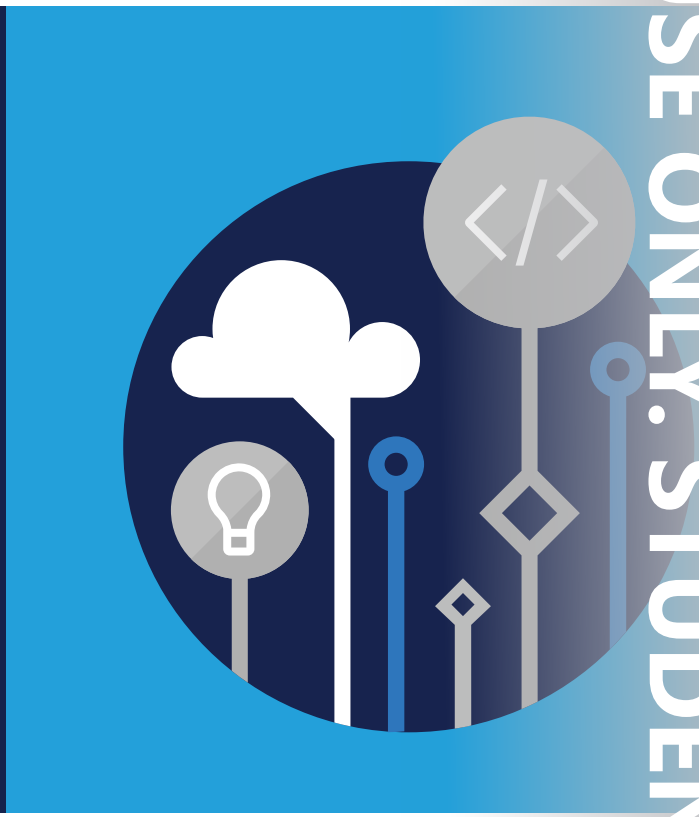
Microsoft

Microsoft Official Course



AZ-500T00

Microsoft_Azure_Security_Technologies

AZ-500T00

Microsoft_Azure_Security_Technologies

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links may be provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

MICROSOFT LICENSE TERMS
MICROSOFT INSTRUCTOR-LED COURSEWARE

These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to your use of the content accompanying this agreement which includes the media on which you received it, if any. These license terms also apply to Trainer Content and any updates and supplements for the Licensed Content unless other terms accompany those items. If so, those terms apply.

BY ACCESSING, DOWNLOADING OR USING THE LICENSED CONTENT, YOU ACCEPT THESE TERMS. IF YOU DO NOT ACCEPT THEM, DO NOT ACCESS, DOWNLOAD OR USE THE LICENSED CONTENT.

If you comply with these license terms, you have the rights below for each license you acquire.

1. **DEFINITIONS.**
   a) "Authorized Learning Center" means a Microsoft IT Academy Program Member, Microsoft Learning Competency Member, or such other entity as Microsoft may designate from time to time.
   b) "Authorized Training Session" means the instructor-led training class using Microsoft Instructor-Led Courseware conducted by a Trainer at or through an Authorized Learning Center.
   c) "Classroom Device" means one (1) dedicated, secure computer that an Authorized Learning Center owns or controls that is located at an Authorized Learning Center's training facilities that meets or exceeds the hardware level specified for the particular Microsoft Instructor-Led Courseware.
   d) "End User" means an individual who is (i) duly enrolled in and attending an Authorized Training Session or Private Training Session, (ii) an employee of a MPN Member, or (iii) a Microsoft full-time employee.
   e) "Licensed Content" means the content accompanying this agreement which may include the Microsoft Instructor-Led Courseware or Trainer Content.
   f) "Microsoft Certified Trainer" or "MCT" means an individual who is (i) engaged to teach a training session to End Users on behalf of an Authorized Learning Center or MPN Member, and (ii) currently certified as a Microsoft Certified Trainer under the Microsoft Certification Program.
   g) "Microsoft Instructor-Led Courseware" means the Microsoft-branded instructor-led training course that educates IT professionals and developers on Microsoft technologies. A Microsoft Instructor-Led Courseware title may be branded as MOC, Microsoft Dynamics or Microsoft Business Group courseware.
   h) "Microsoft IT Academy Program Member" means an active member of the Microsoft IT Academy Program.
   i) "Microsoft Learning Competency Member" means an active member of the Microsoft Partner Network program in good standing that currently holds the Learning Competency status.
   j) "MOC" means the "Official Microsoft Learning Product" instructor-led courseware known as Microsoft Official Course that educates IT professionals and developers on Microsoft technologies.
   k) "MPN Member" means an active Microsoft Partner Network program member in good standing.
   l) "Personal Device" means one (1) personal computer, device, workstation or other digital electronic device that you personally own or control that meets or exceeds the hardware level specified for the particular Microsoft Instructor-Led Courseware.

m) "Private Training Session" means the instructor-led training classes provided by MPN Members for corporate customers to teach a predefined learning objective using Microsoft Instructor-Led Courseware. These classes are not advertised or promoted to the general public and class attendance is restricted to individuals employed by or contracted by the corporate customer.

n) Trainer" means (i) an academically accredited educator engaged by a Microsoft IT Academy Program Member to teach an Authorized Training Session, and/or (ii) a MCT.

o) "Trainer Content" means the trainer version of the Microsoft Instructor-Led Courseware and additional supplemental content designated solely for Trainers' use to teach a training session using the Microsoft Instructor-Led Courseware. Trainer Content may include Microsoft PowerPoint presentations, trainer preparation guide, train the trainer materials, Microsoft One Note packs, classroom setup guide and Pre-release course feedback form. To clarify, Trainer Content does not include any software, virtual hard disks or virtual machines.

2. **USE RIGHTS.** The Licensed Content is licensed not sold. The Licensed Content is licensed on a ***one copy per user*** basis, such that you must acquire a license for each individual that accesses or uses the Licensed Content.

2.1. Below are five separate sets of use rights. Only one set of rights apply to you.

a) **If you are a Microsoft IT Academy Program Member:**

i) Each license acquired on behalf of yourself may only be used to review one (1) copy of the Microsoft Instructor-Led Courseware in the form provided to you. If the Microsoft Instructor-Led Courseware is in digital format, you may install one (1) copy on up to three (3) Personal Devices. You may not install the Microsoft Instructor-Led Courseware on a device you do not own or control.

ii) For each license you acquire on behalf of an End User or Trainer, you may either:

1. distribute one (1) hard copy version of the Microsoft Instructor-Led Courseware to one (1) End User who is enrolled in the Authorized Training Session, and only immediately prior to the commencement of the Authorized Training Session that is the subject matter of the Microsoft Instructor-Led Courseware being provided, **or**

2. provide one (1) End User with the unique redemption code and instructions on how they can access one (1) digital version of the Microsoft Instructor-Led Courseware, **or**

3. provide one (1) Trainer with the unique redemption code and instructions on how they can access one (1) Trainer Content,

provided you comply with the following:

iii) you will only provide access to the Licensed Content to those individuals who have acquired a valid license to the Licensed Content,

iv) you will ensure each End User attending an Authorized Training Session has their own valid licensed copy of the Microsoft Instructor-Led Courseware that is the subject of the Authorized Training Session,

v) you will ensure that each End User provided with the hard-copy version of the Microsoft Instructor-Led Courseware will be presented with a copy of this agreement and each End User will agree that their use of the Microsoft Instructor-Led Courseware will be subject to the terms in this agreement prior to providing them with the Microsoft Instructor-Led Courseware. Each individual will be required to denote their acceptance of this agreement in a manner that is enforceable under local law prior to their accessing the Microsoft Instructor-Led Courseware,

vi) you will ensure that each Trainer teaching an Authorized Training Session has their own valid licensed copy of the Trainer Content that is the subject of the Authorized Training Session,

vii) you will only use qualified Trainers who have in-depth knowledge of and experience with the Microsoft technology that is the subject of the Microsoft Instructor-Led Courseware being taught for all your Authorized Training Sessions,

viii) you will only deliver a maximum of 15 hours of training per week for each Authorized Training Session that uses a MOC title, and

ix) you acknowledge that Trainers that are not MCTs will not have access to all of the trainer resources for the Microsoft Instructor-Led Courseware.

b) **If you are a Microsoft Learning Competency Member:**

i) Each license acquired on behalf of yourself may only be used to review one (1) copy of the Microsoft Instructor-Led Courseware in the form provided to you. If the Microsoft Instructor-Led Courseware is in digital format, you may install one (1) copy on up to three (3) Personal Devices. You may not install the Microsoft Instructor-Led Courseware on a device you do not own or control.

ii) For each license you acquire on behalf of an End User or MCT, you may either:

1. distribute one (1) hard copy version of the Microsoft Instructor-Led Courseware to one (1) End User attending the Authorized Training Session and only immediately prior to the commencement of the Authorized Training Session that is the subject matter of the Microsoft Instructor-Led Courseware provided, **or**

2. provide one (1) End User attending the Authorized Training Session with the unique redemption code and instructions on how they can access one (1) digital version of the Microsoft Instructor-Led Courseware, **or**

3. you will provide one (1) MCT with the unique redemption code and instructions on how they can access one (1) Trainer Content,

provided you comply with the following:

iii) you will only provide access to the Licensed Content to those individuals who have acquired a valid license to the Licensed Content,

iv) you will ensure that each End User attending an Authorized Training Session has their own valid licensed copy of the Microsoft Instructor-Led Courseware that is the subject of the Authorized Training Session,

v) you will ensure that each End User provided with a hard-copy version of the Microsoft Instructor-Led Courseware will be presented with a copy of this agreement and each End User will agree that their use of the Microsoft Instructor-Led Courseware will be subject to the terms in this agreement prior to providing them with the Microsoft Instructor-Led Courseware. Each individual will be required to denote their acceptance of this agreement in a manner that is enforceable under local law prior to their accessing the Microsoft Instructor-Led Courseware,

vi) you will ensure that each MCT teaching an Authorized Training Session has their own valid licensed copy of the Trainer Content that is the subject of the Authorized Training Session,

vii) you will only use qualified MCTs who also hold the applicable Microsoft Certification credential that is the subject of the MOC title being taught for all your Authorized Training Sessions using MOC,

viii) you will only provide access to the Microsoft Instructor-Led Courseware to End Users, and

ix) you will only provide access to the Trainer Content to MCTs.

c) **If you are a MPN Member:**

i) Each license acquired on behalf of yourself may only be used to review one (1) copy of the Microsoft Instructor-Led Courseware in the form provided to you. If the Microsoft Instructor-Led Courseware is in digital format, you may install one (1) copy on up to three (3) Personal Devices. You may not install the Microsoft Instructor-Led Courseware on a device you do not own or control.

ii) For each license you acquire on behalf of an End User or Trainer, you may either:

1. distribute one (1) hard copy version of the Microsoft Instructor-Led Courseware to one (1) End User attending the Private Training Session, and only immediately prior to the commencement of the Private Training Session that is the subject matter of the Microsoft Instructor-Led Courseware being provided, **or**
2. provide one (1) End User who is attending the Private Training Session with the unique redemption code and instructions on how they can access one (1) digital version of the Microsoft Instructor-Led Courseware, **or**
3. you will provide one (1) Trainer who is teaching the Private Training Session with the unique redemption code and instructions on how they can access one (1) Trainer Content,

provided you comply with the following:

iii) you will only provide access to the Licensed Content to those individuals who have acquired a valid license to the Licensed Content,

iv) you will ensure that each End User attending an Private Training Session has their own valid licensed copy of the Microsoft Instructor-Led Courseware that is the subject of the Private Training Session,

v) you will ensure that each End User provided with a hard copy version of the Microsoft Instructor-Led Courseware will be presented with a copy of this agreement and each End User will agree that their use of the Microsoft Instructor-Led Courseware will be subject to the terms in this agreement prior to providing them with the Microsoft Instructor-Led Courseware. Each individual will be required to denote their acceptance of this agreement in a manner that is enforceable under local law prior to their accessing the Microsoft Instructor-Led Courseware,

vi) you will ensure that each Trainer teaching an Private Training Session has their own valid licensed copy of the Trainer Content that is the subject of the Private Training Session,

vii) you will only use qualified Trainers who hold the applicable Microsoft Certification credential that is the subject of the Microsoft Instructor-Led Courseware being taught for all your Private Training Sessions,

viii) you will only use qualified MCTs who hold the applicable Microsoft Certification credential that is the subject of the MOC title being taught for all your Private Training Sessions using MOC,

ix) you will only provide access to the Microsoft Instructor-Led Courseware to End Users, and

x) you will only provide access to the Trainer Content to Trainers.

d) **If you are an End User:**

For each license you acquire, you may use the Microsoft Instructor-Led Courseware solely for your personal training use. If the Microsoft Instructor-Led Courseware is in digital format, you may access the Microsoft Instructor-Led Courseware online using the unique redemption code provided to you by the training provider and install and use one (1) copy of the Microsoft Instructor-Led Courseware on up to three (3) Personal Devices. You may also print one (1) copy of the Microsoft

Instructor-Led Courseware. You may not install the Microsoft Instructor-Led Courseware on a device you do not own or control.

e) **If you are a Trainer.**

i) For each license you acquire, you may install and use one (1) copy of the Trainer Content in the form provided to you on one (1) Personal Device solely to prepare and deliver an Authorized Training Session or Private Training Session, and install one (1) additional copy on another Personal Device as a backup copy, which may be used only to reinstall the Trainer Content. You may not install or use a copy of the Trainer Content on a device you do not own or control. You may also print one (1) copy of the Trainer Content solely to prepare for and deliver an Authorized Training Session or Private Training Session.

ii) You may customize the written portions of the Trainer Content that are logically associated with instruction of a training session in accordance with the most recent version of the MCT agreement. If you elect to exercise the foregoing rights, you agree to comply with the following: (i) customizations may only be used for teaching Authorized Training Sessions and Private Training Sessions, and (ii) all customizations will comply with this agreement. For clarity, any use of "*customize*" refers only to changing the order of slides and content, and/or not using all the slides or content, it does not mean changing or modifying any slide or content.

2.2. **Separation of Components.** The Licensed Content is licensed as a single unit and you may not separate their components and install them on different devices.

2.3. **Redistribution of Licensed Content.** Except as expressly provided in the use rights above, you may not distribute any Licensed Content or any portion thereof (including any permitted modifications) to any third parties without the express written permission of Microsoft.

2.4. **Third Party Notices.** The Licensed Content may include third party code that Microsoft, not the third party, licenses to you under this agreement. Notices, if any, for the third party code are included for your information only.

2.5. **Additional Terms.** Some Licensed Content may contain components with additional terms, conditions, and licenses regarding its use. Any non-conflicting terms in those conditions and licenses also apply to your use of that respective component and supplements the terms described in this agreement.

3. **LICENSED CONTENT BASED ON PRE-RELEASE TECHNOLOGY.** If the Licensed Content's subject matter is based on a pre-release version of Microsoft technology ("**Pre-release**"), then in addition to the other provisions in this agreement, these terms also apply:

a) **Pre-Release Licensed Content.** This Licensed Content subject matter is on the Pre-release version of the Microsoft technology. The technology may not work the way a final version of the technology will and we may change the technology for the final version. We also may not release a final version. Licensed Content based on the final version of the technology may not contain the same information as the Licensed Content based on the Pre-release version. Microsoft is under no obligation to provide you with any further content, including any Licensed Content based on the final version of the technology.

b) **Feedback.** If you agree to give feedback about the Licensed Content to Microsoft, either directly or through its third party designee, you give to Microsoft without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft technology, Microsoft product, or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its technology, technologies, or products to third parties because we include your feedback in them. These rights survive this agreement.

c) **Pre-release Term.** If you are an Microsoft IT Academy Program Member, Microsoft Learning Competency Member, MPN Member or Trainer, you will cease using all copies of the Licensed Content on the Pre-release technology upon (i) the date which Microsoft informs you is the end date for using the Licensed Content on the Pre-release technology, or (ii) sixty (60) days after the commercial release of the technology that is the subject of the Licensed Content, whichever is earliest ("**Pre-release term**"). Upon expiration or termination of the Pre-release term, you will irretrievably delete and destroy all copies of the Licensed Content in your possession or under your control.

4. **SCOPE OF LICENSE.** The Licensed Content is licensed, not sold. This agreement only gives you some rights to use the Licensed Content. Microsoft reserves all other rights. Unless applicable law gives you more rights despite this limitation, you may use the Licensed Content only as expressly permitted in this agreement. In doing so, you must comply with any technical limitations in the Licensed Content that only allows you to use it in certain ways. Except as expressly permitted in this agreement, you may not:
   - access or allow any individual to access the Licensed Content if they have not acquired a valid license for the Licensed Content,
   - alter, remove or obscure any copyright or other protective notices (including watermarks), branding or identifications contained in the Licensed Content,
   - modify or create a derivative work of any Licensed Content,
   - publicly display, or make the Licensed Content available for others to access or use,
   - copy, print, install, sell, publish, transmit, lend, adapt, reuse, link to or post, make available or distribute the Licensed Content to any third party,
   - work around any technical limitations in the Licensed Content, or
   - reverse engineer, decompile, remove or otherwise thwart any protections or disassemble the Licensed Content except and only to the extent that applicable law expressly permits, despite this limitation.
5. **RESERVATION OF RIGHTS AND OWNERSHIP.** Microsoft reserves all rights not expressly granted to you in this agreement. The Licensed Content is protected by copyright and other intellectual property laws and treaties. Microsoft or its suppliers own the title, copyright, and other intellectual property rights in the Licensed Content.
6. **EXPORT RESTRICTIONS.** The Licensed Content is subject to United States export laws and regulations. You must comply with all domestic and international export laws and regulations that apply to the Licensed Content. These laws include restrictions on destinations, end users and end use. For additional information, see www.microsoft.com/exporting.
7. **SUPPORT SERVICES.** Because the Licensed Content is "as is", we may not provide support services for it.
8. **TERMINATION.** Without prejudice to any other rights, Microsoft may terminate this agreement if you fail to comply with the terms and conditions of this agreement. Upon termination of this agreement for any reason, you will immediately stop all use of and delete and destroy all copies of the Licensed Content in your possession or under your control.
9. **LINKS TO THIRD PARTY SITES.** You may link to third party sites through the use of the Licensed Content. The third party sites are not under the control of Microsoft, and Microsoft is not responsible for the contents of any third party sites, any links contained in third party sites, or any changes or updates to third party sites. Microsoft is not responsible for webcasting or any other form of transmission received from any third party sites. Microsoft is providing these links to third party sites to you

only as a convenience, and the inclusion of any link does not imply an endorsement by Microsoft of the third party site.

10. **ENTIRE AGREEMENT.** This agreement, and any additional terms for the Trainer Content, updates and supplements are the entire agreement for the Licensed Content, updates and supplements.
11. **APPLICABLE LAW.**
    a) United States. If you acquired the Licensed Content in the United States, Washington state law governs the interpretation of this agreement and applies to claims for breach of it, regardless of conflict of laws principles. The laws of the state where you live govern all other claims, including claims under state consumer protection laws, unfair competition laws, and in tort.
    b) Outside the United States. If you acquired the Licensed Content in any other country, the laws of that country apply.
12. **LEGAL EFFECT.** This agreement describes certain legal rights. You may have other rights under the laws of your country. You may also have rights with respect to the party from whom you acquired the Licensed Content. This agreement does not change your rights under the laws of your country if the laws of your country do not permit it to do so.
13. **DISCLAIMER OF WARRANTY. THE LICENSED CONTENT IS LICENSED "AS-IS" AND "AS AVAILABLE." YOU BEAR THE RISK OF USING IT. MICROSOFT AND ITS RESPECTIVE AFFILIATES GIVES NO EXPRESS WARRANTIES, GUARANTEES, OR CONDITIONS. YOU MAY HAVE ADDITIONAL CONSUMER RIGHTS UNDER YOUR LOCAL LAWS WHICH THIS AGREEMENT CANNOT CHANGE. TO THE EXTENT PERMITTED UNDER YOUR LOCAL LAWS, MICROSOFT AND ITS RESPECTIVE AFFILIATES EXCLUDES ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.**
14. **LIMITATION ON AND EXCLUSION OF REMEDIES AND DAMAGES. YOU CAN RECOVER FROM MICROSOFT, ITS RESPECTIVE AFFILIATES AND ITS SUPPLIERS ONLY DIRECT DAMAGES UP TO US$5.00. YOU CANNOT RECOVER ANY OTHER DAMAGES, INCLUDING CONSEQUENTIAL, LOST PROFITS, SPECIAL, INDIRECT OR INCIDENTAL DAMAGES.**

    This limitation applies to

    - anything related to the Licensed Content, services, content (including code) on third party Internet sites or third-party programs; and
    - claims for breach of contract, breach of warranty, guarantee or condition, strict liability, negligence, or other tort to the extent permitted by applicable law.

    It also applies even if Microsoft knew or should have known about the possibility of the damages. The above limitation or exclusion may not apply to you because your country may not allow the exclusion or limitation of incidental, consequential or other damages.

Please note: As this Licensed Content is distributed in Quebec, Canada, some of the clauses in this agreement are provided below in French.

Remarque : Ce le contenu sous licence étant distribué au Québec, Canada, certaines des clauses dans ce contrat sont fournies ci-dessous en français.

EXONÉRATION DE GARANTIE. Le contenu sous licence visé par une licence est offert « tel quel ». Toute utilisation de ce contenu sous licence est à votre seule risque et péril. Microsoft n'accorde aucune autre garantie expresse. Vous pouvez bénéficier de droits additionnels en vertu du droit local sur la protection dues consommateurs, que ce contrat ne peut modifier. La ou elles sont permises par le droit locale, les garanties implicites de qualité marchande, d'adéquation à un usage particulier et d'absence de contrefaçon sont exclues.

LIMITATION DES DOMMAGES-INTÉRÊTS ET EXCLUSION DE RESPONSABILITÉ POUR LES DOMMAGES. Vous pouvez obtenir de Microsoft et de ses fournisseurs une indemnisation en cas de dommages directs uniquement à hauteur de 5,00 $ US. Vous ne pouvez prétendre à aucune indemnisation pour les autres dommages, y compris les dommages spéciaux, indirects ou accessoires et pertes de bénéfices.

Cette limitation concerne:

- tout ce qui est relié au le contenu sous licence, aux services ou au contenu (y compris le code) figurant sur des sites Internet tiers ou dans des programmes tiers; et.
- les réclamations au titre de violation de contrat ou de garantie, ou au titre de responsabilité stricte, de négligence ou d'une autre faute dans la limite autorisée par la loi en vigueur.

