azure. You can find most next-generation firewalls as NVAs on Azure Marketplace. NVAs typically provide a richer next-generation feature set that is a must-have for some organizations.

Table 4-11 provides a feature comparison between Azure Firewall and typical NVAs.

TABLE 4-11 Azure Firewall and NVA feature comparison

Feature	Azure Firewall	Typical NVA
FQDN filtering and SSL termination	Yes	Yes
Inbound/outbound traffic filtering and 5-tuple rules (source IP, destination IP, source port, destination port, and protocol)	Yes	Yes
Network address translation (NAT), secure network address translation (SNAT), and destination network address translation (DNAT)	Yes	Yes
Traffic filtering based on a threat intelligence feed to identify high-risk sources/destinations	Yes	Yes
Full logging, including security information event and management (SIEM) integrations	Yes	Yes
Built-in high availability with unrestricted cloud scalability	Yes	Not all vendors provide this.

		Some vendors offer VM-based options.
Azure service and FQDN tags for easy policy management	Yes	No
Integrated monitoring and management; zero maintenance	Yes	No
Easy DevOps integration with Azure REST/PS/CLI/ARM/Terraform	All	ARM and Terraform
SSL termination with deep packet inspection (DPI) to identify known threats	Roadmap	Yes
Traffic filtering rules by target URI (full path, including SSL termination)	Roadmap	Yes
Central management	Azure Firewall Manager Third-party solutions	Vendor-specific options
Application and user-aware traffic filtering rules	Roadmap	Yes
IPSec and SSL VPN gateway	Azure VPN Gateway	Yes
Advanced next-generation firewall; sandboxing features	No	Yes

Azure Private Link

The Azure Private Link service lets you use a private endpoint in your network to access Azure platform services (such as Azure Storage, Azure SQL Database, Cosmos DB, and so on) and Azure-hosted and customer-owned or partner services. With Azure Private Link, traffic between your Azure VNet and these services travels over Microsoft’s backbone network instead of the internet (see Figure 4-16), so you need not consume these services over the public internet. Similarly, you can create your own Private Link service in your VNet and deliver it to your customers to consume.