Elle s'applique également, même si Microsoft connaissait ou devrait connaître l'éventualité d'un tel dommage. Si votre pays n'autorise pas l'exclusion ou la limitation de responsabilité pour les dommages indirects, accessoires ou de quelque nature que ce soit, il se peut que la limitation ou l'exclusion ci-dessus ne s'appliquera pas à votre égard.

EFFET JURIDIQUE. Le présent contrat décrit certains droits juridiques. Vous pourriez avoir d'autres droits prévus par les lois de votre pays. Le présent contrat ne modifie pas les droits que vous confèrent les lois de votre pays si celles-ci ne le permettent pas.

Revised November 2014

Contents

Module 0 Welcome

Start here

About this course

Introduction to AZ-500: Azure Security Engineer

The Azure Security Engineer implements security controls, maintains the security posture, and finds and remediates vulnerabilities by using a variety of security tools. Responsibilities include helping protect data, applications, and networks; managing identity and access; implementing threat protection; and responding to security incident escalations. The Azure Security Engineer often serves as part of a larger team dedicated to cloud-based management and security. The Azure Security Engineer might also help secure hybrid environments as part of an end-to-end infrastructure.

The Azure Security Engineer should have strong skills in scripting and automation; a deep understanding of networking, virtualization, and cloud n-tier architecture; and a strong familiarity with cloud capabilities in general and Microsoft Azure products and services in particular. The Azure Security Engineer should also be familiar with other Microsoft products and services.

The Azure Security Engineer role doesn't focus on helping secure Microsoft 365 and remains separate from the M365 Security and Compliance Administrator role.

Course Syllabus

AZURE-500: Microsoft Azure Security Technologies

Module 1 – Manage Identity and Access

Gone are the days when security focused on a strong perimeter defense to keep malicious hackers out. Anything outside the perimeter was treated as hostile, whereas inside the wall, an organization's systems were trusted. Today's security posture is to assume breach and use the Zero Trust model. Security professionals no longer focus on perimeter defense. Modern organizations have to support access to

data and services evenly from both inside and outside the corporate firewall.
This module will serve as your roadmap as you create and move applications and data to Microsoft Azure. Understanding the security services offered by Azure is key in implementing security-enhanced services. In this module, you will:

- Understand the Zero Trust Model.
- Configure Azure Active Directory for workloads.
- Configure Azure AD Privileged Identity Management.
- Configure Azure tenant security.

Module 2 – Implement Platform Protection

We know that security is job one in the cloud and how important it is that you find accurate and timely information about Azure security. One of the best reasons to use Azure for your applications and services is to take advantage of its wide array of security tools and capabilities. These tools and capabilities help make it possible to create secure solutions on the secure Azure platform. In this module, you will:

- Implement network security.
- Implement host security.
- Configure container Security.
- Implement Azure Resource Manager security.

Module 3 – Manage Security Operations

Azure provides security mechanisms to aid administrators who manage Azure cloud services and virtual machines. These mechanisms include: Authentication and role-based access control. Monitoring, logging, and auditing. Certificates and encrypted communications. A web management portal. In this module, you will learn:

- Configure security services.
- Configure security policies.
- Manage security alerts.

Module 4 – Secure Data and Applications

Azure security for data and applications offers a comprehensive solution that helps organizations take full advantage of the promise of cloud applications while maintaining control with improved visibility into activity. It also increases protection of critical data across cloud applications. With tools to help uncover Shadow IT, assess risk, enforce policies, investigate activities and stop threats, organizations can safely move to the cloud while maintaining control of critical data. In this module, you will:

- Configure security policies to manage data.
- Configure security for data infrastructure.
- Configure encryption for data at rest.
- Implement security for application delivery.
- Configure application security.
- Configure and manage Key Vault.

Setting up a Free Microsoft Azure 30-Day Trial

Setting up a Free Microsoft Azure Trial

Getting started with Azure is now even easier and the benefits have been recently updated.

You can try Azure for free and we'll add a $200 credit for you, which allows you to experiment with any combination of Azure services for 30 days.

When you sign up, you'll also get 12-months of free compute, storage, network, and database services, and over 30 services that are continuously free, to learn and build your next ideas into prototypes.

Get the details, activate your free account, and get to work developing with Azure today - **https://azure.microsoft.com/en-us/free/?OCID=AID624663_OLA_205658197_93454499**.

Course Resources

There is a lot of information on Microsoft Azure. Here are just a few resources that are available.

- Start with the **Azure**[1] home page for links to blogs, product information, sales, pricing, resources, and support.
- Visit the **Azure Documentation**[2] center for the latest information about Azure.
- The **Microsoft Azure Blog**[3] is a great place to read about new product announcements.
- Check out the Microsoft Windows Server **TechNet Library**[4] for Technical Information, Downloads, and Resources.
- Subscribe to **Channel 9**[5] for videos, forums, and events.
- The **Microsoft Press Store**[6] offers a large variety of Azure ebooks and books.

1 https://aka.ms/edx-azure202x-az1
2 https://azure.microsoft.com/en-us/documentation/
3 https://azure.microsoft.com/en-us/blog/
4 https://technet.microsoft.com/en-us/library/bb625087.aspx
5 https://channel9.msdn.com/Search?term=azure#ch9Search
6 https://www.microsoftpressstore.com/search/index.aspx?query=windows+server+2012&x=0&y=0

Module 1 Manage Identity and Access

Configure Azure AD PIM

Zero Trust model

Gone are the days when security focused on a strong perimeter defense to keep malicious hackers out.

Anything outside the perimeter was treated as hostile, whereas inside the wall, an organization's systems were trusted.
Today's security posture is to assume breach and use the Zero Trust model. Security professionals no longer focus on perimeter defense.
Modern organizations have to support access to data and services evenly from both inside and outside the corporate firewall.

This course will serve as your roadmap as you create and move applications and data to Microsoft Azure. Understanding the
security services offered by Azure is key in implementing security-enhanced services.

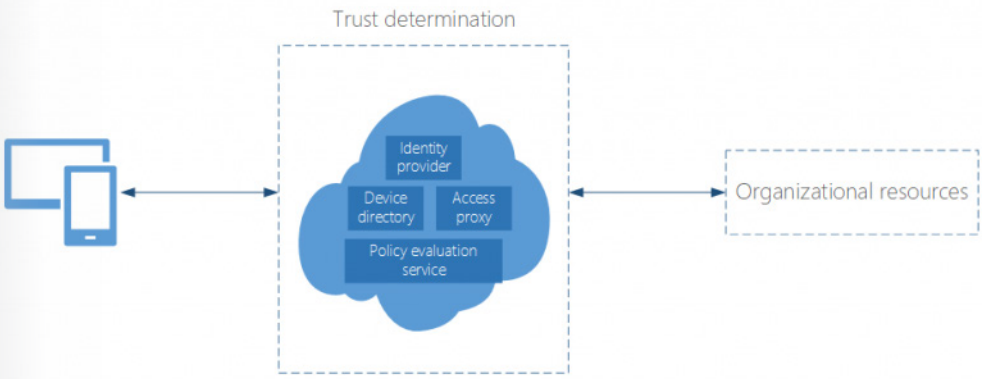
Zero Trust model

The analyst firm Forrester Research introduced the Zero Trust model, which states that you should never assume trust
but instead continually validate trust. When users, devices, and data all resided inside the organization's firewall,
they were assumed to be trusted. This assumed trust allowed for easy lateral movement after a malicious hacker compromised
an endpoint device.

With most users now accessing applications and data from the internet, most of the components of the transactions—that is,
the users, network, and devices—are no longer under organizational control.

The Zero Trust model relies on verifiable user and device trust claims to grant access to organizational resources.
No longer is trust assumed based on the location inside an organization's perimeter.

The following figure depicts the basic components of the Zero Trust model.



Reference: **https://cloudblogs.microsoft.com/microsoftsecure/2018/06/14/building-zero-trust-networks-with-microsoft-365/**

Notice the trust determination components:

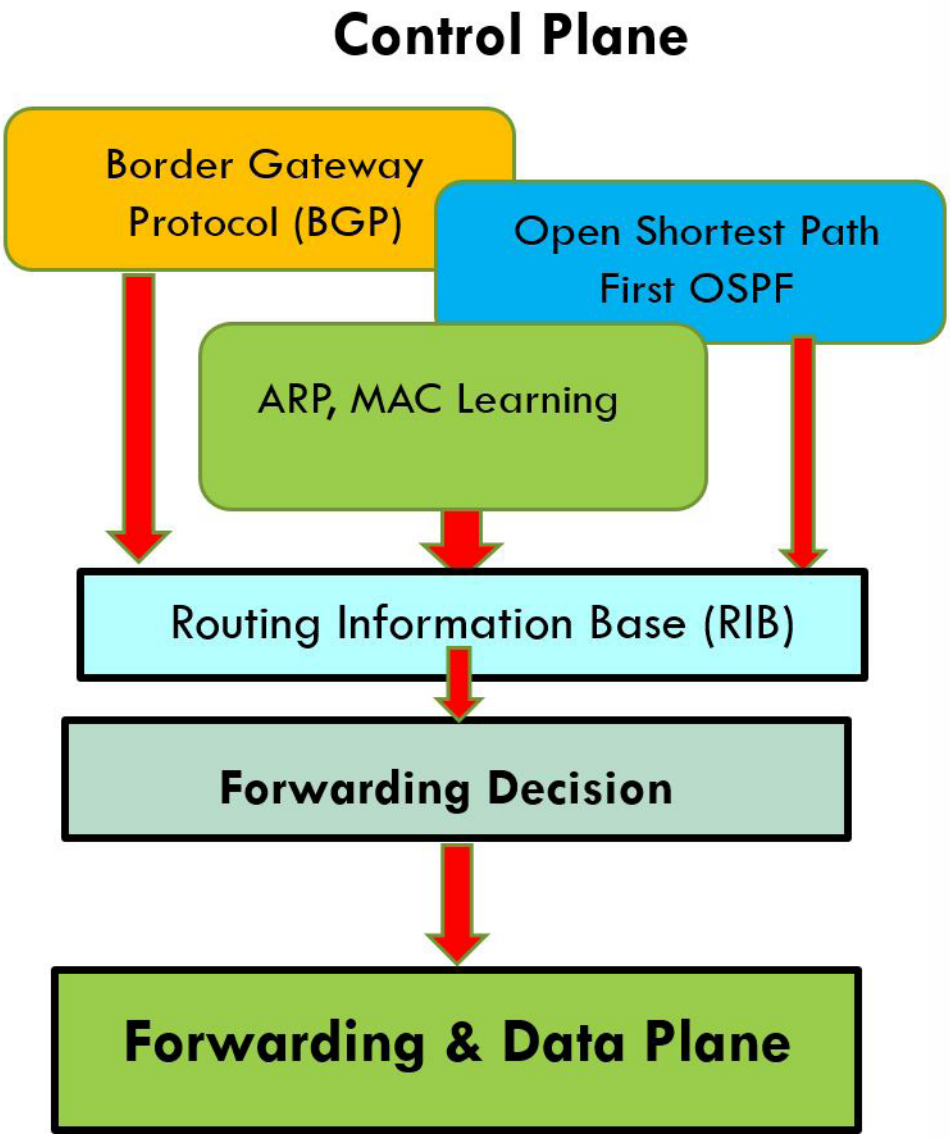
- **Identity provider**. Establishes a user's identity and related information.
- **Device directory**. Validates a device and the device integrity.
- **Policy evaluation service**. Determines whether the user and device conform to security policies.
- **Access proxy**. Determines which organizational resources can be accessed.

The user is the common denominator of these components. That's why a user's identity and how that identity is
managed is now called the control plane. If you can't determine who the user is, you can't establish a trust
relationship for other transactions.

the new control plane

Identity as a Service—the new control plane

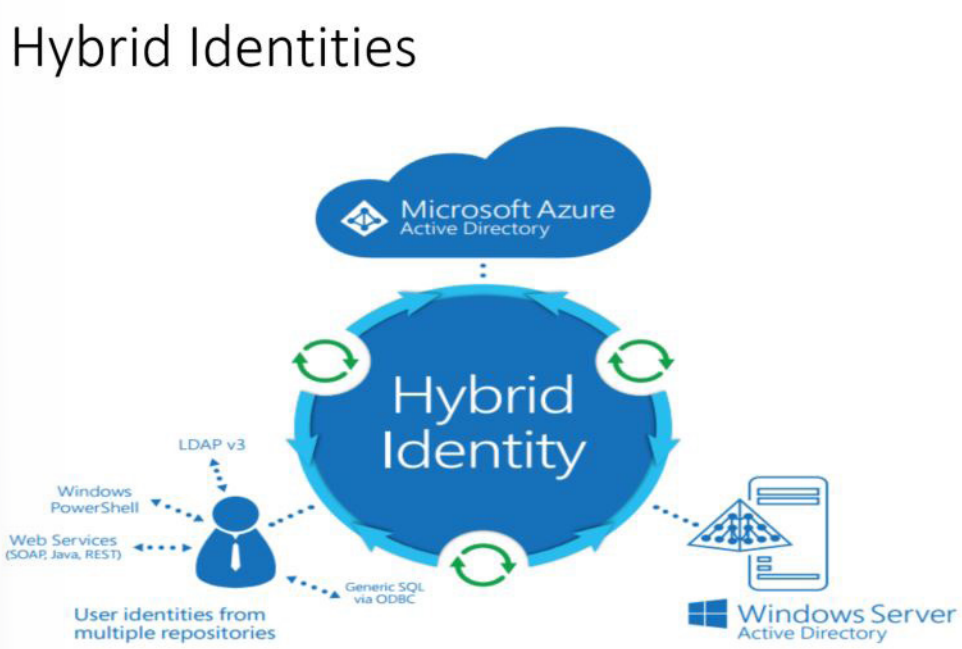
What is the basis for saying that identity management is the new control plane? First, what is the control plane?
In a switch or router, the control plane is the part that controls where the traffic is to go, but it's not responsible for the movement of the traffic. The control plane learns the routes, either static or dynamic. The part responsible for moving the traffic is the forwarding plane. The following figure depicts a simple switch diagram.



A user's identity is like a control plane, because it controls which protocols the user will interact with, which
organizational programs the user can access, and which devices the user can employ to access those programs. Identity
is what helps protect user and corporate data. For example, should that data be encrypted, deleted, or ignored when an issue occurs?

Identity management

On-premises Active Directory, Azure Active Directory (Azure AD), or a hybrid combination of the two all offer services for
user and device authentication, identity and role management, and provisioning.



Identity has become the common factor among many services, like Microsoft Office 365 and Xbox Live, where the person is
the center of the services. Identity is now the security boundary, the new firewall, the control plane—whichever
comparison you prefer. Your digital identity is the combination of who you are and what you're allowed to do. That is:

Credentials + privileges = digital identity

First, you need to help protect your privileged accounts.

These identities have more than the normal user rights and, if compromised, allow a malicious hacker to access sensitive
corporate assets. Helping secure these privileged identities is a critical first step to establishing security assurances
for business assets in a modern organization. Cybercriminals target these accounts and other privileged services in their
kill chain to carry out their objectives.

We recommend Azure AD Privileged Identity Management as the service for this process.

Azure AD PIM

With the Azure AD Privileged Identity Management (PIM) service, you can manage, control, and monitor access to
important resources in your organization. This includes access to resources in Azure AD; Azure; and other Microsoft Online Services, like Office 365 and Microsoft Intune.
However, users still need to carry out privileged operations in Azure AD, Azure, Office 365, and Software as a
Service (SaaS) apps.

Organizations can give users just-in-time privileged access to Azure resources and Azure AD. Oversight is needed for what those users do with their administrator privileges. PIM helps mitigate the risk of exces-

sive,
unnecessary, or misused access rights.

Here are some of the key features of PIM:

- Providing just-in-time privileged access to Azure AD and Azure resources
- Assigning time-bound access to resources by using start and end dates
- Requiring approval to activate privileged roles
- Enforcing Azure Multi-Factor Authentication (MFA) to activate any role
- Using justification to understand why users activate
- Getting notifications when privileged roles are activated
- Conducting access reviews to ensure that users still need roles
- Downloading an audit history for an internal or external audit

Prerequisites

To use PIM, you need one of the following paid or trial licenses:

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

For information about licenses for users, refer to "License requirements to use PIM" at **https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/subscription-requirements**.

Configure PIM

The first person to use PIM in your instance of Azure AD is automatically assigned the **Security Administrator**[1] and **Privileged Role Administrator**[2] roles in the directory. This person must be an eligible Azure AD user. Only privileged role administrators can manage the Azure AD directory role assignments of users. In addition, you can choose to run the **security wizard**[3] that walks you through the initial discovery and assignment experience.

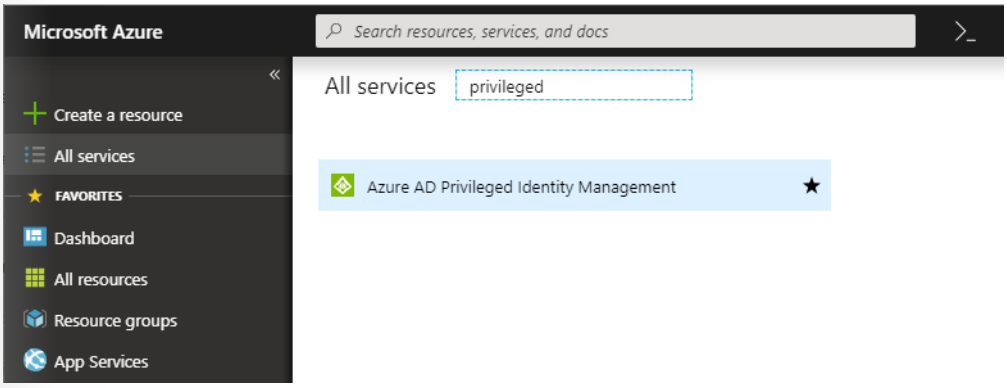
Users or members of a group assigned to the Owner or User Access Administrator roles, and Global Administrators that enable subscription management in Azure AD, are Resource Administrators. These administrators can assign roles, configure role settings, and review access by using PIM for Azure resources. View the list of **built-in roles for Azure resources**[4].

Exercise

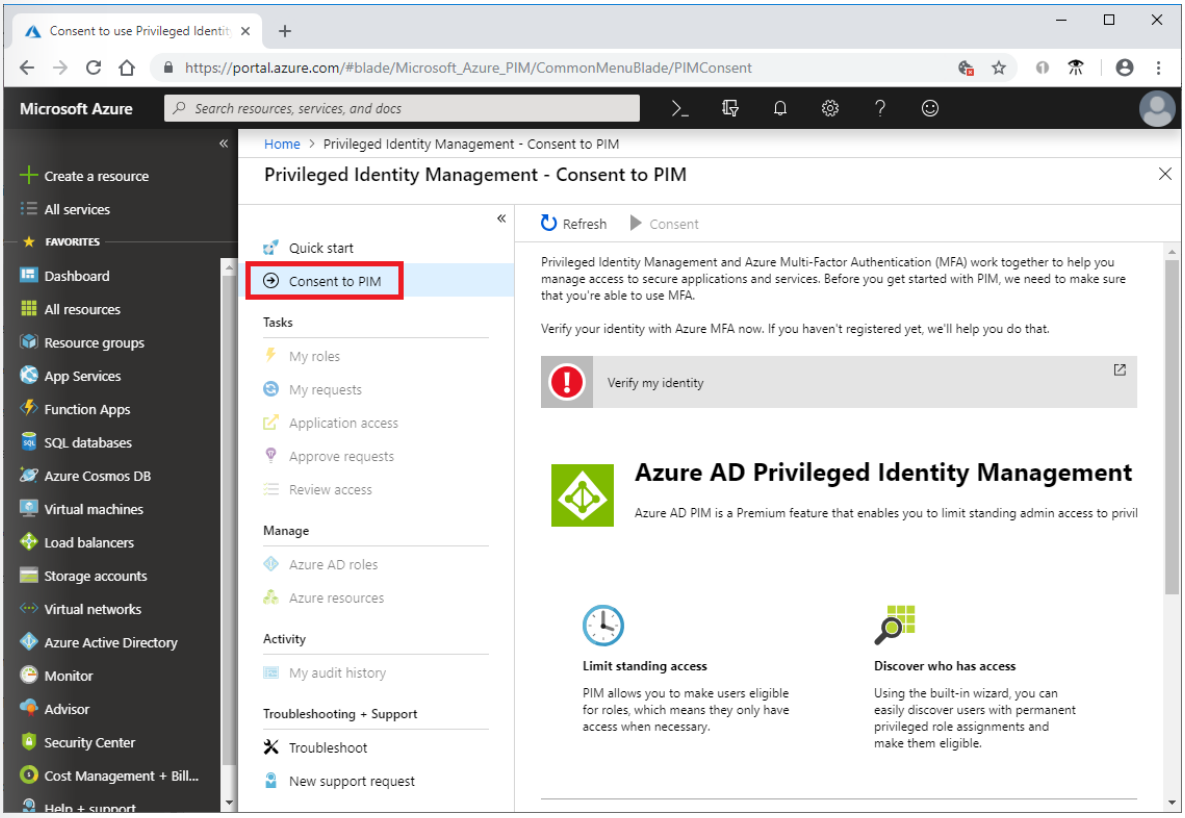
1 https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles
2 https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles
3 https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-security-wizard
4 https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles

To start using PIM in your directory, first enable PIM:

1. Sign in to the **Azure portal**[5] as a Global administrator of your directory.
   Note that you must be a Global administrator with an organizational account (for example, one that contains @yourdomain.com) to enable PIM for a directory.
2. Select **All services**, and then find the **Azure AD Privileged Identity Management** service.