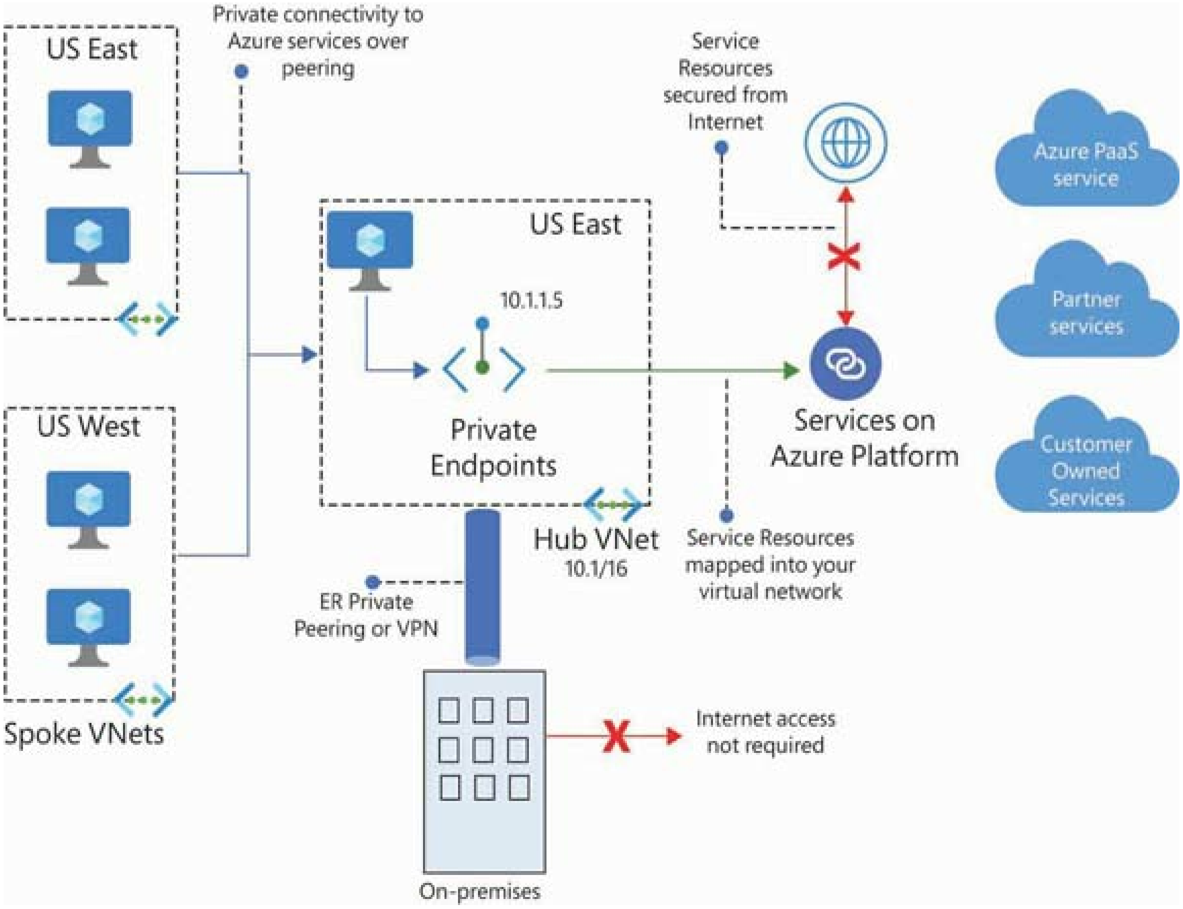

US East
Private connectivity to Azure services over peering
Service Resources secured from Internet
Azure PaaS service
US East
10.1.1.5
Partner services
Private Endpoints
Services on Azure Platform
US West
Customer Owned Services
Hub VNet
10.1/16
Service Resources mapped into your virtual network
ER Private Peering or VPN
Spoke VNets
Internet access not required
On-premises

FIGURE 4-16 Azure Private Link

Application Gateway

Application Gateway is an application layer (OSI Layer 7) load balancer that allows you to manage traffic to your web applications. Azure load balancers operate at the transport layer (OSI Layer 4) and route traffic based on IP address, protocol, and port to a destination IP address and port. Application Gateway uses host-based bindings.

To achieve application layer load balancing, Application Gateway makes routing decisions based on URI path or host headers. For example, suppose you need to route traffic based on the incoming URL. If the /images text appears in the incoming URL, you can route traffic to a specific set of servers (known as a pool) that are configured for images. If the /video text appears in the URL, Application Gateway routes traffic to another pool that is optimized for videos. (See Figure 4-17.)

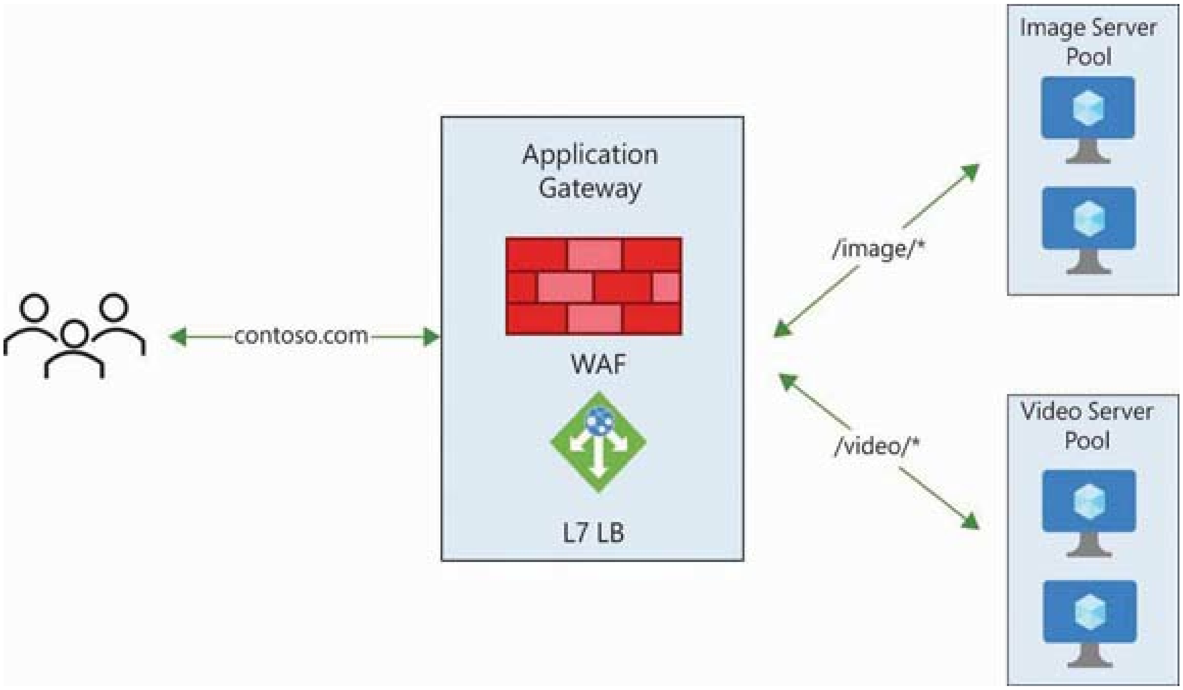


FIGURE 4-17 Application Gateway

EXAM TIP

When it comes to securing traffic from your virtual network to Azure PaaS resources, Azure Private Link is a better option than service endpoints. With Private Link, you assign a private IP address to the PaaS resource from your virtual network. Traffic can reach the PaaS resource from your virtual network or from on-premises without using public endpoints. On the other hand, a service endpoint remains a publicly routable IP address.

Azure Web Application Firewall (WAF)

Azure Web Application Firewall (WAF) protects web applications from common exploits and vulnerabilities. Modern attackers are increasingly targeting web applications with malicious attacks that exploit commonly known vulnerabilities, such as SQL injection attacks and cross-site scripting attacks. Preventing such attacks in application code can be challenging and require rigorous maintenance, patching, and monitoring at many of the application’s layers. A centralized WAF makes security management much more straightforward and ensures protection against such threats. With a WAF solution in place, you can react to a security threat more quickly by remediating a known vulnerability centrally rather than remediating it on multiple individual web applications.

In Azure, you can convert existing Application Gateways to WAF–enabled Application Gateways. Specifically, Azure allows you to enable WAF features with Application Gateway and Azure Front Door. In addition, enabling the WAF feature on the Azure Content Delivery Network (CDN) service is currently under preview.

More Info **Azure Security Baseline for Vnet**

Microsoft has comprehensive security baseline recommendations for protecting assets from internet threats and vulnerabilities. To learn more, see *https://docs.microsoft.com/en-us/azure/virtual-network/security-baseline*.

Recommend a solution for load balancing and traffic routing

In networking, load balancing is significant for any application architecture related to traffic distribution across multiple computing resources. You can use load balancing to make workloads redundant and highly available. Load balancing generally helps with optimizing resource use, maximizing throughput, minimizing response times, and avoiding overloading any single resource.

Azure provides multiple services to manage how you distribute and load balance network traffic. You can use these load-balancing and traffic-routing services individually or together. Depending on your use cases, you can build optimal solutions by combining these services. Following are the primary load-balancing services currently available in Azure:

- **Azure Front Door** Azure Front Door is an application delivery network service that provides global load balancing and site acceleration for web applications. This service lets you manage global routing for your web traffic by optimizing the best performance and instant global failover. It also offers Layer 7 capabilities for your applications, such as SSL offloading, path-based routing, fast failover, and caching, to improve performance and availability.