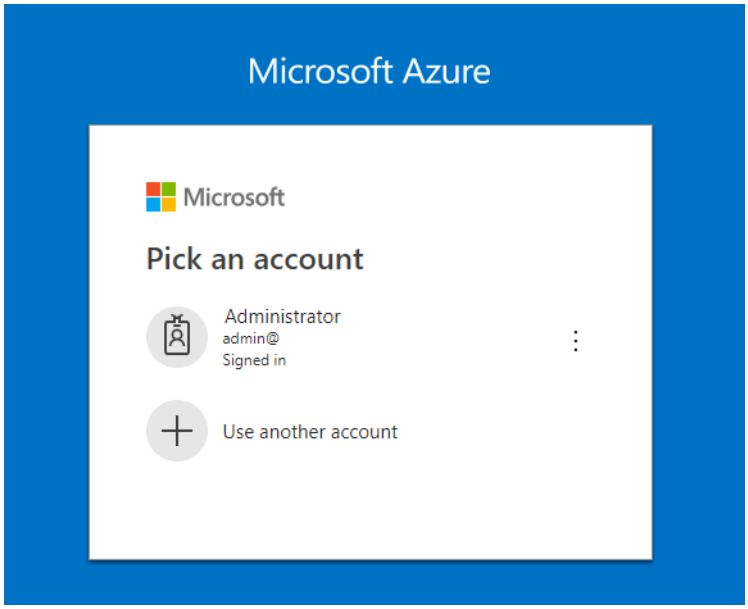


1. Open the PIM Quick start.
2. In the list, select **Consent to PIM**.



1. Select **Verify my identity** to verify your identity with Azure MFA. You're prompted to pick an account.

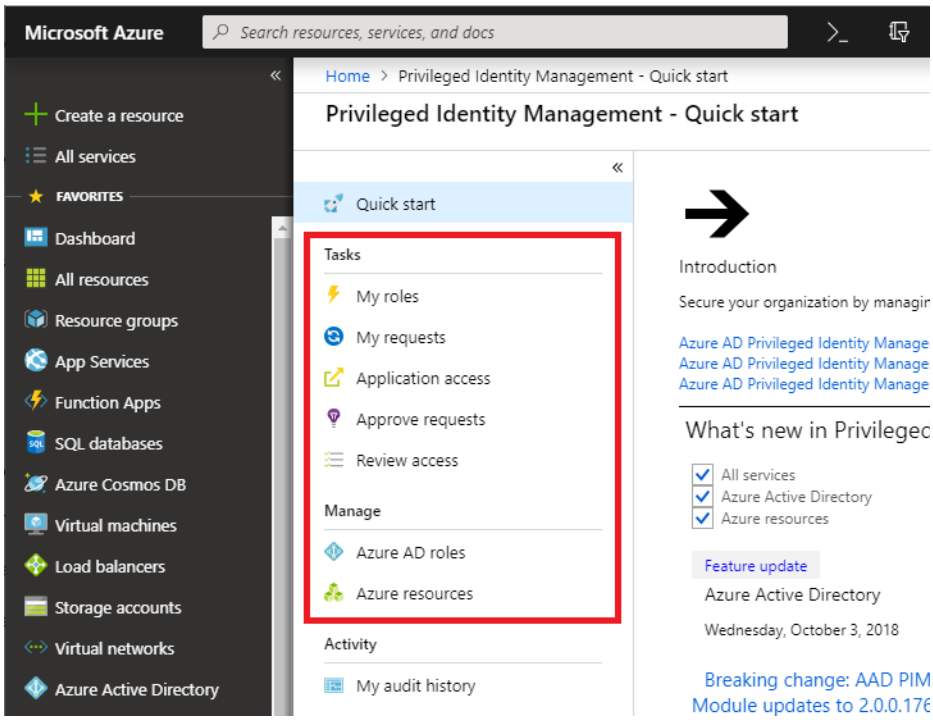
5 https://portal.azure.com/



1. After completing the verification process, select **Consent**.
2. In the message that appears, select **Yes** to consent to the PIM service.

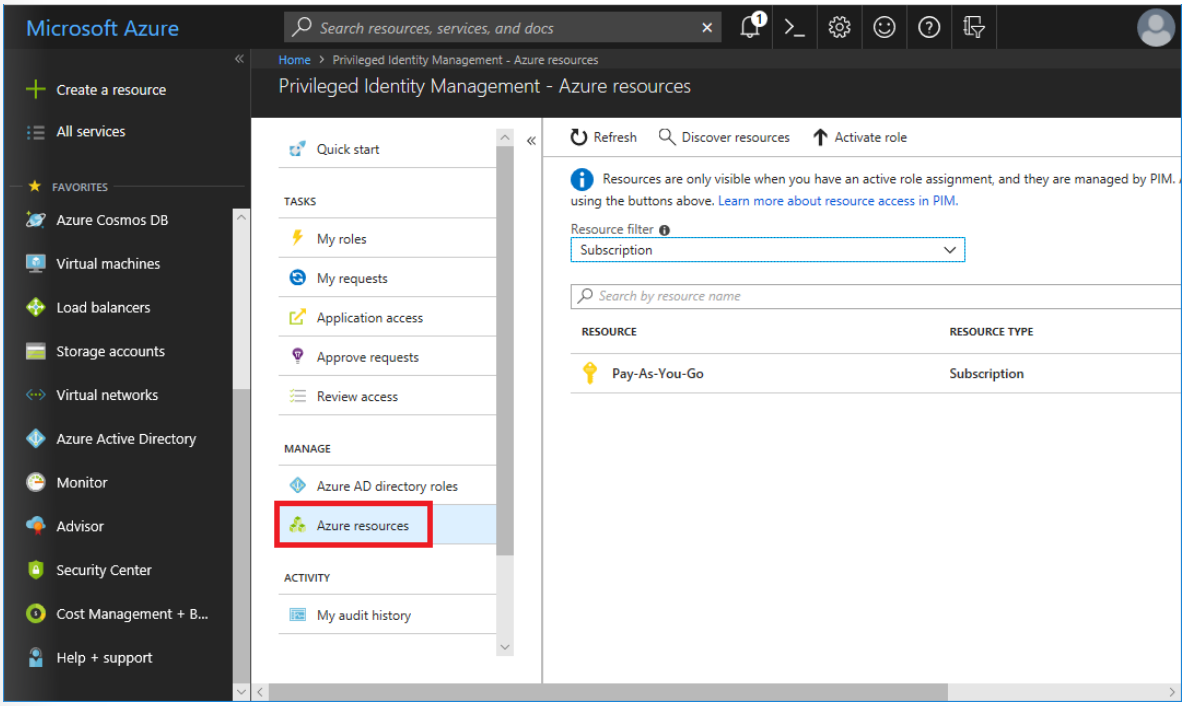
Now that PIM is enabled for your directory, you need to sign up PIM to manage Azure AD roles:

1. Open Azure AD Privileged Identity Management.
2. Select **Azure AD roles**.



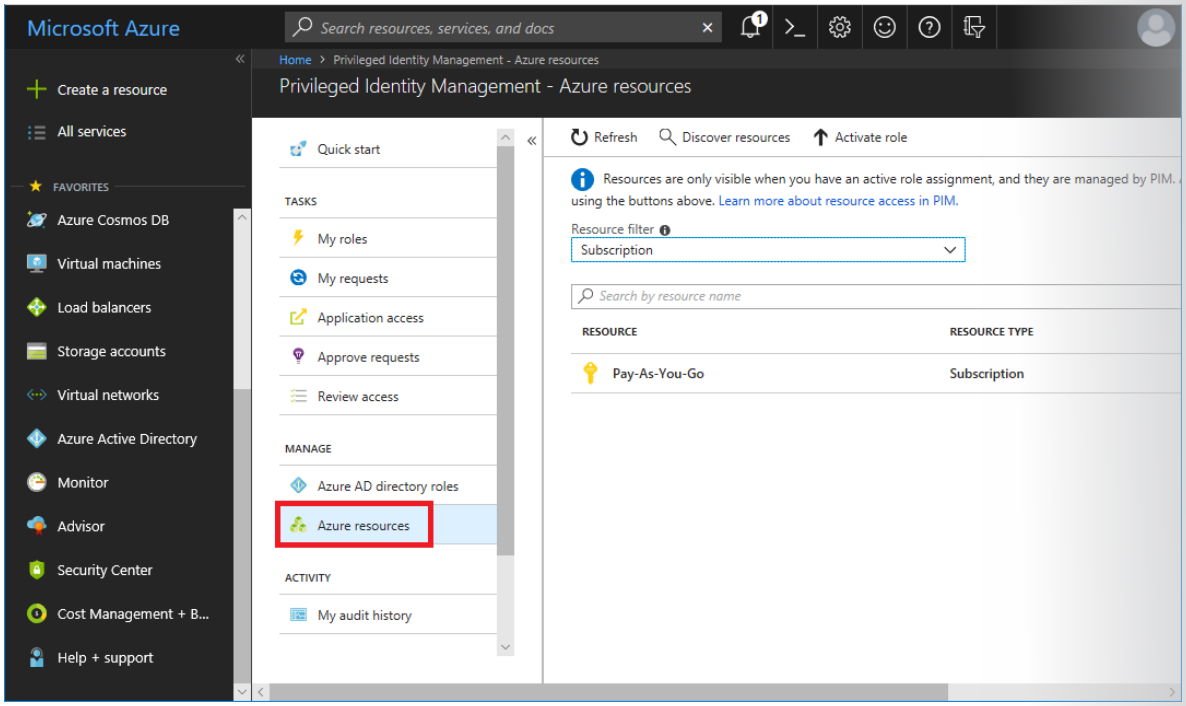
When the signing up completes, the Azure AD options are enabled. You might need to refresh the portal.

After PIM is set up, you can perform your identity management tasks, as the following figure depicts.



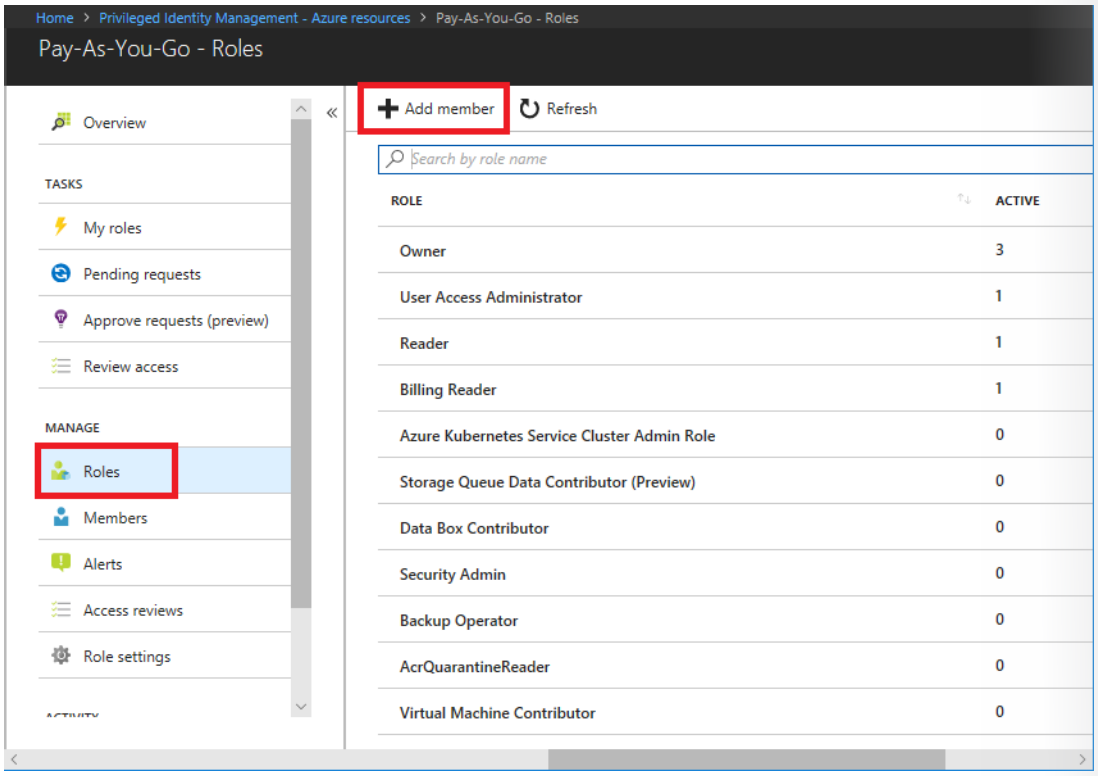
For a detailed description of the various PIM roles and how to active them, refer to "Activate my Azure AD
directory roles in PIM"
at **https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-how-to-activate-role**.

After you define your roles, you can start adding users to those roles. Both Azure AD directory and Azure resource roles exist.
The following figure depicts the resource roles selected.



To manage a resource:

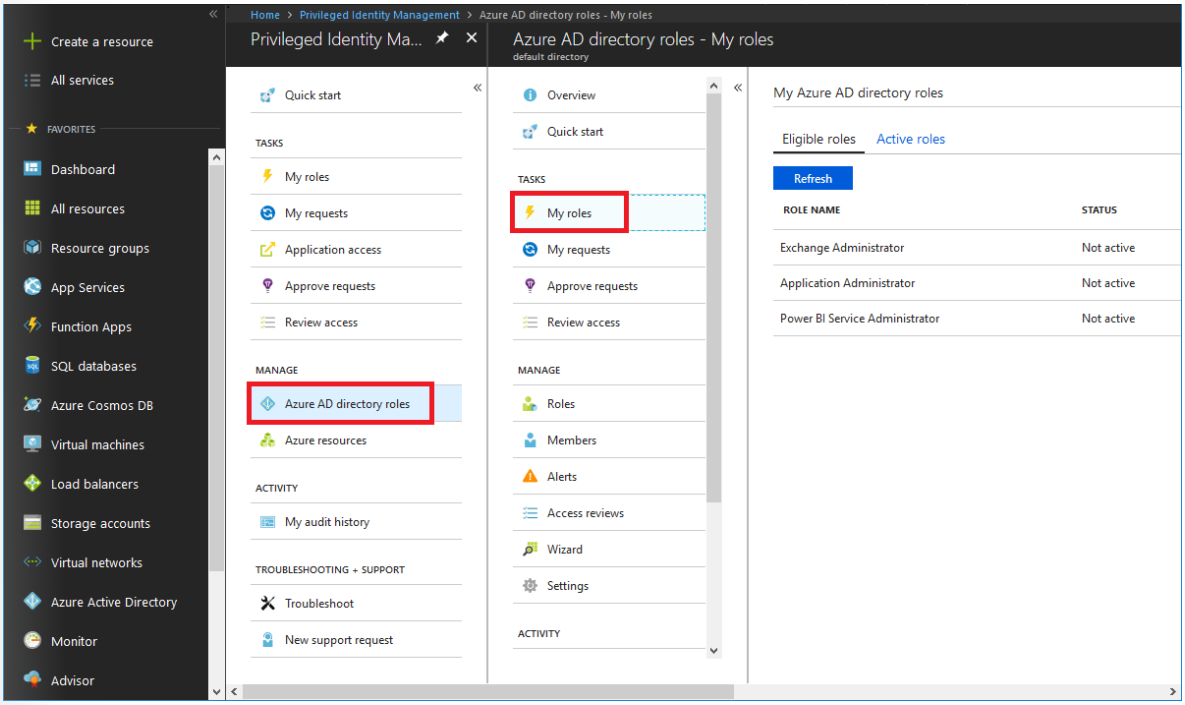
1. Select the resource you want to manage, such as a subscription or management group.
2. Under **Manage**, select **Roles** to see the list of roles for Azure resources. Notice the default roles available for a Pay-As-You-Go subscription.



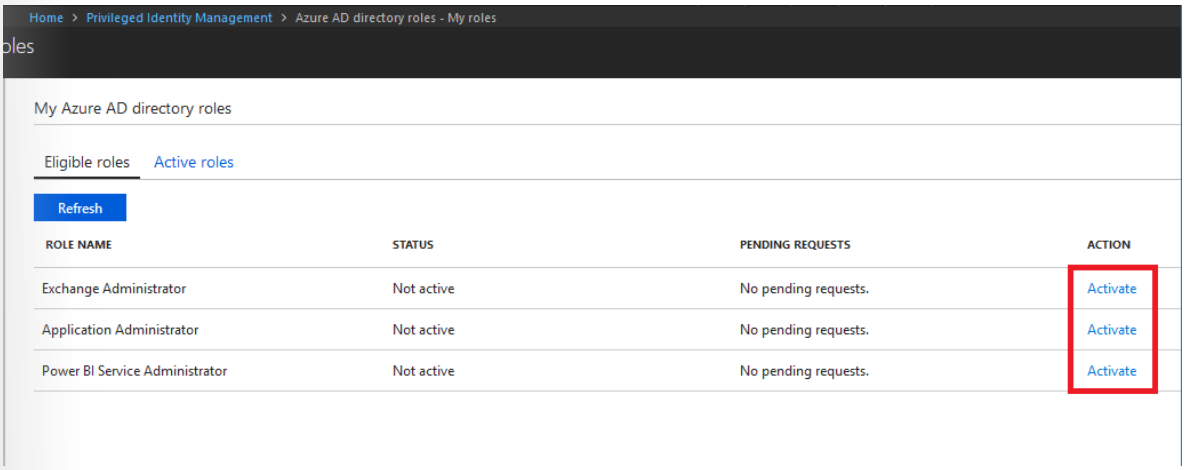
Activate a role

With PIM enabled, you need to activate access to privileged operations when the need to perform privileged actions arises.
If you've been designated as an administrator and you need to take on an Azure AD directory role, you can request activation by selecting **My roles** in PIM. To do so by using the Azure portal:

1. Open the **PIM** blade.
2. Select **Azure AD directory roles**.
3. Select **My roles** to see a list of the roles you are eligible for.

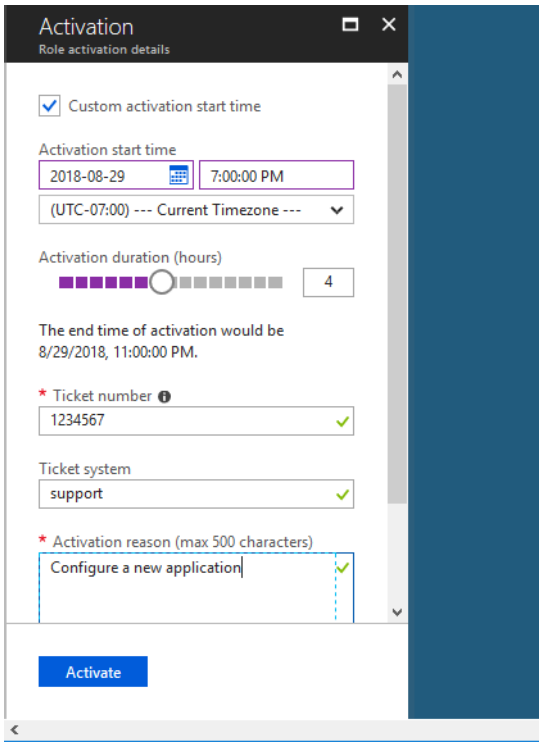


1. Find the role to activate, and then in the **ACTION** column, select **Activate**.



1. Verify your identity if the role requires Azure MFA.
2. Note that the **Activation** pane is now open.

3. If necessary, supply the start time, duration, and reason for activation request. Select **Activate**.



If the role doesn't require approval, it is activated and added to the list of active roles. It takes at least 10 minutes before you can access the corresponding administrative portal or perform functions within a specific
administrative workload. To force an update of your permissions, use the **Application access** page on the **PIM** blade:

1. Select the **Application access** page.
2. Select **Azure Active Directory** to reopen the portal on the All Users page. This invalidates your current
   token and forces the Azure portal to obtain a new token that should have your updated permissions.

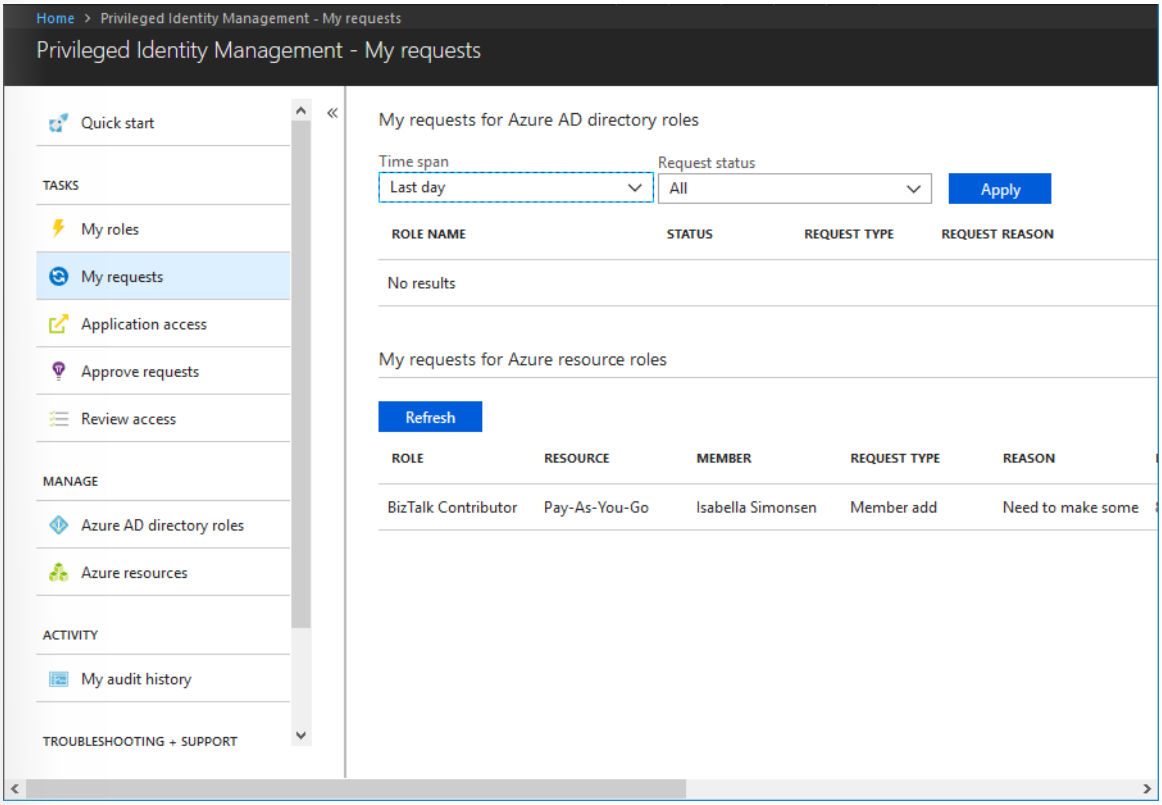
If the **role requires approval**[6] to activate, a notification appears in the upper-right corner of your browser informing you that the request is pending approval.

Monitor the status of your requests

You can view the status of your pending requests to activate:

1. Open Azure AD Privileged Identity Management.
2. Select **My requests** to see a list of your Azure AD directory role and Azure resource role requests.
3. Scroll to the right side to view the **REQUEST STATUS** column.

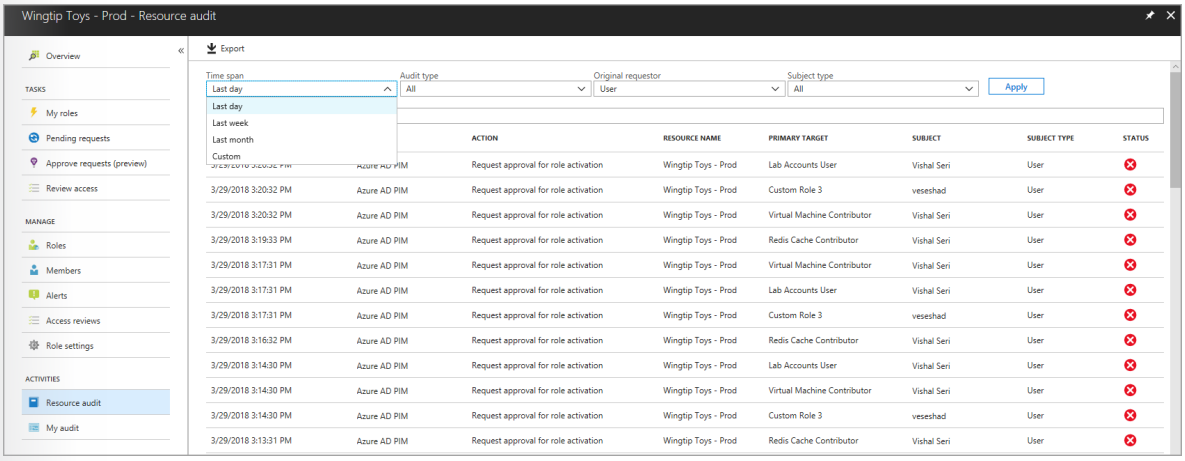
6 https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/azure-ad-pim-approval-workflow



Resource audit history

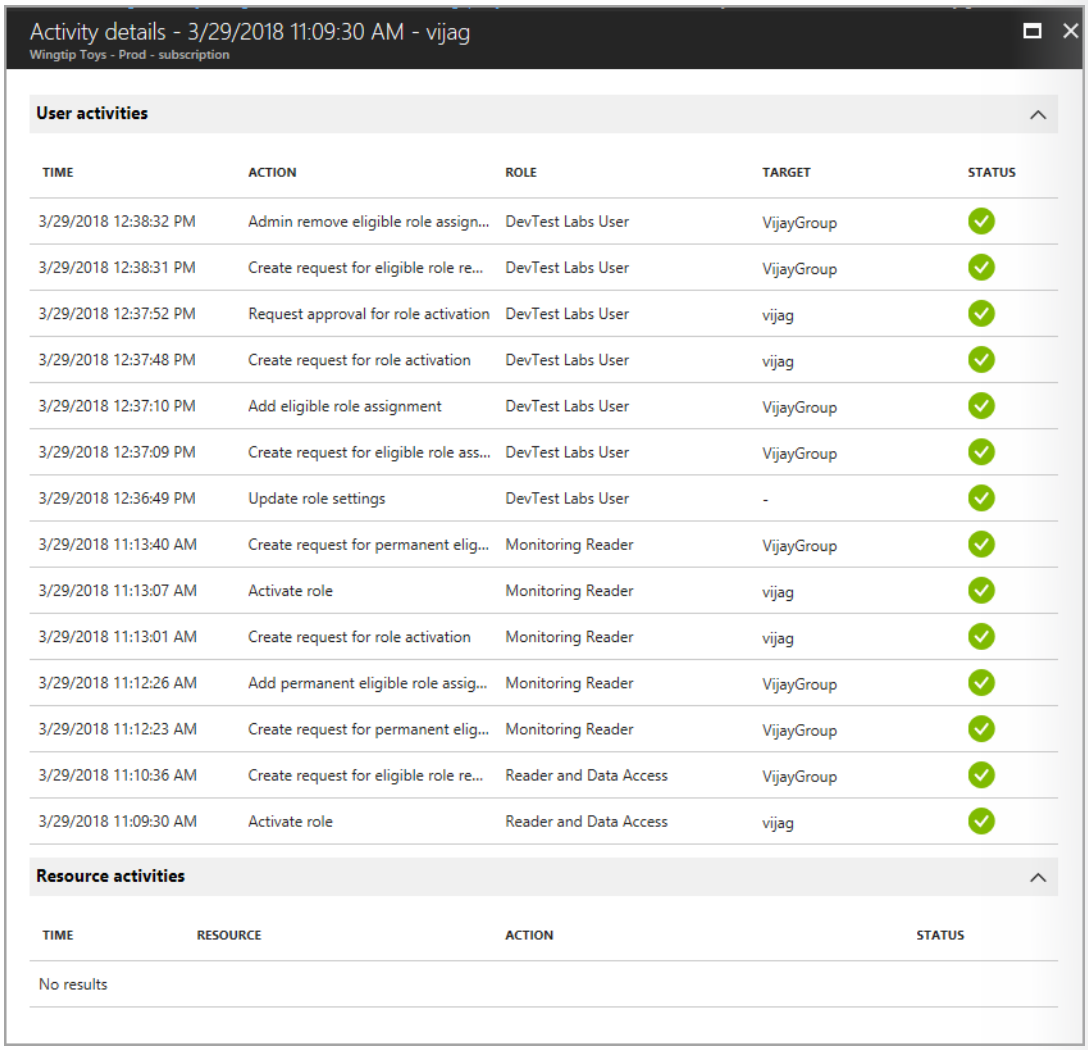
Resource audit gives you a view of all roles activity for a resource:

1. Open Azure AD Privileged Identity Management.
2. Select **Azure resources**.
3. Select the resource for which you want to view the audit history.
4. Select **Resource audit**.
5. Filter the history by using a predefined date or a custom range.



1. For **Audit type**, select **Activate (Assigned + Activated)**.

2. Under **ACTION**, select (**activity**) for a user to see that user's activity details in Azure resources.

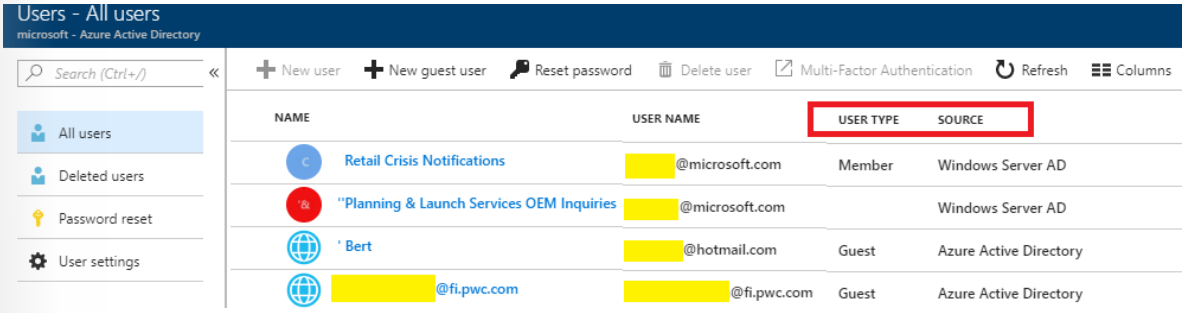


Configure Azure AD for Azure workloads

Understand users and groups

In Azure AD, every user who needs access to resources needs a user account. A user account is a synced Active Directory Domain
Services (AD DS) object or an Azure AD user object that contains all the information needed to authenticate and authorize the
user during the sign-on- process and to build the user's access token.

To view the Azure AD users, access the **All users** blade. Take a minute to access the portal and view your users. Notice the
USER TYPE and **SOURCE** columns, as the following figure depicts.



Typically, Azure AD defines users in three ways:

- Cloud identities. These users exist only in Azure AD. Examples are administrator accounts and users that you manage yourself. Their source is Azure AD.
- Directory-synchronized identities. These users exist in on-premises Active Directory. A synchronization activity that occurs via Azure AD Connect brings these users in to Azure.
- Guest users. These users exist outside Azure. Examples are accounts from other cloud providers and Microsoft accounts.

Take a moment to discuss the types of users you will need.

Add users

You can add cloud identities to Azure AD in multiple ways:

- **Using the Azure portal** You can add new users through the Azure portal. In addition to **Name** and **User name**, you add profile information, like **Job Title** and **Department**.
- **Using Azure PowerShell** You can use the New-AzureADUser Azure PowerShell command to add cloud-based users:

##Create a password object.

```
$PasswordProfile = New-Object -TypeName Microsoft.Open.AzureAD.Model.Pass-
wordProfile
```

Assign the password.

```
$PasswordProfile.Password = "<Password>"
```

Create the new user.

```
New-AzureADUser -AccountEnabled $True -DisplayName "Abby Brown" -Password-
Profile $PasswordProfile -MailNickName "AbbyB" -UserPrincipalName <a href="-
mailto:AbbyB@contoso.com" title="" target="_blank" data-generated=''>AbbyB@
contoso.com</a>
```

- You can also add users to Azure AD through the Office 365 admin center, the Microsoft Intune admin console, and the command-line interface.

For more information, refer to:

- "Add or update a user's profile information using Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-users-profile-azure-portal**.
- "Creating a new user in Azure AD" at **https://docs.microsoft.com/en-us/powershell/azure/active-directory/new-user-sample?view=azureadps-2.0**.
- "az ad user" at **https://docs.microsoft.com/en-us/cli/azure/ad/user?view=azure-cli-latest#az_ad_user_create**[7].

Add user accounts in bulk

You can use Azure PowerShell in several ways to import data into your directory, but the most commonly used way is via a comma-separated values file (CSV). You can either manually create this file (for example, by using Microsoft Excel) or export the file from an existing data source (such as a SQL database or human resources application).

If you'll use a CSV, here are some things to think about:

- Naming conventions. Establish or implement a naming convention for usernames, display names, and aliases. For example, a username might consist of the last name, followed by a period (.), followed by the first name—for example, Smith.John@contoso.com.
- Passwords. Implement a convention for the initial password of a newly created user. Determine how new users will receive their passwords in a security-enhanced way. A commonly used method is generating a random password and then emailing it to the new user or their manager.

To use a CSV:

1. Use **Connect-AzureAD** to create an Azure PowerShell connection to your directory. Connect with an admin account that has privileges on your directory.
2. Create new password profiles for the new users. The passwords for the new users need to conform to the password complexity rules you have set for your directory.
3. Use **Import-CSV** to import the CSV. You need to specify the path and file name of the CSV.
4. Loop through the users in the file, constructing the user parameters needed for each user. Example parameters are User Principal Name, Display Name, Given Name, Department, and Job Title.
5. Use **New-ADUser** to create each user. Be sure to enable each account.

For more information and sample Azure PowerShell scripts, refer to:

- "Importing data into my directory" at **https://docs.microsoft.com/en-us/powershell/azure/active-directory/importing-data?view=azureadps-2.0**.

7 https://docs.microsoft.com/en-us/cli/azure/ad/user?view=azure-cli-latest

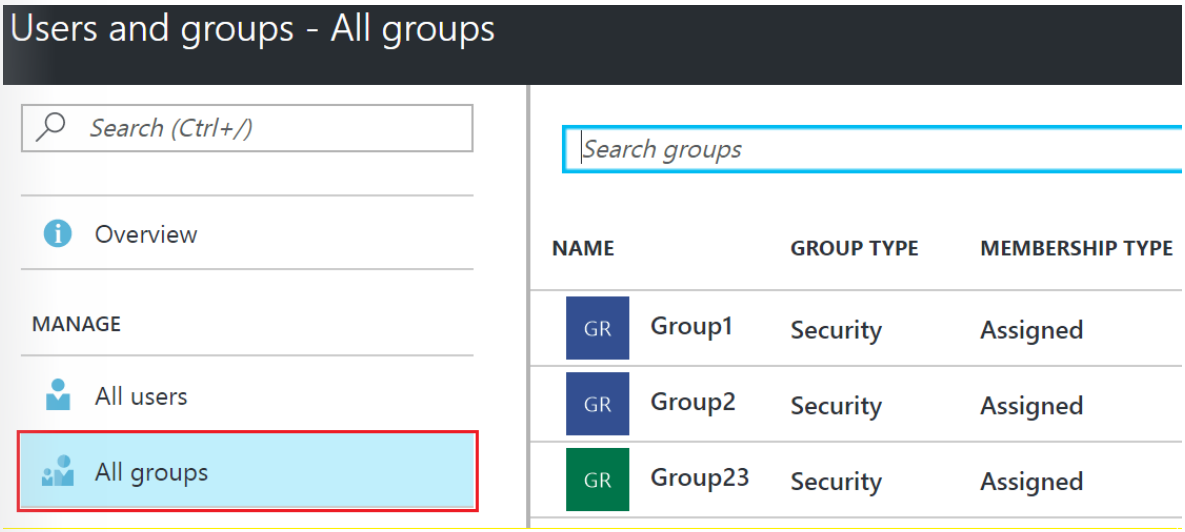
- "New-ADUser" at **https://docs.microsoft.com/en-us/powershell/module/azuread/new-azurea-duser?view=azureadps-2.0**.

Groups

A group helps organize users to make it easier to manage permissions. You can easily add groups through the portal. Two types of groups exist:

- Security groups. You use these security-enabled groups to assign permissions and control access to various resources.
- Distribution groups. Mainly email applications use these groups, which aren't security enabled.

The following figure depicts all the existing groups.



Add groups

You can also use Azure PowerShell to add a group via the **New-AzureADGroup** command:

New-AzureADGroup -Description "Marketing" -DisplayName "Marketing" -MailEnabled $false -SecurityEnabled $true -MailNickName "Marketing"

Add members to groups

Two ways to add members to Azure groups exist:

- Directly assign membership. You create a group and then manually add individual user accounts to the group.
- Dynamically assign membership. You create rules based on characteristics to enable attribute-based dynamic memberships for groups. For example, if a user's department is Sales, that user will be dynamically assigned to the Sales group. You can set up a rule for dynamic membership on security groups or on Office 365 groups. This feature requires an Azure AD Premium P1 license.

Which groups do you need to create? Would you directly assign or dynamically assign membership?

For more information, refer to:

- "Add or remove group members using Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-members-azure-portal**.

- "Dynamic membership rules for groups in Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-dynamic-membership-azure-portal**.
- "Create a basic group and add members using Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-create-azure-portal**.
- "New-AzureADGroup" at **https://docs.microsoft.com/en-us/powershell/module/azuread/new-azureadgroup?view=azureadps-2.0**.

Exercise - Manage groups

Manage group membership

Try to manage group membership for users in your Azure AD tenant. For an explanation of how to do so, refer to
"Add or remove group members using Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-members-azure-portal**.

Create a group and add members

Try to create a group and add members in Azure AD. For an explanation of how to do so, refer to "Create a basic group and add
members using Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-create-azure-portal**. Use a group to perform management tasks, such as assigning licenses or permissions to several users or devices simultaneously.

Manage profile information

Try to add or change profile information for a user in Azure AD. For an explanation of how to do so, refer to "Add or update a user's
profile information using Azure Active Directory" at **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-users-profile-azure-portal**).

✓ As you have time, experiment with other user and group administrative tasks. If you want to try some of these tasks by using Azure PowerShell,
refer to "AzureAD" at **https://docs.microsoft.com/en-us/powershell/module/Azuread/?view=azureadps-2.0**.

Azure Multi-Factor Authentication

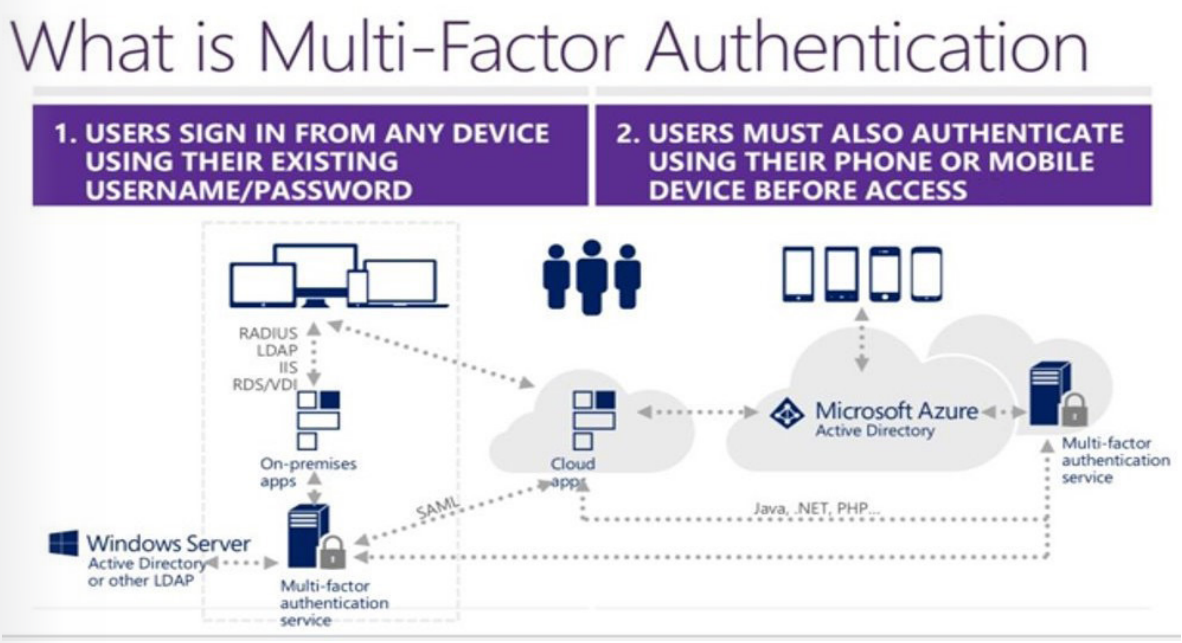
Azure Multi-Factor Authentication (MFA) supplies added security for your identities by requiring two or more elements for full authentication.
These elements fall into three categories:

- Something you know—which might be a password or the answer to a security question.
- Something you possess—which might be a mobile app that receives a notification or a token-generating device.
- Something you are—which typically is a biometric property, such as a fingerprint or face scan used on many mobile devices.

Using Azure MFA increases identity security by limiting the impact of credential exposure. To fully authenticate, a malicious hacker who
has a user's password would also need their phone or their fingerprint, for example. Authentication with only a single factor is insufficient,
and without authentication from Azure MFA, a malicious hacker is unable to use those credentials to

authenticate. You should enable Azure MFA
wherever possible, because it adds enormous benefits to security.

Azure MFA is the Microsoft two-step verification solution. Azure MFA helps safeguard access to data and applications while meeting user
demand for a simple sign-in process. It delivers strong authentication via a range of verification methods, including phone call, text
message, or mobile app verification. The security of Azure MFA lies in its layered approach. Compromising multiple authentication factors
presents a significant challenge for malicious hackers. Even if a malicious hacker manages to learn the user's password, it is useless
without also possessing the trusted device. If the user loses the device, a person who finds it won't be able to use it without the user's password.



Here's what happens when someone tries to connect to a resource that's security enhanced by Azure MFA, and the service is on-premises:

1. The local Azure MFA service validates the initial sign-in request by passing the authentication request to on-premises Active Directory.
2. If the correct credentials were entered and validated, the service sends the request to Azure Multi-Factor Authentication Server.
3. Azure Multi-Factor Authentication Server sends an additional verification challenge to the user. The methods you can easily configure are:
   - Phone call. Azure Multi-Factor Authentication Server places a call to the user's registered phone.
   - Text message. Azure Multi-Factor Authentication Server sends a six-digit code to the user's mobile phone.
   - Mobile app notification. Azure Multi-Factor Authentication Server sends a verification request to a user's smartphone, which asks them to complete the verification by selecting Verify in the mobile app.

- Mobile app verification code. Azure Multi-Factor Authentication Server sends a six-digit code to the user's mobile app. The user then enters this code on the sign-in page.
- Initiative for Open Authentication (OATH)–compliant tokens. You can use these as a verification method.

If the service is running in Azure:

1. The service sends the sign-in request first to Azure AD for the initial validation and then to Azure Multi-Factor Authentication Server.
2. Azure Multi-Factor Authentication Server sends an additional verification challenge to the user, as just described.

Azure MFA allows the provider of the request service to validate that users are real people and not bots, that they have their devices with them,
and that they can provide any additional information.

Azure MFA improves security for the requesting users, because someone can't easily impersonate them. You should require Azure MFA on all services,
especially on mobile services.

Azure MFA comes as part of the following offerings:

- Azure AD Premium licenses. These licenses include the full-featured use of the Azure MFA service (in the cloud) or Azure Multi-Factor Authentication Server (on-premises): The Azure MFA service (in the cloud). We recommend this choice for new deployments. Azure MFA in the cloud needs no on-premises infrastructure and can be used with your federated or cloud-only users.
  Azure Multi-Factor Authentication Server. If your organization wants to manage the associated infrastructure elements and has deployed Active Directory Federation Services (AD FS) in your on-premises environment, this might be a choice.
- Azure MFA for Office 365. A subset of the Azure MFA capabilities is available as part of your subscription. For more information about Azure MFA for Office 365, refer to **Plan for multi-factor authentication for Office 365 Deployments**[8].
- Azure AD Global administrator roles. A subset of the Azure MFA capabilities is available to help protect Global administrator accounts.

Other identity-related features of Azure AD are:

- Self-service password reset (SSPR).
- An SSPR link on the Windows 10 sign-in screen.
- Azure AD Identity Protection. To help protect your organization's identities, you can configure risk-based policies that automatically respond to risky behaviors. These policies can either automatically block the behaviors or initiate remediation, including requiring password changes and enforcing Azure MFA. For more information, refer to **What is Azure Active Directory Identity Protection?**[9].
- Azure AD password protection. You can block commonly used and compromised passwords via a globally banned-password list.
- Azure AD smart lockout. Smart lockout helps lock out malicious hackers who are trying to guess your users' passwords or use brute-force methods to get in. It recognizes sign-ins coming from valid users and treats them differently than the ones of malicious hackers and other unknown sources.

8 https://support.office.com/article/plan-for-multi-factor-authentication-for-office-365-deployments-043807b2-21db-4d5c-b430-c8a6dee0e6ba
9 https://docs.microsoft.com/en-us/azure/active-directory/active-directory-identityprotection

- Azure AD Application Proxy. You can provision security-enhanced remote access to on-premises web applications.
- Single sign-on (SSO) access to your applications. This includes thousands of pre-integrated SaaS apps.
- Azure AD Connect. Create and manage a single identity for each user across your hybrid enterprise, keeping users, groups, and devices in sync.