- **Traffic Manager** This DNS-based traffic load balancer allows you to distribute traffic optimally to services across global Azure regions. Because Traffic Manager is a DNS-based load-balancing service, it works at the domain level.
- **Microsoft Application Gateway** Microsoft Application Gateway offers several Layer 7 load-balancing capabilities. For example, it lets you optimize web farm productivity by offloading SSL termination at the gateway.
- **Azure Load Balancer** This is a high-performance, ultra-low-latency layer for load-balancing inbound and outbound services for all UDP and TCP protocols. Azure Load Balancer is a highly scalable service that can handle millions of requests per second. It supports zone redundancy, ensuring high availability across availability zones.

Of course, every application has unique requirements. You can refer to the decision tree in Figure 4-18 as a starting point.

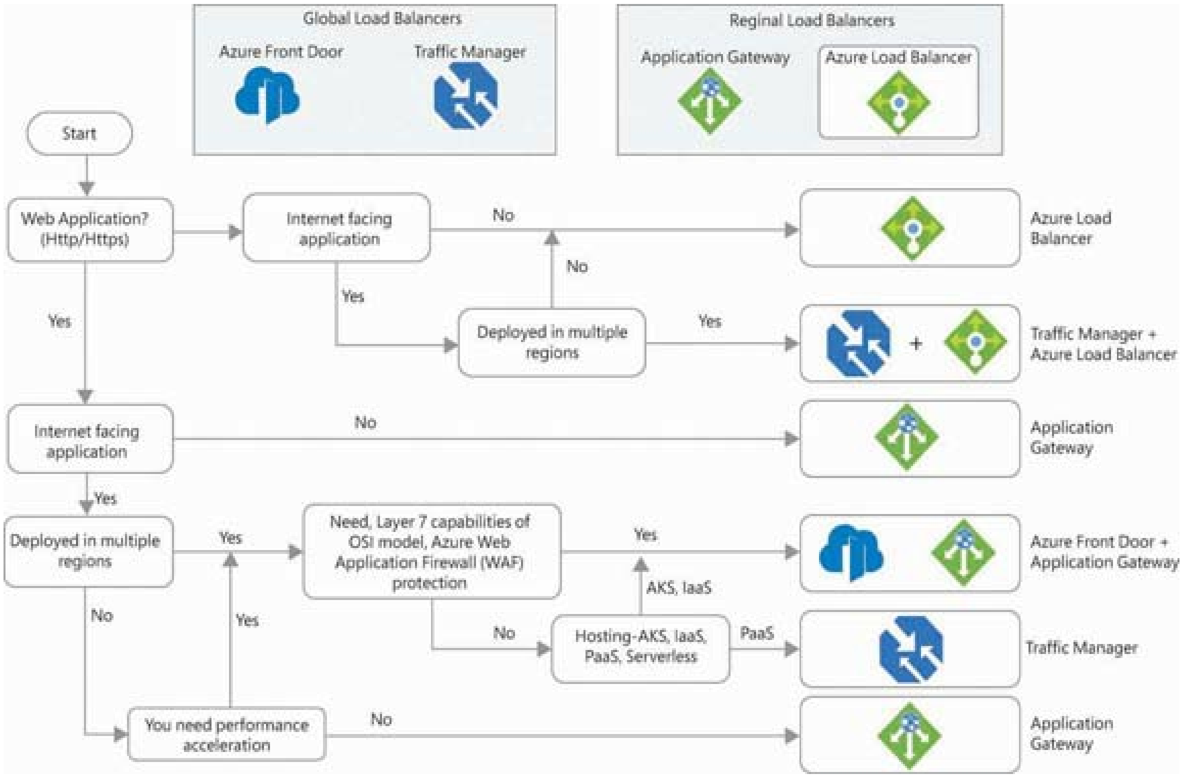

Global Load Balancers
Azure Front Door
Traffic Manager
Reginal Load Balancers
Application Gateway
Azure Load Balancer
Start
Web Application? (Http/Https)
Internet facing application
No
Azure Load Balancer
No
Yes
Yes
Deployed in multiple regions
Yes
+
Traffic Manager + Azure Load Balancer
Internet facing application
No
Application Gateway
Yes
Deployed in multiple regions
Yes
Need, Layer 7 capabilities of OSI model, Azure Web Application Firewall (WAF) protection
Yes
Azure Front Door + Application Gateway
AKS, IaaS
No
Yes
No
Hosting-AKS, IaaS, PaaS, Serverless
PaaS
Traffic Manager
You need performance acceleration
No
Application Gateway

FIGURE 4-18 Decision tree for load-balancing options in Azure

EXAM TIP

The AZ-305 exam typically includes one or more scenario questions to test this skill. The following tips should help you arrive at your recommendations:

- Include Azure Front Door or Traffic Manager in your recommendation if the requirement is a multiregion deployment.
- Determine which load-balancing option is more appropriate when SSL/TLS offloading, WAF, cookie-based session affinity, and URL-path-based routing are the requirements.
- You might need more than one traffic-routing or load-balancing service in your final design, such as Traffic Manager and Azure Load Balancer or Azure Front Door and Microsoft Application Gateway. Refer to the decision tree in Figure 4-18 for various options based on your requirements.

Chapter summary

- An Azure VM is an IaaS that provides virtual processor, memory, storage, and networking resources and the operating system of your choice.
- Azure Container Instances gives you the ability to spin up containers on demand without worrying about existing infrastructure such as Azure VMs.
- Containers provide an immutable infrastructure for your application. They allow you to bundle your application code, libraries, dependencies, and configuration as a container image.
- AKS is a fully managed Kubernetes service that allows you to deploy and manage containerized applications with full-fledged container-orchestration capabilities.