Exercise - MFA

Configure Azure MFA for applications

1. Sign in to the **Azure portal**[10] by using a Global administrator account.
2. Browse to **Azure Active Directory > Conditional access**.
3. Select **New policy**.
4. Name your policy.
5. Under **users and groups**, select **Select users and groups**, select a group, and then select **Done**.
6. Under **Cloud apps**, select **Select apps**, choose the cloud applications for which you want to enable Azure MFA, select **Select**, and then select **Done**.
7. Review the **Conditions** section, and then select the conditions you want.
8. Under **Grant**, make sure that **Grant access** is selected, select the **Require multi-factor authentication** check box, and then select **Select**.
9. Set **Enable policy** to **On**.
10. Select **Create**.

MFA is now enabled for selected applications.

Configure Azure MFA for passwords

1. In the Azure portal, open the **Azure Active Directory**[11] blade.
2. Select **Users**.
3. At the top of the **Users** blade, select **Multi-Factor Authentication**. The Azure MFA management portal opens.
4. Select **service settings**.
5. Scroll to **app passwords**, and then select the app passwords option you want to use.
6. Select **save**.

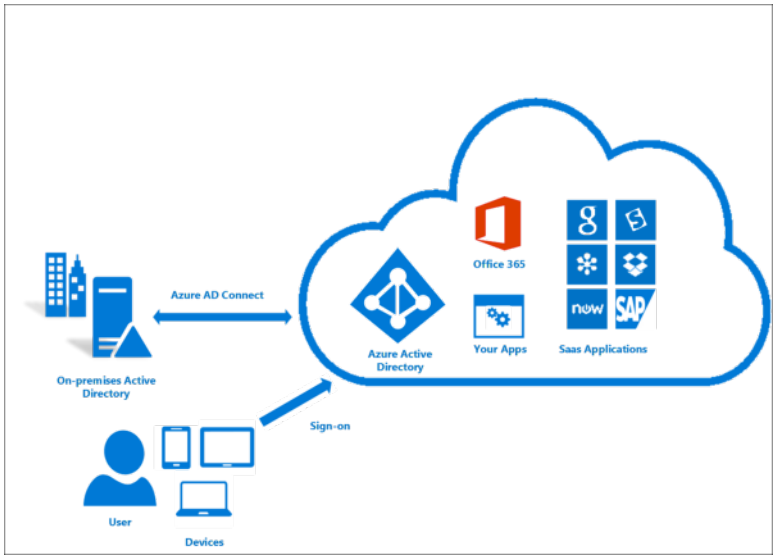
MFA is now enabled for users passwords.

Azure AD Connect

Azure AD Connect integrates your on-premises directories with Azure AD. With Azure AD Connect, you can provide your
users with a common identity for Office 365, Azure, and SaaS applications integrated with Azure AD in a hybrid identity environment.

10 https://portal.azure.com/
11 https://portal.azure.com/

To download Azure AD Connect for free, refer to "Microsoft Azure Active Directory Connect" at **https://www.microsoft.com/en-us/download/details.aspx?id=47594**.



Azure AD Connect includes:

- Sync services. This component is responsible for creating users, groups, and other objects. It also makes sure that identity information for your on-premises users and groups matches that in the cloud.
- Health monitoring. Azure AD Connect Health supplies robust monitoring and a central location in the Azure portal for viewing this activity.
- AD FS. Federation is an optional part of Azure AD Connect that you can use to configure a hybrid environment via an on-premises AD FS infrastructure. Organizations can use this to address complex deployments, such as domain join SSO, enforcement of the Active Directory sign-in policy, and smart card or third-party multi-factor authentication.
- Password hash synchronization. This feature is a sign-in method that synchronizes a hash of a user's on-premises Active Directory password with Azure AD.
- Pass-through authentication.

Integrating your on-premises directories with Azure AD makes your users more productive by supplying a common identity for accessing both cloud and on-premises resources. Users and organizations get the following advantages:

- Users can use a single identity to access both on-premises applications and cloud services, such as Office 365.
- A single tool provides an easy deployment experience for synchronization and sign-in.
- Integration provides the newest capabilities for your scenarios. Azure AD Connect replaces older versions of identity integration tools, such as DirSync and Azure AD Sync. For more information, see **Hybrid Identity directory integration tools comparison**[12].
- Installation guides and roadmap at **https://docs.microsoft.com/en-us/azure/active-directory/hybrid/how-to-connect-install-roadmap**.

12 https://docs.microsoft.com/en-us/azure/active-directory/hybrid/plan-hybrid-identity-design-considerations-tools-comparison

Manage Azure AD directory roles

Azure AD provides many **built-in roles**[13] to cover the most common security scenarios. To understand how the roles work, you'll examine three roles that apply to all resource types:

- Owner, which has full access to all resources, including the right to delegate access to others.
- Contributor, which can create and manage all types of Azure resources but can't grant access to others.
- Reader, which can view existing Azure resources.

Role definitions

Each role is a set of properties defined in a JavaScript Object Notation (JSON) file. This role definition includes **Name, Id**, and **Description**. It also includes the allowable permissions (**Actions**), denied permissions (**NotActions**), and scope (for example, read access) for the role.

For the Owner role, that means all actions, indicated by an asterisk (*); no denied actions; and all scopes, indicated by a forward slash (/).

You can get this information via the **Get-AzureRmRoleDefinition** cmdlet:

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

✓ Take a minute to open the Azure portal, open the Subscriptions or Resource Group blade, and then select Access Control (IAM).
Select **Add**, and then take a few minutes to review the built-in roles. *Which role are you most interested in using?*

Role definitions

Actions and NotActions

You can tailor the Actions and NotActions properties to grant and deny the exact permissions you need. Review the following table, which describes how Owner, Contributor, and Reader are defined.

Built-in Role	Actions	NotActions
Owner (allow all actions)	*	None

13 https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-built-in-roles

Built-in Role	Actions	NotActions
Contributor (allow all actions except writing or deleting role assignments)	*	Microsoft.Authorization//Delete, Microsoft.Authorization//Write, Microsoft.Authorization//elevateAccess/Action
Reader (allow all read actions)	*/read	None

AssignableScopes

Defining the **Actions** and **NotActions** properties is not enough to fully implement a role. You also need to properly scope your role.

The **AssignableScopes** property of the role specifies the scopes (subscriptions, resource groups, or resources) within which the custom
role is available for assignment. You can make the custom role available for assignment just in the subscriptions or resource groups that
need it, thus avoiding cluttering the user experience for the rest of the subscriptions or resource groups.

```
"/subscriptions/[subscription id]"
```

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

```
"/subscriptions/[subscription id]/resourceGroups/[resource group name]/[re-
source]"
```

Example 1

Make a role available for assignment in two subscriptions:

```
"/subscriptions/c276fc76-9cd4-44c9-99a7-4fd71546436e", "/subscriptions/
e91d47c4-76f3-4271-a796-21b4ecfe3624"
```

Example 2

Make a role available for assignment only in the Network resource group:

```
"/subscriptions/c276fc76-9cd4-44c9-99a7-4fd71546436e/resourceGroups/Network"
```

✓ Take a minute to open the Azure portal, use the **Access Control** blade to add a role, and then assign it to a user.
Which role assignments do you need for your organization?

For more information, refer to:

- "Custom roles for Azure resources" at **https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-control-custom-roles**.
- "Get-AzureRmRoleDefinition" at **https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/get-azurermroledefinition?view=azurermps-5.3.0**.

Configure authentication methods

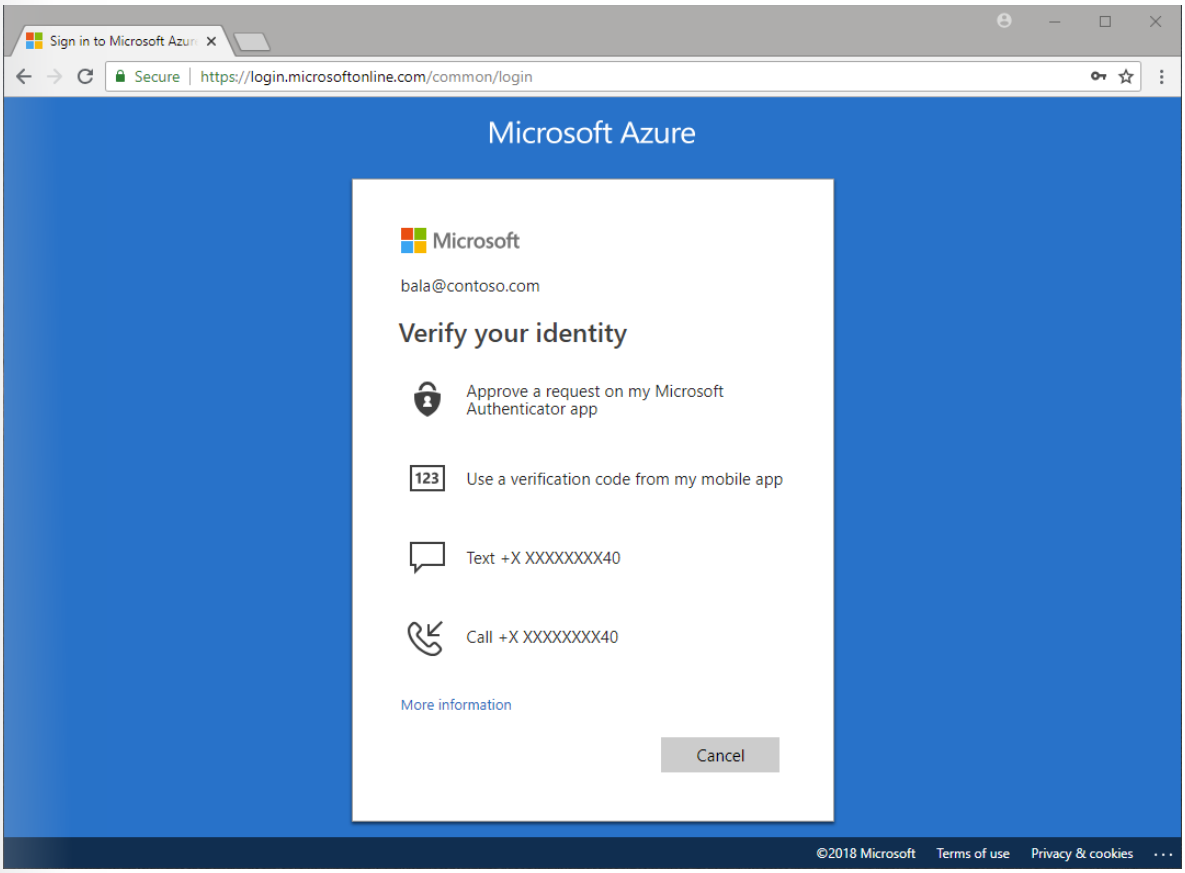
This module previously discussed Azure MFA and self-service password reset (SSPR) in Azure AD. These services might
require additional information to confirm your identity. This additional information is called an authentication method.

We recommend that administrators enable users to be able to select more than the minimum number of required authentication

methods in case they do not have access to a certain one. The following table lists the authentication methods and the services that use them.

Authentication method	Services
Password	Azure MFA and SSPR
Security questions	SSPR
Email address	SSPR
Microsoft Authenticator app	Azure MFA and SSPR
OATH hardware token	Azure MFA and SSPR
Text message	Azure MFA and SSPR
Vocie call	Azure MFA and SSPR
App passwords	Azure MFA in certain cases

The following figure depicts asking for the additional authentication information.



Azure authentication methods

- Password. This is the only method that you can't disable.
- Security questions. This method is available only for nonadministrative accounts that use SSPR:
- o Azure stores security questions privately and in a security-enhanced manner on a user object in the directory.
  Only users can answer the questions and only during registration. An administrator can't read or change a user's questions or answers.

  o Azure provides 35 predefined questions, all translated and localized

based on the browser locale.

o You can customize the questions by using the administrative interface; however, Azure displays them in the language entered.
The maximum length is 200 characters.

- Email address. This method is available only in SSPR. We recommend avoiding the use of an email account that doesn't require the user's Azure AD password to access it.
- Microsoft Authenticator app. This method is available for Android and iOS. Users can register their mobile app at **https://aka.ms/mfasetup**:

- o The Microsoft Authenticator app helps prevent unauthorized access to accounts and helps stop fraudulent transactions by pushing
a notification to your smartphone or tablet. Users view the notification and, if it's legitimate, select Verify. Otherwise, they select Deny.

o Users can use the Microsoft Authenticator app or a third-party app as a software token to generate an OATH verification code.
After entering the username and password, the users enter the code provided by the app on the sign-in screen. The verification code provides
a second form of authentication.

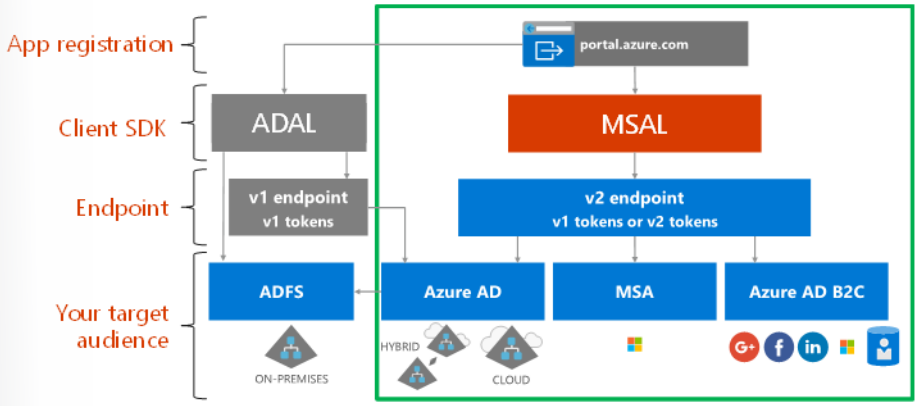
- OATH hardware tokens. **OATH**[14] is an open standard that specifies how to generate one-time password codes.
Azure AD supports the use of OATH-TOTP SHA-1 tokens of the 30-second or 60-second variety. Customers can get these tokens from the vendor of
their choice. Note that secret keys are limited to 128 characters, which might not be compatible with all tokens.
- Mobile phone. Two options are available: text message and phone call.
- App password. Certain non-browser apps don't support Azure MFA. If users are enabled for Azure MFA and try to use non-browser apps,
they'll be unable to authenticate. The app password allows users to continue to authenticate. (Note that configuration considerations exist when
using federated SSO with an app password. You can find additional information **here**[15].)

Manage app registration

The Microsoft identity platform is an evolution of the Azure AD identity service and developer platform. It allows developers
to build applications that sign in all Microsoft identities and get tokens to call Microsoft Graph, other Microsoft APIs, or
APIs that developers have built. It's a full-featured platform that consists of an authentication service, open-source libraries,
application registration and configuration, full developer documentation, code samples, and other developer content. The Microsoft
identity platform supports industry standard protocols such as Open Authorization (OAuth) 2.0 and OpenID Connect.

14 https://openauthentication.org/
15 https://docs.microsoft.com/en-us/azure/active-directory/authentication/concept-authentication-methods

The following diagram depicts the Microsoft identity experience at a high level, including the app registration experience, software
development kits (SDKs), endpoints, and supported identities.



The Microsoft identity platform has two endpoints (v1.0 and v2.0) and two sets of client libraries to handle these endpoints. When developing a new
application, consider the advantages and the current state of the endpoints and the authentication libraries. Also consider that:

- The supported platforms are as follows:
  - The Azure AD Authentication Library (**ADAL**[16])
    supports Microsoft .NET, JavaScript, iOS, Android, Java, and Python.
  - The Microsoft Authentication Library (**MSAL**[17])
    Preview supports .NET, JavaScript, iOS, and Android.
  - other endpoints support .NET and Node.js server middleware for protecting APIs and sign-in.
- The bulk of innovation, such as dynamic consent and incremental consent, is happening on the v2.0 endpoint and MSAL while Microsoft
  continues to support v1.0 and ADAL.

These are the five primary application scenarios that Azure AD supports:

- **Single-page application (SPA)**[18]. A user needs to sign in to a
  single-page application that Azure AD helps secure.
- **Web browser to web application**[19]. A user needs to sign in to a web
  application that Azure AD helps secure.
- **Native application to web API**[20]. A native application that runs
  on a phone, tablet, or computer needs to authenticate a user to get resources from a web API that Azure AD helps secure.
- **Web application to web API**[21]. A web application needs to get resources from
  a web API that Azure AD helps secure.

16 https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-authentication-libraries
17 https://docs.microsoft.com/en-us/azure/active-directory/develop/reference-v2-libraries
18 https://docs.microsoft.com/en-us/azure/active-directory/develop/single-page-application
19 https://docs.microsoft.com/en-us/azure/active-directory/develop/web-app
20 https://docs.microsoft.com/en-us/azure/active-directory/develop/native-app
21 https://docs.microsoft.com/en-us/azure/active-directory/develop/web-api

- **Daemon or server application to web API**[22]. A daemon application or a server application with no web user interface needs to get resources from a web API that Azure AD helps secure.

App registration

Registration of an app that uses the Azure AD v1.0 endpoint

Any application that outsources authentication to Azure AD needs to be registered in a directory. This step involves telling Azure AD about your application, including the URL where it's located, the URL to send replies to after authentication, the URI to identify your application, and more. For details, learn how to **register an app with the Azure AD v1.0 endpoint**[23].

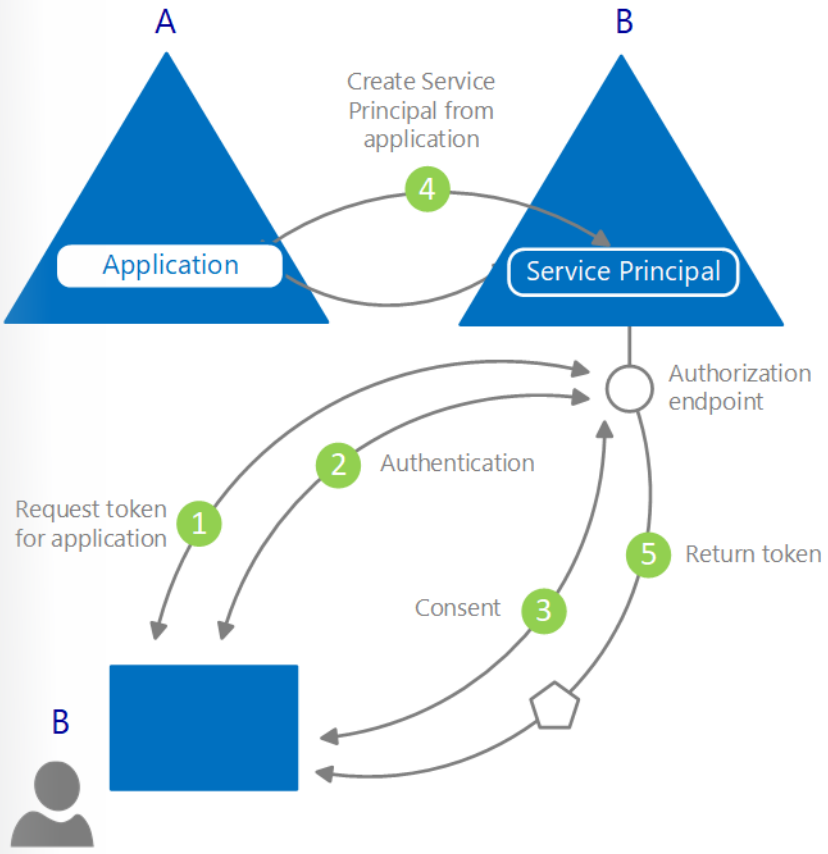
Azure AD represents applications following a specific model that's designed to fulfill two main functions:

- Identify the app according to the authentication protocols it supports. This involves enumerating all the identifiers, URLs, secrets, and related information that Azure AD needs at authentication time. Here, Azure AD:
  - Holds all the data needed to support authentication at run time.
  - Holds all the data for deciding which resources an app might need to access, whether it should fulfill a particular request, and under what circumstances it should fulfill the request.
  - Supplies the infrastructure for implementing app provisioning both within the app developer's tenant and to any other Azure AD tenant.
- Handle user consent during token request time and facilitate the dynamic provisioning of apps across tenants. Here, Azure AD:
  - Enables users and administrators to dynamically grant or deny consent for the app to access resources on their behalf.
  - Enables administrators to ultimately decide what apps are allowed to do, which users can use specific apps, and how directory resources are accessed.

In Azure AD, an application object describes an application as an abstract entity. Developers work with applications. At deployment time, Azure AD uses a specific application object as a blueprint to create a service principal, which represents a concrete instance of an application within a directory or tenant. It's the service principal that defines what the app can do in a specific target directory, who can use it, what resources it has access to, and so on. Azure AD creates a service principal from an application object through consent.

The following diagram depicts a simplified Azure AD provisioning flow driven by consent.

22 https://docs.microsoft.com/en-us/azure/active-directory/develop/service-to-service
23 https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-v1-add-azure-ad-app



In this provisioning flow:

1. A user from B tries to sign in with the app.
2. Azure AD gets and verifies the user credentials.
3. Azure AD prompts the user to consent for the app to gain access to tenant B.
4. Azure AD uses the application object in A as a blueprint for creating a service principal in B.
5. The user receives the requested token.

You can repeat this process as many times as you want for other tenants (C, D, and so on). Directory A keeps the blueprint for the app (application object). Users and admins of all the other tenants where the app is given consent to retain control over what the application can do through the corresponding service principal object in each tenant. For more information, refer to **Application and service principal objects in Azure Active Directory**[24].

If you're interested in software development, these **quickstart tutorials**[25] take you through building an app and adding functionality like tokens, signing in users, displaying user info, and more.

24 https://docs.microsoft.com/en-us/azure/active-directory/develop/app-objects-and-service-principals
25 https://docs.microsoft.com/en-us/azure/active-directory/develop/

Security for an Azure subscription

Configure custom role-based access control in Azure

When it comes to identity and access, most organizations that are considering using the public cloud are concerned about two things:

- Ensuring that when people leave the organization, they lose access to resources in the cloud.
- Striking the right balance between autonomy and central governance—for example, giving project teams the ability to create and manage virtual machines in the cloud while centrally controlling the networks to which those virtual machines connect.