- Azure Function is a function as a service (FaaS) that abstracts underlying infrastructure and operating systems and allows you to execute smaller tasks at scheduled times or when they are triggered by external events.
- Azure Logic Apps is a designer-first integration service that uses a low code/no-code approach to create workflows to automate business processes and orchestrate tasks to integrate line of business (LOB) applications.
- Azure Cache for Redis is a fully managed cache service. It provides an in-memory data store and a critical low-latency and high-throughput data storage solution for modern applications.
- Azure Queue Storage is a queueing service offered by Azure Storage. Using Azure Queue Storage, the producer can push messages to the queue, and the consumer can consume them through some polling mechanism.
- Azure Service Bus is a queueing service that offers enterprise messaging capabilities. These include queueing, a pub/sub model, and advanced integration patterns for cloud-native applications that require advanced queueing capabilities such as FIFO, dead-lettering, and duplicate detection.

- Azure Event Grid is a managed event routing service that relies on a pub/sub model to route information between systems.
- Event Hubs is a big-data pipeline solution for a massive real-time stream of event data from various event producers such as IoT devices and GPS systems.
- DevOps practices enable organizations to achieve continuous delivery and continuous deployment in the software development lifecycle.
- APIM is a front-door gateway service for publishing and managing REST APIs for enterprise-grade applications.
- CAF offers details and proven guidance, best practices, tools, and documentation to help cloud architects and IT stakeholders accelerate cloud adoption and achieve the organization’s business goals.
- Azure Migrate is a native tool for assessing and migrating to Azure.
- The Azure Migrate Server Assessment tool can be used to assess on-premises VMware VMs, Hyper-V VMs, and physical servers for Azure migration.
- DMA is Microsoft’s database assessment and migration tool. It is freely available to download, install, and execute locally.
- DMS is a fully managed service that helps you easily migrate schema, data, and objects from multiple on-premises sources to the Azure platform.
- Storage Migration Service is a graphical tool for migrating storage from a Windows server to Azure.
- Azure Data Box is a family of products designed to transfer a massive amount of data.
- SQL Server Migration Assistant (SSMA) is Microsoft’s database migration tool for heterogeneous migration. It is freely available to download, install, and execute locally.
- Azure VNets are foundational building blocks for most workloads in Azure.
- A hub-and-spoke network topology isolates workloads while sharing services, such as identity, connectivity, and security.

- Azure Virtual WAN is a Microsoft-managed networking solution that provides end-to-end global transit connectivity.
- Azure Bastion is a native PaaS that provides secure RDP/SSH connectivity to VMs.
- Service endpoints allow for the routing of traffic from VNets to specific Azure PaaS services such that traffic always remains on the Microsoft Azure backbone network.
- ExpressRoute enables extensions of on-premises networks into Azure over a private connection facilitated by a connectivity provider.
- Azure VPN Gateway is a networking service that enables you to create encrypted cross-premises connections from your VNet to on-premises locations or to create encrypted connections between various VNets.
- Application Gateway is an application layer (OSI Layer 7) load balancer that allows you to manage traffic to your web applications.
- Azure WAF protects web applications from common exploits and vulnerabilities.
- Azure provides various solutions for network connectivity, such as VNets, ExpressRoute, VPN Gateways, Azure Virtual WAN, VNet NAT Gateway, and Azure Bastion.
- Azure provides various native network security services such as Azure Firewall, Azure WAF, and Azure Front Door.

Thought experiment

Now it is time to validate your skills and knowledge of the concepts you learned in this chapter. You can find answers to this thought experiment in the next section, “Thought experiment answers.”

As a Cloud Solution Architect, you need to recommend a solution for a department in your company that wants to create a web application that serves two types of content: images and dynamically rendered webpages. The website must be secure and geographically redundant, and it should serve its users from the location that is closest to them and offers the lowest latency. Additionally, the default VM pool serving the dynamic content needs to talk to a back-end database hosted on a high-availability cluster.

Thought experiment answers

This section contains the answers to the “Thought experiment” section. The following list contains the critical technical characteristics that must be addressed by this solution:

- **Multi-geo-redundancy** If one region goes down, Traffic Manager routes traffic to the secondary region without manual intervention.
- **Reduced latency** Because Traffic Manager automatically directs the customer to the closest region, the customer experiences lower latency when requesting the web page contents.
- **Independent scalability** You can separate the web application workload by type of content, which allows the application owner to scale request workloads independently of each other. Application Gateway ensures you route traffic to the right application pools based on the specified rules and the application’s health.
- **Internal load balancing** Because the load balancer is in front of the high-availability cluster, the application will connect only to the active and healthy database endpoint. The load balancer delivers connections to the high-availability cluster and ensures that only healthy databases receive connection requests.
- **Security** Transport Layer Security (TLS), previously known as Secure Sockets Layer (SSL), is the standard security technology for establishing an encrypted link between a web server and a browser. This link ensures that all data passed between the web server and browsers remains private and encrypted. Application Gateway supports both TLS termination at the gateway and end-to-end TLS encryption.

Index

Pearson

Exam Ref AZ-305 Designing Microsoft Azure Infrastructure Solutions

List of URLs

Chapter 1: Design identity, governance, and monitoring solutions

- https://www.microsoft.com/en-in/security/business/identity-access/azure-active-directory-pricing
- https://learn.microsoft.com/en-us/azure/role-based-access-control/built-in-roles
- https://learn.microsoft.com/en-us/azure/role-based-access-control/custom-roles
- https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/
- https://learn.microsoft.com/en-us/azure/architecture/framework/
- https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/diagnostic-settings?tabs=portal
- https://learn.microsoft.com/en-us/azure/azure-monitor/agents/data-collection-rule-azure-monitor-agent?tabs=portal
- https://learn.microsoft.com/en-us/azure/azure-monitor/monitor-reference#insights-and-curated-visualizations
- https://learn.microsoft.com/en-us/azure/role-based-access-control/overview#how-azure-rbac-determines-if-a-user-has-access-to-a-resource
- https://learn.microsoft.com/en-us/azure/active-directory/roles/permissions-reference