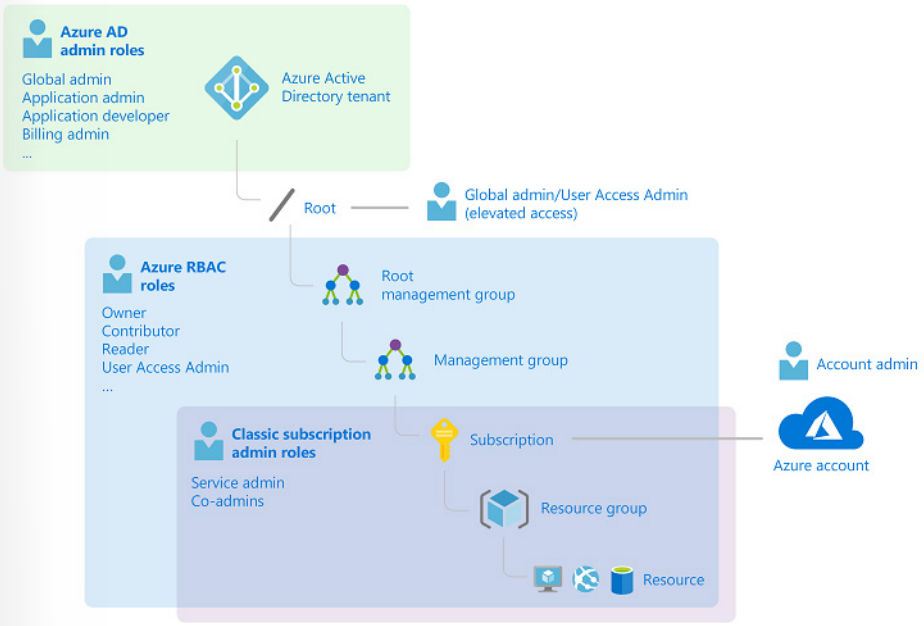
Azure AD and RBAC make it simple for you to carry out these goals. After you extend your on-premises Active Directory to the cloud by using Azure AD Connect, your employees can use and manage their Azure subscriptions by using their existing work identities. These Azure subscriptions automatically connect to Azure AD for SSO and access management. When you disable an on-premises Active Directory account, it automatically loses access to all Azure subscriptions connected with Azure AD. RBAC enables fine-grained access management for Azure.
Using RBAC, you can grant just the amount of access that users need to perform their jobs. For example, you can use RBAC to let one employee manage virtual machines in a subscription while another manages SQL databases within the same subscription.

Each Azure subscription is associated with one Azure AD directory. Users, groups, and applications in that directory can manage resources in the Azure subscription. Grant access by assigning the appropriate RBAC role to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource. A role assigned at a parent scope also grants access to the child scopes contained within it. For example, a user with access to a resource group can manage all the resources it contains, like websites, virtual machines, and subnets. The RBAC role that you assign dictates what resources the user, group, or application can manage within that scope. The scope of a role assignment can be a subscription, a resource group, or a single resource.

RBAC includes many **built-in roles**[26], which you can assign at different scopes, and allows you to create your own custom roles. To manage resources in Azure AD, such as users, groups, and domains, several Azure AD administrator roles exist.

The following diagram depicts how the classic subscription administrator roles, RBAC roles, and Azure AD administrator roles are related at a high level. Roles assigned at a higher scope, like an entire subscription, are inherited by child scopes, like service instances.

26 https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles

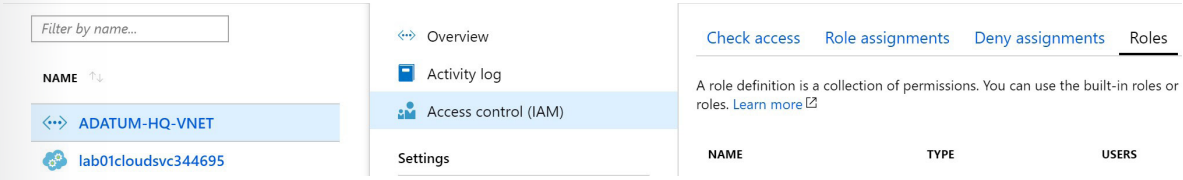


In the preceding diagram, a subscription is associated with only one Azure AD tenant. Also note that a resource group can have
multiple resources but is associated with only one subscription. Although it's not obvious from the diagram, a resource can be bound to only one resource group.

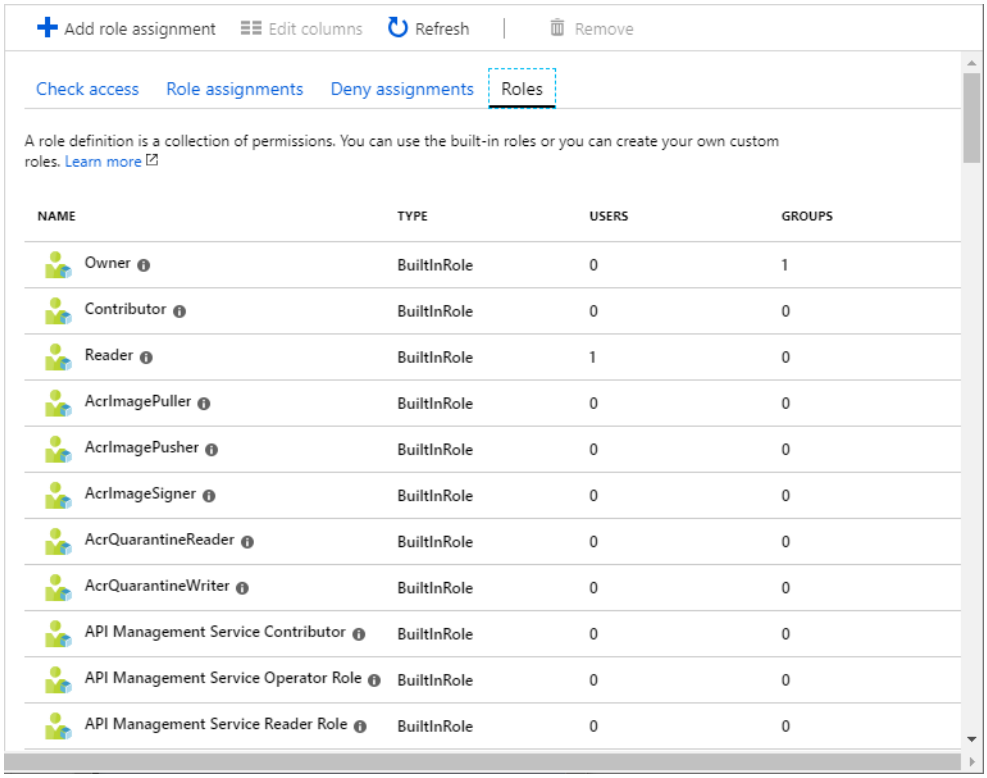
Exercise - Azure Roles

Exercise Azure Roles

1. Using a web browser, open the Azure portal. Select **All services**.
2. Select any resource, and the select it again.
3. Select **Access control (IAM)**.
4. Select **Roles**.



1. Review the list of built-in roles. Note that you can also create custom roles.



Here are four RBAC roles in Azure that apply to resource types:

- Owner. Has full access to all resources, including the right to delegate access to others.
- Contributor. Can create and manage all types of Azure resources but can't grant access to others.
- Reader. Can view existing Azure resources.
- User access administrator. Can remove access to resources.

The rest of the RBAC roles in Azure allow for managing specific Azure resources. For example, the Virtual Machine Contributor role allows the user to create and manage virtual machines. It does not grant access to the virtual network or the subnet that the virtual machine connects to.

As a best practice when deploying Azure RBAC, consider creating new resource groups instead of new subscriptions for newly onboarded teams.

Resource groups allow you to implement RBAC so that users can contribute to services but not own them.

To manage access by using RBAC and Azure PowerShell, review the documentation page at **https://docs.microsoft.com/en-us/azure/role-based-access-control/role-assignments-powershell**. It has examples of how to:

- List roles
- List access
- Grant access
- Remove access

Configure subscription and resource permissions

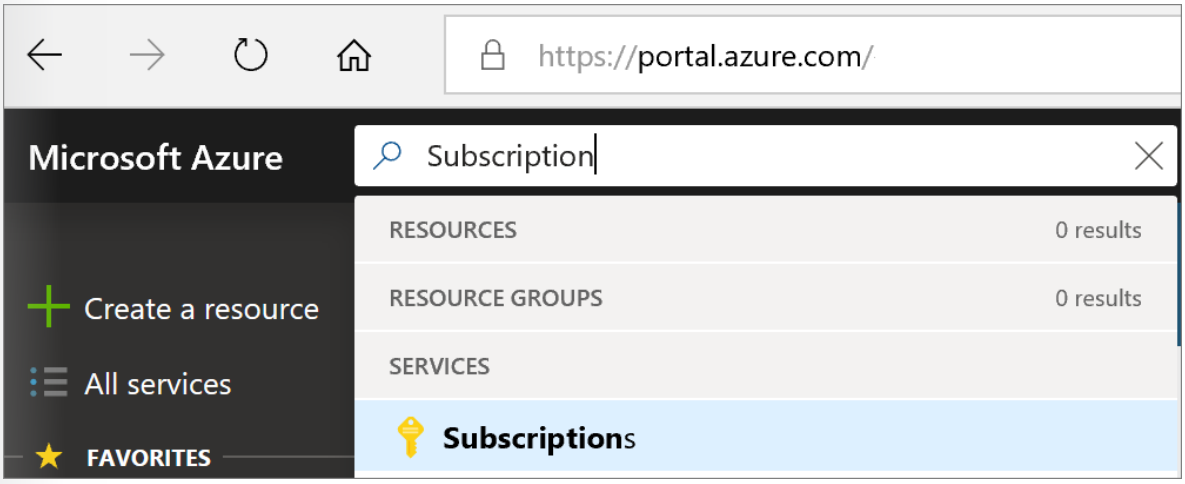
Configure subscription and resource permissions

You can create additional subscriptions for your account in Azure. You might want an additional subscription to avoid reaching
subscription limits, to create separate environments for billing and security, or to isolate data for compliance reasons.

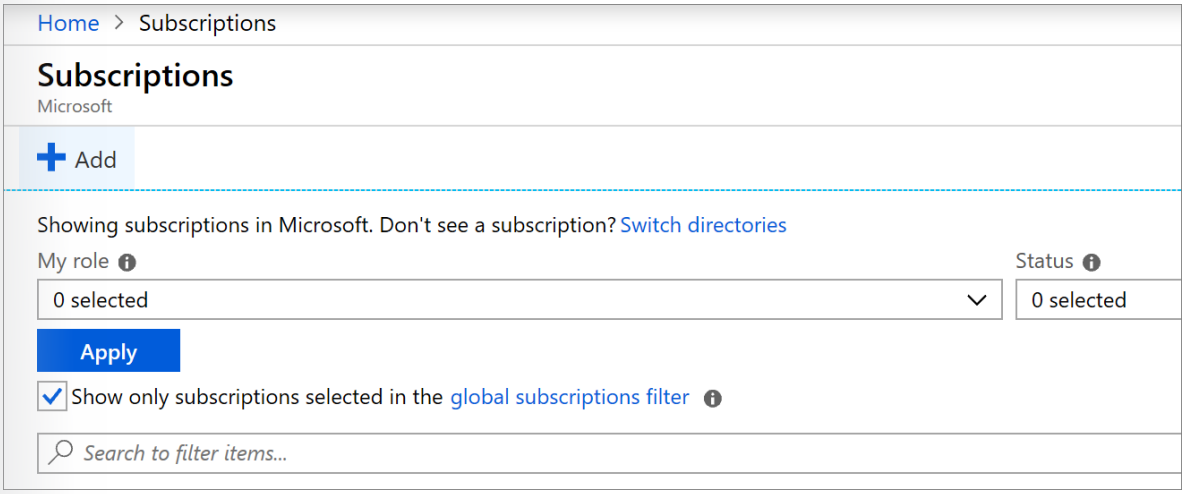
If you want to create Azure subscriptions under your organization's Enterprise Agreement (EA), you need to have the Account
Owner role for your organization.

Create an additional Azure subscription

1. Sign in to the **Azure portal**[27].
2. Search for **Subscriptions**.



1. Select **Add**.



27 https://portal.azure.com/

As discussed earlier, **RBAC**[28] is how you manage
access to resources in Azure. The following section details some of the common tasks.

List all available roles

To list the RBAC roles that are available for assignment and to inspect the operations that they grant access to,
use **Get-AzureRmRoleDefinition**[29]:

```
Get-AzureRmRoleDefinition | FT Name, Description
```

List a specific role

To list a specific role, use **Get-AzureRmRoleDefinition**[30] <role name>:

```
PS C:\> Get-AzureRmRoleDefinition "Contributor"
```

List the actions of a role

To list the actions of a role, use Get-AzureRmRoleDefinition <role name> | FL Actions, NotActions:

```
PS C:\> Get-AzureRmRoleDefinition "Contributor" | FL Actions, NotActions
```

List access

To list access in RBAC, list the role assignments.

You can list all the role assignments for a specified subscription, resource group, or resource. For example, to list all the active assignments for a resource group, use **Get-AzureRmRoleAssignment**[31]:

```
Get-AzureRmRoleAssignment -ResourceGroupName resource group name
PS C:\> Get-AzureRmRoleAssignment -ResourceGroupName pharma-sales-project-
forecast | FL DisplayName, RoleDefinitionName, Scope
```

Identify external accounts

Identify external accounts that have Azure management access

This module hasn't discussed Azure Security Center, but a later module will discuss it. Security Center is free and is now installed
on virtual machines created in Azure.

Security Center is a unified infrastructure security management system that strengthens the security posture of your datacenters and
provides advanced threat protection across your hybrid workloads in the cloud—whether they're in Azure—and on-premises.

Security Center helps you, as an Azure administrator, identify external accounts and the roles associated with those accounts.
Security Center helps you identify shadow IT subscriptions. By finding subscriptions labeled not covered on your dashboard, you'll
immediately know when newly created subscriptions exist, and you can make sure they'll be covered by your policies and that Security
Center will help protect them.

Azure Policy is a service you use to create, assign and, manage policies. These policies enforce different rules and effects over your

28 https://docs.microsoft.com/en-us/azure/role-based-access-control/overview
29 https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/get-azurermroledefinition
30 https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/get-azurermroledefinition
31 https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/get-azurermroleassignment

resources so that those resources stay compliant with your corporate standards and service level agreements. Azure Policy meets this need by evaluating your resources for noncompliance with assigned policies. For example, you might have a policy that allows virtual machines of only a certain size in your environment. After this policy is implemented, new and existing resources are evaluated for compliance. With the right type of policy, existing resources can be brought into compliance.

By default, Azure has set security policies that work across subscriptions or on management groups. If these policies need to be augmented with your own organizational policies, new policies can be created.

How do Azure Policy and RBAC differ?

A few key differences between Azure Policy and RBAC exist. RBAC focuses on user actions at different scopes. You might be added to the contributor role for a resource group, allowing you to make changes to that resource group. Azure Policy focuses on resource properties during deployment and for already-existing resources. Azure Policy controls properties such as the types or locations of resources.
Unlike RBAC, Azure Policy is a default-allow-and-explicit-deny system.

Azure Policy has several permissions, known as operations, in two resource providers:

- **Microsoft.Authorization**[32]
- **Microsoft.PolicyInsights**[33]

Many built-in roles grant permissions to Azure Policy resources. The Resource Policy Contributor role includes most Azure Policy operations.
The Owner role has full rights. Both Contributor and Reader can use all Azure Policy read operations, but Contributor can also trigger remediation.

If none of the built-in roles have the required permissions, create a **custom role**[34].

The following table describes just three of the built-in policies that Azure Security Center monitors.

Policy	Reason for the policy
In preview: Audit external accounts with owner permissions on a subscription	External accounts with owner permissions should be removed from your subscription to prevent unmonitored access.
In preview: Audit external accounts with write permissions on a subscription	External accounts with write permissions should be removed from your subscription to prevent unmonitored access.
In preview: Audit external accounts with read permissions on a subscription	External accounts with read permissions should be removed from your subscription to prevent unmonitored access.

Those three policies allow for the easy recognition of external accounts that have management access. For a list of the built-in Azure security policies, refer to **https://docs.microsoft.com/en-us/azure/security-center/security-center-policy-definitions**.

32 https://docs.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations
33 https://docs.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations
34 https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles

To learn how to assign a policy by using Azure PowerShell, work through the exercise at **https://docs.microsoft.com/en-us/azure/governance/policy/assign-policy-powershell**.

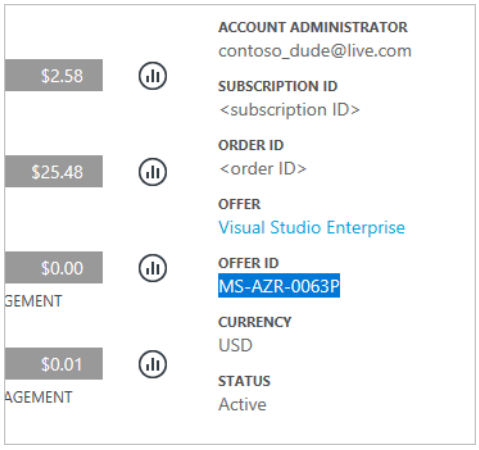
Transfer Azure subscriptions

Transfer Azure subscriptions between Azure AD tenants

Many large organizations buy their Azure subscriptions through Enterprise Agreements (EAs). Typically, they assign their subscriptions
to various business units in the company.

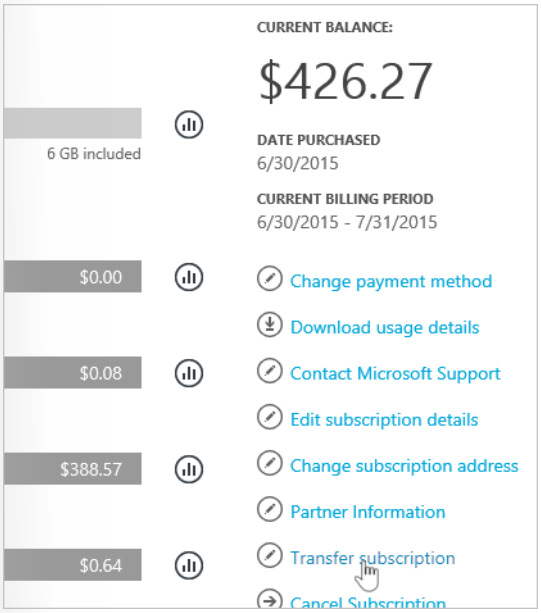
Occasionally, a need arises for transferring a subscription from an owner to an Azure AD tenant. To transfer the ownership of an Azure subscription:

1. Sign in at the **Azure Account Center**[35] as the account admin.
2. Select the subscription to transfer.
3. Verify that your subscription is eligible for self-serve transfer by checking the **Offer** and **Offer ID** against the **supported offers list**[36].



1. Select **Transfer subscription**.

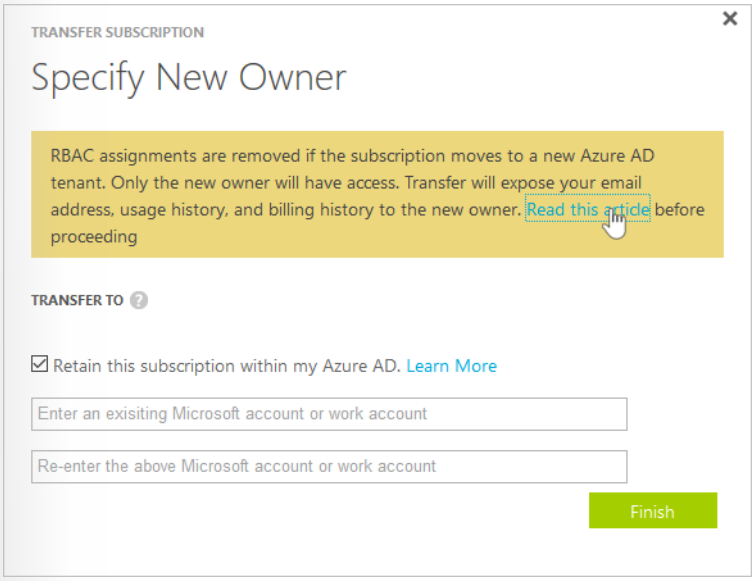
35 https://account.windowsazure.com/Subscriptions
36 https://docs.microsoft.com/en-us/azure/billing/billing-subscription-transfer



1. Specify the recipient.

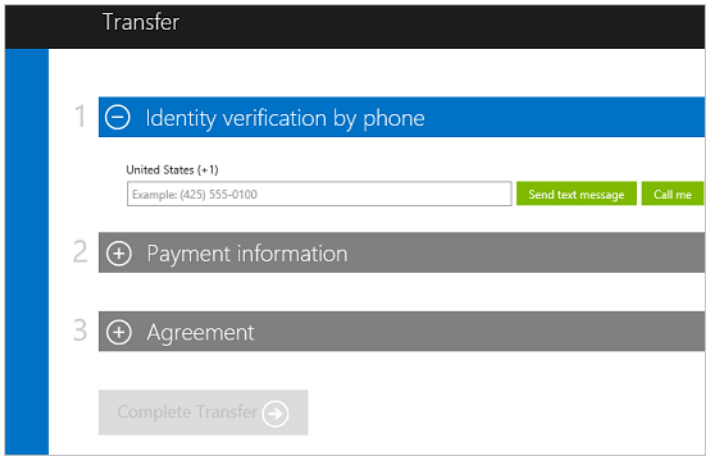
Note:

If you transfer a subscription to a new Azure AD tenant, all role assignments in **RBAC**[37] will be permanently deleted from the source tenant and not migrated to the target tenant.



1. The recipient automatically gets an email with an acceptance link.
2. The recipient selects the link and follows the instructions, including entering their payment information.

37 https://docs.microsoft.com/en-us/azure/role-based-access-control/overview



1. Azure completes the subscription transfer.

Suggested next steps are **here**[38].

Manage API access to Azure subscriptions and resources

When you publish APIs through API Management, it's easy and common to gain access to those APIs by using subscription keys.
Client applications that consume the published APIs need to include a valid subscription key in HTTP requests when they make
calls to those APIs. To get a subscription key for accessing APIs, a subscription is required. A subscription is essentially
a named container for a pair of subscription keys. Developers who need to consume the published APIs can get subscriptions,
and they don't need approval from API publishers. API publishers can also directly create subscriptions for API consumers.

Note:

API Management supports additional mechanisms for gaining access to APIs, including:

- **OAuth 2.0**[39]
- **Client certificates**[40]
- **IP whitelisting**[41]

Azure policies encapsulate common API management functions, like those for access control, protection, transformation, and caching.
You can chain these policies together into a pipeline that mutates a request's context or changes the API behavior. You can apply
these policies to a variety of scopes, trigger them on an error, and set them in the inbound and outbound directions.

For a list of inbound and outbound policies, go **here**[42].

38 https://docs.microsoft.com/en-us/azure/billing/billing-subscription-transfer
39 https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-protect-backend-with-aad
40 https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-mutual-certificates-for-clients
41 https://docs.microsoft.com/azure/api-management/api-management-access-restriction-policies
42 https://docs.microsoft.com/en-us/azure/api-management/policy-samples

Module 2 Implement Platform Protection

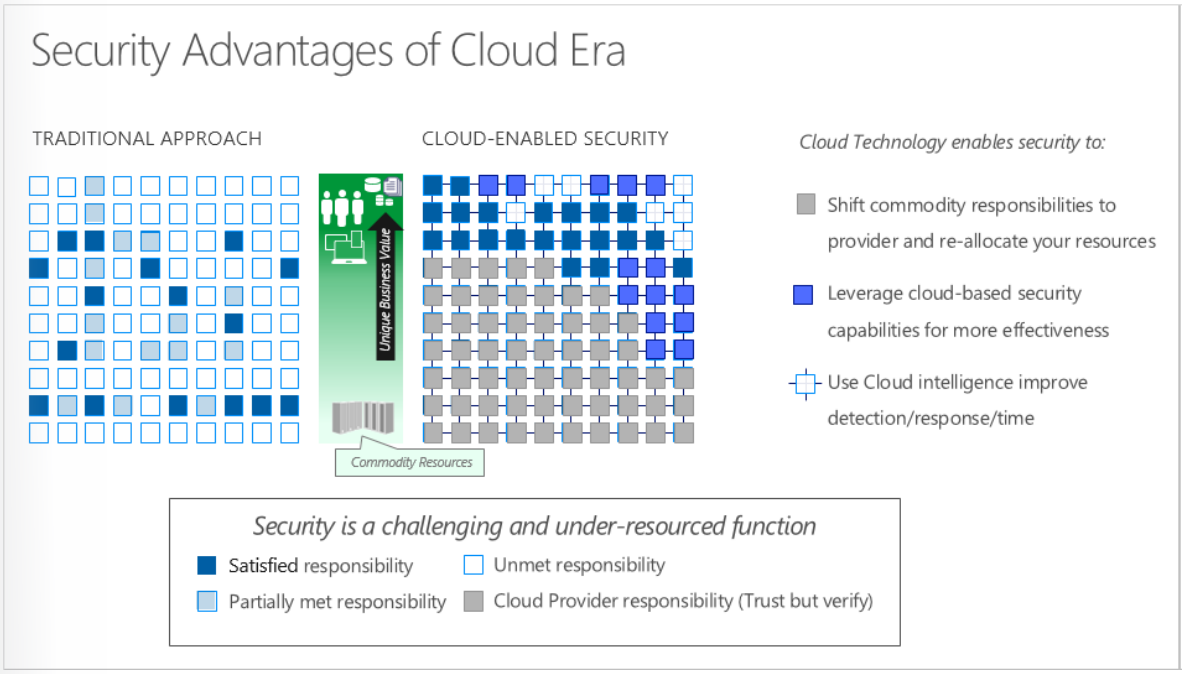
Understand cloud security

The shared responsibility model

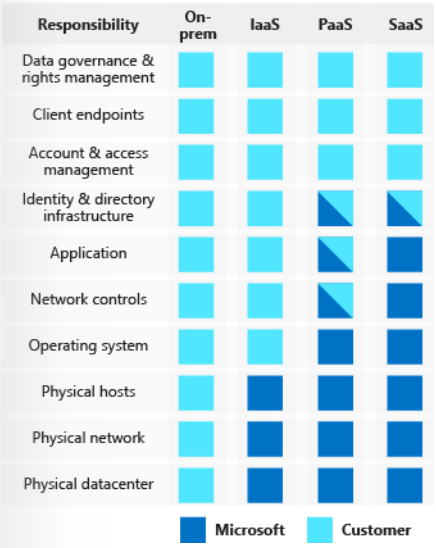
Understand the shared responsibility model: Introduction

Organizations face many challenges with securing their datacenters, including recruiting and keeping security experts, using many
security tools, and keeping pace with the volume and complexity of threats.

Microsoft Azure is uniquely positioned to help organizations with these challenges. Azure helps protect business assets while
reducing security costs and complexity. Built-in security controls and intelligence help admins easily find and respond to threats
and security gaps, so organizations can rapidly improve their security posture. By shifting responsibilities to Azure, organizations
can get more security coverage—which allows them to move security resources and budget allocations to other business priorities.
The following figure depicts this point.

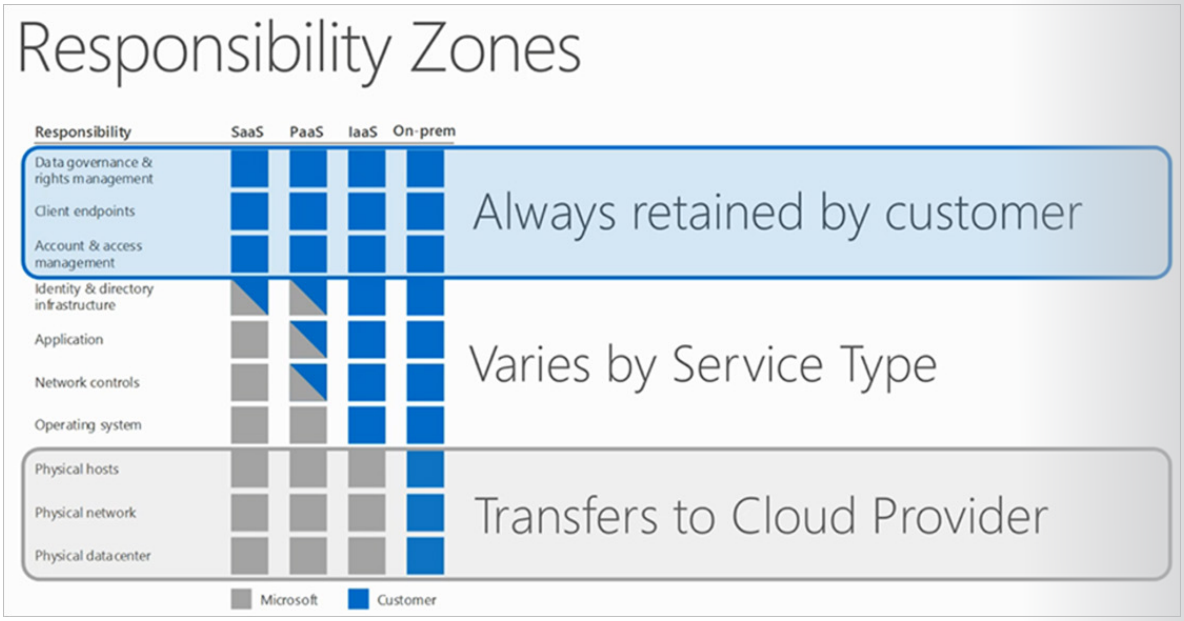


The first thing to understand about cloud security is that different scopes of responsibility exist, depending on the kinds of services you use.



For example, if you use virtual machines (VMs) in Azure, which provide Infrastructure as a Service (IaaS), Microsoft will be
responsible for helping secure the physical network, physical storage, and virtualization platform, which includes updating the
virtualization hosts. But you'll need to take care of helping secure your virtual network and public endpoints and updating the
guest operating system (OS) of your VMs.

The following figure depicts the various responsibility zones.



For all cloud deployment types, you own your data and identities. You are responsible for helping secure your data and identities, your on-premises resources, and the cloud components you control (which vary by service type).

Regardless of the deployment type, you always retain responsibility for the following:

- Data
- Endpoints
- Accounts
- Access management

It's important to understand the division of responsibility between you and Microsoft in a Software as a Service (SaaS), Platform as a Service (PaaS), or IaaS deployment.

Understand elasticity and scalability

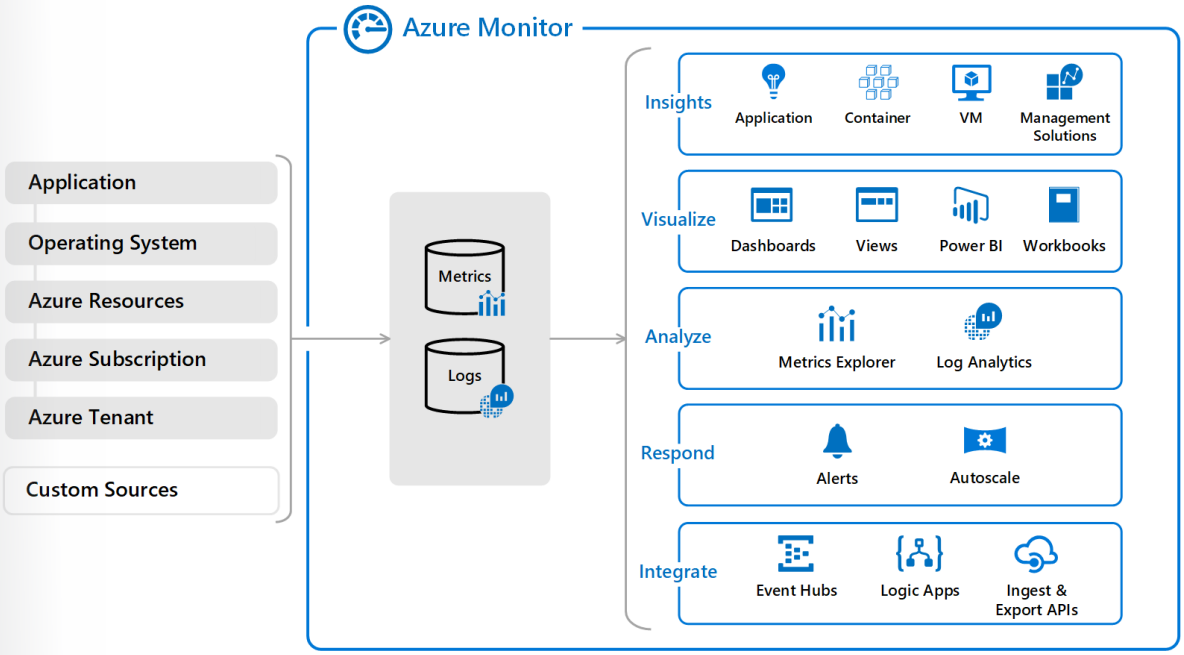
The cloud needs a new method of management because of its focus on scale, elasticity, and automation. The cloud represents a paradigm shift in the way people think about embracing IT.

DevOps has completely changed the way applications are developed and maintained. The hyperscale nature of the cloud provides a new meaning to scalability, elasticity, and resiliency and has redefined how applications are designed and delivered.

Cloud applications typically encounter variable workloads and peaks in activity. Applications should be able to scale out within limits to meet peaks in demand and scale in when demand decreases. Scalability concerns not just compute instances but also other elements, such as data storage and messaging infrastructure. This automatic resource allocation and deallocation is what defines the elasticity of the cloud.

One of the Azure tools you use to understand how your applications are performing is Azure Monitor. Azure Monitor maximizes the
availability and performance of your applications by delivering a comprehensive solution for collecting, analyzing, and acting on
telemetry from your cloud and on-premises environments.

The following diagram depicts a high-level view of Azure Monitor. At the center of the diagram are the data stores for metrics and
logs, which are the two fundamental types of data that Azure Monitor uses. On the left side are the **sources of monitoring data**[1] that populate these **data stores**[2]. On the right side are the different functions that Azure Monitor performs with this collected data, such as analysis, alerting, and streaming to external systems.



Azure Monitor can collect data from a variety of sources. You can think of monitoring data for your applications as occurring
in tiers that range from your application to any OS and the services it relies on to the platform itself. Azure Monitor collects
data from each of the following tiers:

- Application monitoring data. Data about the performance and functionality of the code you have written, regardless of its platform.
- Guest OS monitoring data. Data about the OS on which your application is running. It might be running in Azure, in another cloud, or on-premises.
- Azure resource monitoring data. Data about the operation of an Azure resource.
- Azure subscription monitoring data. Data about the operation and management of an Azure subscription and data about the health
and operation of Azure itself.

1 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-sources
2 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-collection

- Azure tenant monitoring data. Data about the operation of tenant-level Azure services, such as Azure Active Directory (Azure AD).

As soon as you create an Azure subscription and start adding resources, such as VMs and web apps, Azure Monitor starts collecting data. **Activity logs**[3] record when resources are created or modified. **Metrics**[4] tell you how the resource is performing and the resources that it's consuming.

Extend the data you're collecting into the actual operation of the resources by **enabling diagnostics**[5] and **adding an agent**[6] to compute resources. This will collect telemetry for the internal operation of the resource and allow you to configure different **data sources**[7] to collect logs and metrics from Windows and Linux guest OSs.

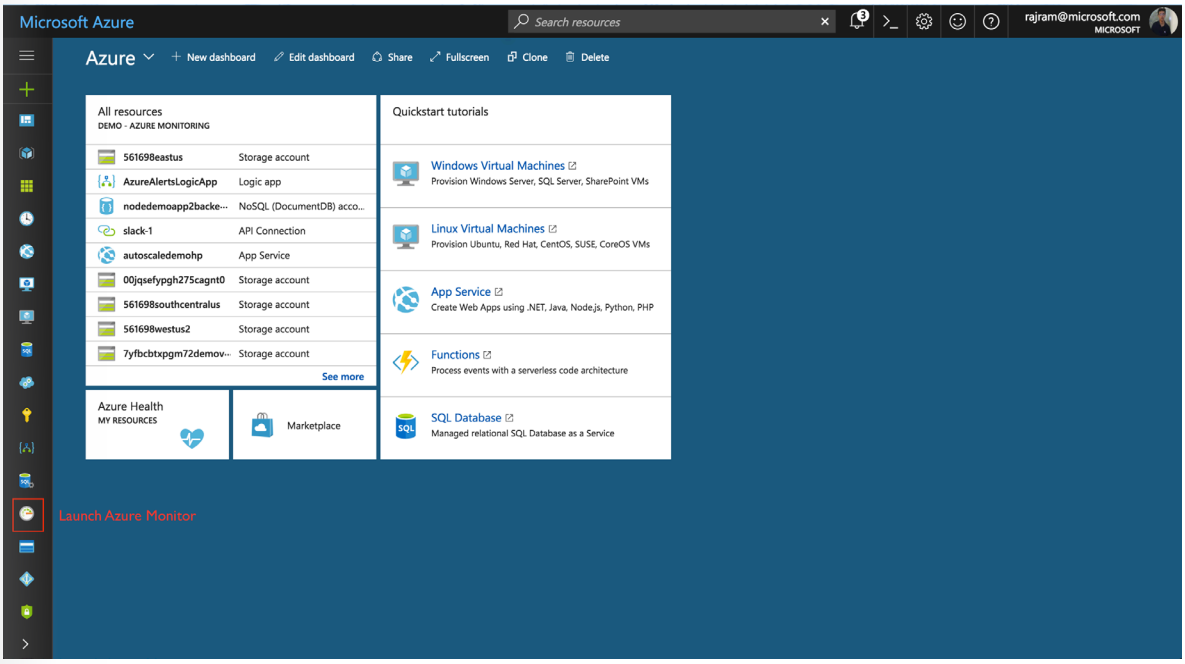
Add an instrumentation package to your application[8] to enable Application Insights to collect detailed information about your application, including page views, application requests, and exceptions. Further verify the availability of your application by configuring an **availability test**[9] to simulate user traffic.

Autoscale in Azure Monitor helps to enable the elastic scaling feature of the cloud. It allows you to have the right amount of resources running to handle the load on your application. It allows you to create rules that use metrics collected by Azure Monitor for two purposes: to determine when to automatically add resources to handle increases in the load and to save money by removing idle resources. You specify a minimum and a maximum number of instances and the logic for when to increase or decrease resources.

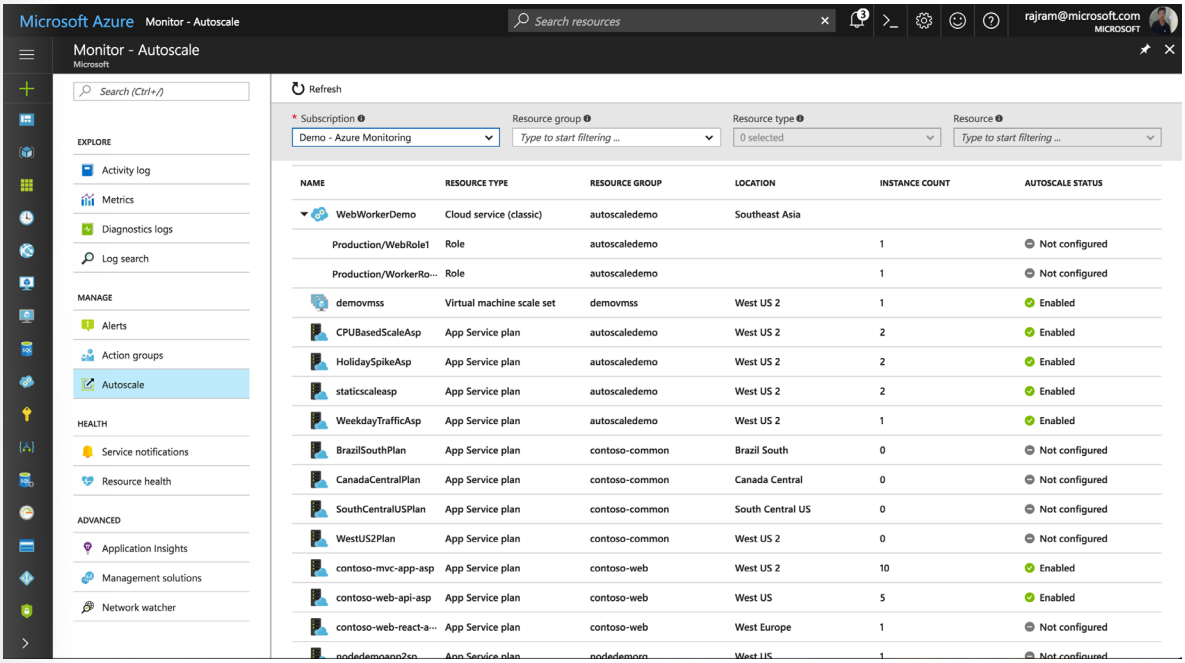
In Azure Monitor, you can discover all the resources that autoscale applies to. Use the following steps:

1. Open the **Azure portal**[10].
2. In the navigation pane on the left side, select the **Azure Monitor** icon.

3 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/activity-logs-overview
4 https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-metrics
5 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/diagnostic-logs-overview
6 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/agent-windows
7 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/agent-data-sources
8 https://docs.microsoft.com/en-us/azure/azure-monitor/app/azure-web-apps
9 https://docs.microsoft.com/en-us/azure/azure-monitor/app/monitor-web-app-availability
10 https://portal.azure.com/



1. Select **Autoscale** to view all the resources that autoscale applies to along with the current autoscale status of each.



You can use the filter pane at the top to select either resources in a specific resource group, specific resource types, or a specific resource.

For each resource, you will find the current instance count and the autoscale status. The autoscale status can be:

- **Not configured**. You have not yet enabled autoscale for this resource.
- **Enabled**. You have enabled autoscale for this resource.

- **Disabled**. You have disabled autoscale for this resource.

Exercises

Step through the following labs:

- "Create your first Autoscale setting," located **here**[11].
- "Scale differently on specific dates," located **here**[12].

virtualization, containers, and serverless computing

Virtualization

Virtualization creates a simulated, or virtual, computing environment as opposed to a physical computing environment. Virtualization often includes computer-generated versions of hardware, OSs, storage devices, and more. This allows organizations to partition a single physical computer or server into several **VMs**[13]. Each VM can then interact independently and run different OSs or applications while sharing the resources of a single host machine.

By creating multiple resources from a single computer or server, virtualization improves scalability and workloads while resulting in the use of fewer overall servers, less energy consumption, and less infrastructure cost and maintenance. Virtualization falls into four main categories. The first is desktop virtualization, which allows one centralized server to deliver and manage individualized desktops. The second is network virtualization, which is designed to split network bandwidth into independent channels to then be assigned to specific servers or devices. The third is software virtualization, which separates applications from the hardware and OS. The fourth is storage virtualization, which combines multiple network storage resources into a single storage device that multiple users can access.

An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs the VM. However, you still need to maintain the VM—that is, configure, update, and maintain the software that runs on the VM.

Azure Virtual Machines lets you create and use VMs in the cloud. Providing what's known as Infrastructure as a Service (IaaS), VM technology can be used in variety of ways.

Examples of when to use virtual machines

- During testing and development. VMs provide a quick and easy way to create different OS and application configurations. Test and development personnel can then easily delete the VMs when they no longer need them.
- When running applications in the cloud. The ability to run certain applications in the public cloud as opposed to creating a traditional infrastructure to run them can provide substantial economic benefits. For example, if an application

11 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/autoscale-get-started
12 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/autoscale-get-started
13 https://azure.microsoft.com/en-us/overview/what-is-a-virtual-machine/

needs to handle fluctuations in demand,
being able to shut down VMs when you don't need them or quickly start them up to meet a suddenly increased demand means you pay only for the
resources you use.

- When extending your datacenter to the cloud. An organization can extend the capabilities of its own on-premises network by creating a virtual
network in Azure and adding VMs to that virtual network. Applications like SharePoint can then run on an Azure VM instead of running locally,
making it easier or less expensive to deploy than in an on-premises environment.
- During disaster recovery. As with running certain types of applications in the cloud and extending an on-premises network to the cloud, you
can get significant costs savings by using an IaaS-based approach to disaster recovery. If a primary datacenter fails, you can create VMs
running on Azure to run your critical applications and then shut them down when the primary datacenter becomes operational again.

For more information about typical scenarios for running Azure VMs, refer to this **article** [14] in the Azure documentation.

Serverless computing

Previously, this module discussed responsibility zones for Azure deployments. With serverless computing based on Azure Functions, you can
significantly reduce the customer's responsibility. Serverless computing is the abstraction of servers, infrastructure, and OSs. When developers
build serverless apps, they don't need to provision and manage any servers, so they can take their minds off infrastructure concerns. The reaction
to events that happen in near real time drive serverless computing—in the cloud. With a fully managed service, server management and capacity planning
are invisible to the developer, and billing is based just on the resources consumed or the actual time the code runs. With serverless architecture,
you simply deploy your code, which then runs with high availability.