https://learn.microsoft.com/en-us/azure/active-directory/cloud-sync/what-is-cloud-sync

https://learn.microsoft.com/en-in/azure/cloud-adoption-framework/ready/enterprise-scale/implementation

https://github.com/Azure/Enterprise-Scale/blob/main/docs/ESLZ-Policies.md

https://learn.microsoft.com/en-us/azure/governance/policy/how-to/remediate-resources?tabs=azure-portal

https://learn.microsoft.com/en-us/azure/key-vault/general/assign-access-policy?tabs=azure-portal

https://learn.microsoft.com/en-us/azure/active-directory/develop/v2-oauth2-implicit-grant-flow?WT.mc_id=Portal-Microsoft_AAD_RegisteredApps

Chapter 2: Design data storage solutions

http://<<YourStorageAccountName>>.blob.core.windows.net

http://<<YourStorageAccountName>>.file.core.windows.net

http://<<YourStorageAccountName>>.table.core.windows.net

http://<<YourStorageAccountName>>.queue.core.windows.net

https://azure.microsoft.com/en-us/pricing/hybrid-benefit/

https://docs.microsoft.com/en-us/azure/data-factory/connector-overview

https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-rehydration?tabs=azure-portal

Chapter 3: Design business continuity solutions

https://docs.microsoft.com/en-us/azure/site-recovery/site-recovery-faq

https://docs.microsoft.com/en-us/azure/storage/common/storage-disaster-recovery-guidance

https://docs.microsoft.com/en-us/azure/backup/archive-tier-support

https://docs.microsoft.com/en-us/azure/availability-zones/az-region#azure-regions-with-availability-zones

https://docs.microsoft.com/en-us/azure/storage/common/storage-disaster-recovery-guidance

https://azure.microsoft.com/support/legal/sla/storage/

Chapter 4: Design infrastructure solutions

https://docs.microsoft.com/en-us/azure/api-management/api-management-policies

https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/virtual-machine-recs

https://docs.microsoft.com/en-us/azure/architecture/best-practices/caching

https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services

https://12factor.net/

https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/resources/tools-templates

https://docs.microsoft.com/en-us/azure/cloud-adoption-framework/migrate/azure-best-practices/

https://docs.microsoft.com/en-us/Azure/cloud-adoption-framework/migrate/Azure-best-practices/

https://docs.microsoft.com/en-us/Azure/dms/dms-tools-matrix

https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/azure-subscriptionservice-limits

https://download.microsoft.com/download/1/8/D/18DC8184-E7E2-45EF-823F-F8A36B9FF240/Azure%20File%20Sync%20-%20Namespace%20Mapping.xlsx

https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/hybrid-networking/hub-spoke

https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-tcpip-performance-tuning

https://docs.microsoft.com/en-us/azure/virtual-network/security-baseline

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.

```
{
    "id": "/providers/Microsoft.Authorization/
roleDefinitions/2a2b9908-6ea1-4ae2-8e65-a410df84e7d1",
    "properties": {
        "roleName": "Storage Blob Data Reader",
        "description": "Allows for read access to Azure Storage blob containers and data",
        "assignableScopes": [
            "/"
        ],
        "permissions": [
            {
                "actions": [
                    "Microsoft.Storage/storageAccounts/blobServices/containers/read",
                    "Microsoft.Storage/storageAccounts/blobServices/
generateUserDelegationKey/action"
                ],
        "notActions": [],
        "dataActions": [
          "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/read"
        ],
        "notDataActions": []
      }
    ]
  }
}
```