Azure Functions

Azure Functions is a serverless application platform. It allows developers to host business logic that can run without the need to provision infrastructure.
Azure Functions provides intrinsic scalability, and the customer is charged only for the resources used. Developers can write their function code in the
language of their choice, including C#, F#, and JavaScript. Azure Functions also includes support for NuGet and npm, so developers can use popular libraries
in their business logic.

Functions are event driven, which means they run only in response to an event, or trigger, such as receiving an HTTP request or having a message added to a queue.
The developer configures a trigger as part of the function definition. This approach simplifies the code by allowing the developer to declare where the data
comes from (trigger/input binding) and where it goes (output binding). Developers don't need to write code to watch queues, binary large objects (blobs),
event hubs, or other structures. They can focus purely on the business logic.

14 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/overview?toc=%2Fazure%2Fvirtual-machines%2Fwindows%2Ftoc.json

Functions are a key component of serverless computing, but they're also a general compute platform for running any type of code. If the needs of the developer's app change, you can deploy the project in an environment that isn't serverless, which provides the flexibility to manage scaling, run on virtual networks, and even completely isolate the functions.

Azure Logic Apps

Azure Logic Apps allows developers to add workflows to support their Azure functions. Every logic app workflow starts with a trigger, which fires when a specific event happens or when newly available data meets specific criteria. Many triggers include basic scheduling capabilities, so developers can specify how regularly their workloads will run. For custom scheduling scenarios, developers can start their workflows with the **Schedule trigger**. Each time the trigger fires, the Logic Apps engine creates a logic app instance that runs the actions in the workflow. These actions can also include data conversions and flow controls, such as conditional statements, switch statements, loops, and branching.

Developers can visually build their logic apps by using the Logic Apps Designer, which is available in the Azure portal and in Microsoft Visual Studio. For custom logic apps, they can create or edit logic app definitions in JavaScript Object Notation (JSON) by working in the code view editor. They can also use Azure PowerShell commands and Azure Resource Manager templates for selected tasks. Logic apps deploy and run in the cloud in Azure. For a more-detailed introduction, refer to this video: **Use Azure Enterprise Integration Services to run cloud apps at scale**[15].

Serverless computing summary

Serverless computing encompasses three ideas: the abstraction of servers, an event-driven scale, and micro-billing:

- The abstraction of servers. Serverless computing is fully managed. Users never explicitly reserve server instances; the platform handles this. Each function execution can run on a different compute instance, and this is completely transparent to the code.
- An event-driven scale. Serverless computing is an excellent fit for workloads that respond to incoming events. Events include those triggered by timers (for example, if a function needs to run every day at 10:00 AM), HTTP (API and webhook scenarios), queues (for example, with order processing), and much more. Instead of writing an entire application, the developer authors a function, which contains both code and metadata about its triggers and bindings. The platform automatically schedules the function to run and scales the number of compute instances based on the rate of incoming events. Triggers define how a function is invoked. **Input and output bindings**[16] provide a declarative way to connect to services from within the code.
- Micro-billing. Traditional computing has the notion of per-second billing, but often, that's not as useful as it seems. Even if a customer's website gets only one hit a day, they still pay for a full day's worth of availability. With serverless computing, they pay only for the time their code runs. If no active function executions occur, they're not charged. For example, if the code runs once a day for two minutes, they're charged for one execution and two minutes of computing time.

Exercise

Go through the creation of a function app in the Azure portal, as depicted **here**[17].

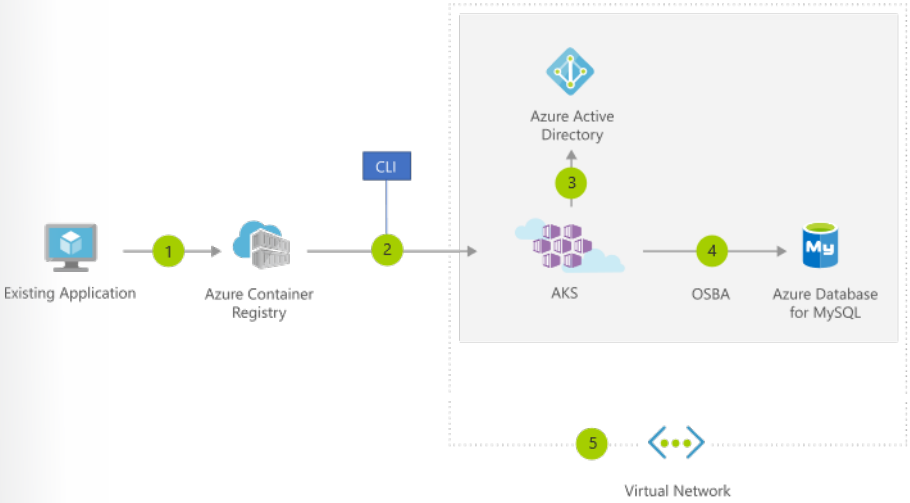
15 https://channel9.msdn.com/Events/Connect/2017/T119/
16 https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings
17 https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-function-app-portal

Containers

A container is a modified runtime environment that prevents a program from accessing protected resources. A container interacts directly with the host OS and augments the containment functions. A container doesn't use virtualization, so it doesn't waste resources simulating virtual hardware with a redundant OS. This typically makes containers more lightweight than VMs. They help you use the container technology built in to the OS. For example, one of the popular services is Azure Kubernetes Service (AKS).

For container management, Azure Container Instances was released in April 2018. Container Instances offers an on-demand compute service delivering the rapid deployment of containers with no VM management and an automatic elastic scale. The Container Instances connector for Kubernetes uses the Kubernetes API to deliver the per-second billing and zero infrastructure execution of Container Instances.

You can move existing applications to containers and run them within AKS. You can control access via integration with Azure AD and access Service Level Agreement (SLA)–backed Azure services, such as Azure Database for MySQL for any data needs, via Open Service Broker for Azure (OSBA).



The preceding figure depicts the process, as follows:

1. You convert an existing application to one or more containers and then publish one or more container images to the Azure Container Registry.
2. By using the Azure portal or the command line, you deploy the containers to an AKS cluster.
3. Azure AD controls access to AKS resources.
4. You access SLA-backed Azure services, such as Azure Database for MySQL, via OSBA.
5. Optionally, AKS is deployed with a virtual network.

Exercise Create an AKS cluster

1. Open the **Azure portal**[18].
2. Select the **Cloud Shell** icon next to the search box.
3. Select **PowerShell**, and then create the Azure file share.

18 https://portal.azure.com/

4. Create a resource group by using Azure PowerShell:

Note: To find a location near you, refer to **https://azure.microsoft.com/regions/services**.

PS Azure: az group create –name AZ500 –location westus

Azure outputs the following:

Azure:/

“id”: "/subscriptions/61f927e9-94e6-4f6d-a737-5d482c6f4316/resourceGroups/AZ500",

“location”: "westus", “managedBy”: null,"name": “AZ500”,"properties": {

“provisioningState”: "Succeeded" },“tags”: null, "type": null}

1. Create the Kubernetes cluster, which is a three-node cluster. The –no-wait returns to your command-line interface (CLI) window while the cluster is being built:

PS Azure: az aks create –resource-group AZ500 -name alamo -node-count 3 –generate-ssh-keys –no-wait

Azure outputs the following:

Azure:/

SSH key files ‘/home/philip/.ssh/id_rsa’ and '/home/philip/.ssh/id_rsa.pub' have been generated under ~/.ssh to allow SSH access to the VM. If using machines without permanent storage like Azure Cloud Shell without an attached file share, back up your keys to a safe location

Finished service principal creation[##################################] 100.0000%

Azure:/

1. Review your resource groups in the Azure portal to find the Kubernetes service you created.

Azure networking

Build a network

Azure has two **deployment models**[19] for creating and working with resources: The Resource Manager deployment model and the Classic Deployment model. For a review of the classic deployment model, go **here.**[20]

We recommend creating most new virtual networks through the Resource Manager deployment model. One of the key reasons is that Azure immediately deletes virtual networks (in the classic deployment model) when a subscription is disabled. Azure deletes virtual networks (in the classic deployment model) regardless of whether they contain resources. If you later reenable the subscription, you have to re-create the resources that the virtual network contained.

Azure Resource Manager

Azure Resource Manager is the deployment and management service for Azure. It provides a consistent management layer that allows you to create, update, and delete resources in your Azure subscription. You can use its access control, auditing, and tagging features to help secure and organize your resources after deployment.

When you take actions through the portal, Azure PowerShell, the Azure CLI, REST APIs, or client software development kits (SDKs), the Resource Manager API handles your request. Because the same API handles all requests, you get consistent results and capabilities from all the different tools. Functionality initially released through APIs should be represented in the portal within 180 days of the initial release.

Azure networking components

The following sections define key terminology for Azure networking. Later, this course will cover each of these areas in more detail.

Virtual networks

Azure Virtual Network is a fundamental component that acts as an organization's network in Azure. Organizations can use virtual networks to connect resources. Virtual networks in Azure are network overlays that you can use to configure and control the connectivity among Azure resources, such as VMs and load balancers.

IP addresses

VMs, Azure load balancers, and application gateways in a single virtual network require unique Internet Protocol (IP) addresses the same way that clients in an on-premises subnet do. This enables these resources to communicate with each other. A virtual network uses two types of IP addresses:

- Private. A private IP address is dynamically or statically allocated to a VM from the defined scope of IP addresses in the virtual network. VMs use these addresses to communicate with other VMs in the same or connected virtual networks through a gateway / Azure ExpressRoute connection. These private IP addresses, or non-routable IP addresses, conform to RFC 1918.
- Public. Public IP addresses, which allow Azure resources to communicate with external clients, are assigned directly at the virtual network adapter of the VM or to the load balancer. Public IP address

19 https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-deployment-model?toc=%2fazure%2fvirtual-network%2ftoc.json

20 https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-deployment-model

can also be added to Azure-only virtual networks. All IP blocks in the virtual network will be routable only within the customer's network, and they won't be reachable from outside. Virtual network packets travel through the high-speed Azure backplane.

You can control the dynamic IP addresses assigned to VMs and cloud services within an Azure virtual network by specifying an IP addressing scheme. Planning an IP addressing scheme within an Azure virtual network is much like planning an IP addressing scheme on-premises. The same ranges are often used, and the same rules applied. However, conditions exist that are unique to Azure virtual networks.

Subnets

You can further divide your network by using subnets for the logical and security-related isolation of Azure resources. Each subnet contains a range of IP addresses that fall within the virtual network address space. Subnetting hides the details of internal network organization from external routers. Subnetting also segments the host within the network, making it easier to apply network security at the interconnections between subnets.

Network adapters

VMs communicate with other VMs and other resources on the network by using virtual network adapters. Virtual network adapters configure VMs with private and, optionally, public IP address. A VM can have more than one network adapter for different network configurations.

DNS

The Domain Name System (DNS) enables clients to resolve user-friendly fully qualified domain names (FQDNs), such as www.adatum.com, to IP addresses. Azure provides a DNS to support many name resolution scenarios. However, in some cases, such as hybrid connection, you might need to configure an external DNS to provide name resolution for the VMs in a virtual network.

Planning for name resolution

The following table describes how to plan for name resolution in Azure virtual networks in different scenarios.

Scenario	Location	Name Resolutions Provision
Between VMs	same cloud service	Use Azure provided name resolution.
Between role instances or VMs	Same VNet but different cloud services	Use your own DNS implementation. For FQDN resolution, you can use Azure name resolution for the first 100 cloud services.
Between VMs ro role instances and on-premises computers	Azure VNets and on-premises	Use your own DNS server/DNS implementation.
Between VMs	Different VNets	Use your own DNS server/DNS implementation.
Between on-premises computers and public endpoints	On-premises to Azure	Use Microsoft Azure external name resolution.

Name resolution is the process by which a computer name is resolved to an IP address. A computer can use the IP address to connect to the named computer by using the IP address, which the user might find difficult to remember.

Azure provides a name resolution service that enables VMs and cloud services within Azure to communicate by name. However, some configurations exceed the reach of the Azure name resolution service. Carefully plan name resolution to ensure that all computers and VMs can connect.

Azure load balancers

To increase availability and scalability, you can create two or more VMs that publish the same application. For example, if three VMs host the same website, you might want to distribute incoming traffic among them to ensure that if one VM fails, traffic will automatically distribute to the other two. You can use an Azure load balancer to enable this traffic distribution among VMs. In this configuration, multiple VMs or services share a single endpoint. For example, you can spread the load of web request traffic across multiple web servers.

You can use two types of Azure load balancers:

- Public. A public load balancer maps the public IP address and port number of incoming traffic to the private IP address and port number of the VM and vice versa for the response traffic from the VM.
- Internal. An internal load balancer directs traffic only to resources that exist inside a virtual network or that use a virtual private network (VPN) to access the Azure infrastructure. In this respect, internal load balancers differ from a public load balancers. The Azure infrastructure restricts access to the load-balanced front-end IP addresses of a virtual network. Front-end IP addresses and virtual networks are never directly exposed to an internet endpoint. Internal line-of-business applications run in Azure and are accessed from within Azure or from on-premises resources.

Load balancer service monitoring

The load balancer can probe the health of the various server instances. When a probe fails to respond, the load balancer stops sending new connections to the unhealthy instances. Existing connections are not impacted.

Azure supports three types of probes:

- Guest agent probe (only on PaaS VMs). The load balancer uses the guest agent inside the VM. It listens and responds with an HTTP 200 OK response only
  when the instance is in the ready state (that is, the instance is not in a state like busy, recycling, or stopping). If the guest agent fails to respond
  with an HTTP 200 OK, the load balancer marks the instance as unresponsive and stops sending traffic to that instance. The load balancer continues to
  ping the instance. If the guest agent responds with an HTTP 200, the load balancer will send traffic to that instance again. When you use a web role,
  your website code typically runs in **w3wp.exe**, which the Azure fabric or guest agent doesn't monitor. This means that failures in **w3wp.exe**
  (for example, HTTP 500 responses) won't be reported to the guest agent, and the load balancer won't know to take that instance out of rotation.
- HTTP custom probe. This probe overrides the default (guest agent) probe. You can use it to create your own custom logic to determine the health
  of the role instance. The load balancer will regularly probe your endpoint (by default, every 15 seconds). The instance will be considered in
  rotation if it responds with a TCP ACK or HTTP 200 within the timeout period (by default, 31 seconds). This is useful for implementing your own
  logic to remove instances from the load balancer's rotation. For example, you can configure the instance to return a status other than 200 if
  its CPU usage is greater than 90 percent. For web roles that use **w3wp.exe**, you also get automatic monitoring of your website, because failures
  in your website code will return a status other than 200 to the probe.

- Transmission Control Protocol (TCP) custom probe. This probe relies on the successful establishment of a TCP session to a defined probe port.

For more information, refer to the **LoadBalancerProbe schema**[21].

Source Network Address Translation (SNAT)

All outbound traffic to the internet that originates from your service undergoes Source Network Address Translation (SNAT) by using the same virtual IP (VIP) as for incoming traffic. SNAT provides important benefits:

- It enables the easy upgrade and disaster recovery of services, because the VIP can be dynamically mapped to another instance of the service.
- It makes access control list (ACL) management easier, because the ACL can be expressed in terms of VIPs and thus don't change as services scale up or down or get redeployed.

The load balancer configuration supports full cone network address translation (NAT) for User Datagram Protocol. Full cone NAT is a type of NAT where the port allows inbound connections from any external host (in response to an outbound request). For each new outbound connection that a VM initiates, the load balancer also allocates an outbound port. The external host sees traffic with a VIP-allocated port. If your scenarios require many outbound connections, we recommend that the VMs use **Instance-level Public**[22] IP addresses so that they have a dedicated outbound IP address for SNAT. This reduces the risk of port exhaustion.

Application gateways

Application gateways provide load-balanced solutions for network traffic that is based on HTTP. They use routing rules as application-level policies that can offload Secure Sockets Layer (SSL) processing from load-balanced VMs. In addition, you can use application gateways for a cookie-based session affinity scenario.

Azure Traffic Manager

Azure Traffic Manager is another load-balancing solution that Azure includes. You can use Traffic Manager to balance loads among endpoints that are located in different Azure regions, at hosted providers, or in on-premises datacenters. These endpoints can include Azure VMs and Azure websites. You can configure this load balancing service to support priorities or to help ensure that users connect to an endpoint that is close to their physical location for faster response times.

Traffic Manager features

Traffic Manager can:

- Improve the availability of critical applications. Traffic Manager allows you to deliver high availability for your critical applications by monitoring your endpoints in Azure and providing automatic failover when an endpoint stops working.
- Improve responsiveness for high-performance applications. Azure allows you to run cloud services or websites in datacenters located around the world. Traffic Manager can improve the responsiveness of your applications by directing users to the endpoint with the lowest network latency from the client.
- Upgrade and perform service maintenance on applications without downtime. You can seamlessly carry out upgrade and other planned maintenance operations on your applications—without downtime for users—by using Traffic Manager to direct traffic to alternative endpoints while maintenance is in progress.

21 https://msdn.microsoft.com/library/azure/jj151530.aspx
22 https://azure.microsoft.com/en-us/documentation/articles/virtual-networks-instance-level-public-ip/

- Combine on-premises and cloud-based applications. Traffic Manager supports external endpoints that aren't in Azure, allowing you to use it with hybrid (cloud and on-premises) deployments, including the burst-to-cloud, migrate-to-cloud, and failover-to-cloud scenarios.
- Distribute traffic for large, complex deployments. You can combine traffic-routing methods by using nested Traffic Manager profiles to create sophisticated and flexible traffic-routing configurations that meet the needs of larger, more-complex deployments.

The most important point to understand is that Traffic Manager works at the DNS level. Traffic Manager uses DNS to direct users to particular service endpoints—based on the chosen traffic-routing method and the current endpoint health. Clients then directly connect to the selected endpoint. Traffic Manager isn't a proxy and doesn't see the traffic passing between the client and the service.

Network security groups

You can use network security groups to provide network isolation for Azure resources by defining rules that allow or deny specific traffic to individual VMs or subnets. You can this design your Azure virtual network to provide a network experience that is similar to an that of an on-premises network. You can achieve the same functionality in your Azure virtual network as you can in on-premises networks, such as perimeter networks.

User Defined Routes

User Defined Routes (UDRs) control network traffic by defining routes that specify the next hop of the traffic flow. You can assign UDRs to virtual network subnets.

Forced tunneling

With forced tunneling, you can redirect internet-bound traffic back to the company's on-premises infrastructure. Forced tunneling is commonly used in a scenario where organizations want to implement packet inspection or do a corporate audit.

Regional virtual networks

Azure Virtual Network is bound to Azure subscriptions, and multiple subscriptions can't use the same Azure virtual network. If different Azure subscriptions need communicate, you need to create separate Azure virtual networks in each subscription and then use site-to-site virtual network connections or the ExpressRoute service to connect them. All new virtual networks are regional virtual networks. This means that they can span a complete Azure region or datacenter. This differs from the earlier implementation of virtual networks in Azure, which were restricted to a single affinity group, allowing you to co-locate virtual networks, storage accounts, and services in physical proximity to each other within the same area of a single datacenter. If you have older virtual networks in your subscription, they might be tied to an affinity group. However, over time, you'll need to migrate all the virtual networks to regional virtual networks and remove their ties to specific affinity groups.

Cross-premises network connectivity

With virtual networks in Azure, you can also extend your on-premises network to the cloud. To extend your on-premises network, you can create a VPN between your on-premises computers or networks and an Azure virtual network. Alternatively, you can use ExpressRoute to provide a connection to an Azure virtual network that doesn't cross the internet. With these two methods, you can enable on-premises users to access Azure services as if they were physically located on-premises in your own datacenter.

To connect to an Azure virtual network from an on-premises network, you can use:

- A point-to-site VPN.
- A site-to-site VPN.
- ExpressRoute.

Exercise Create a virtual network

A virtual network can be created by using the Azure portal, the **Azure CLI**[23] 1.0, or **Azure PowerShell**[24].

Exercise

Using the Azure portal

1. Sign in to the **Azure portal**[25].
2. In the upper-left corner of the screen, select **Create a resource > Networking > Virtual network**.
3. In **Create virtual network**, enter or select the information in the following table.

Setting	Action
Name	Enter **myVirtualNetwork**
Address space	Enter **10.1.0.0/16**.
Subscription	Select your subscription
Resource group	Select **Create new**, enter **myResourceGroup**, and then select **OK**.
Location	Select a location near you.
Subnet - Name	Enter **myVirtualSubnet**.
Subnet - Address range	Enter **10.1.0/24**.

4. Leave the rest of the defaults, and then select Create. Notice these settings. Point to the information icon to review the description of each setting:
   - DDoS protection
   - Service Endpoints
   - Firewall
5. Select the notifications icon, which displays a highlighted number, and then review your new virtual network by selecting Go to resource.

To add two VMs to your virtual network, follow the steps detailed **here**[26].

Azure load balancer

Azure Load Balancer is available in two stock keeping units (SKUs): Basic and Standard. They differ in scale, features, and pricing. Any scenario that's possible with the Basic SKU can also be created with Standard SKU, although the approaches might slightly differ. As you learn about Load Balancer, it's important to familiarize yourself with the fundamentals and the SKU-specific differences. You can fine the SKU comparisons **here**[27].

You can use Azure Load Balancer to:

- Load balance incoming internet traffic to your VMs. This configuration is known as a **public load balancer**[28].

23 https://docs.microsoft.com/en-us/azure/virtual-network/create-virtual-network-classic
24 https://docs.microsoft.com/en-us/azure/virtual-network/create-virtual-network-classic
25 https://portal.azure.com/
26 https://docs.microsoft.com/en-us/azure/virtual-network/quick-create-portal
27 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview
28 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview

- Load balance traffic across VMs inside a virtual network. You can also reach a Load Balancer front end from an on-premises network in a hybrid scenario. Both scenarios use a configuration known as an **internal load balancer**[29].
- Forward traffic to a specific port on specific VMs with inbound NAT rules.
- Provide **outbound connectivity**[30] for VMs inside your virtual network by using a public load balancer.

Azure Load Balancer supports several distribution modes. For more information about the modes, go **here**[31].

You can create a basic public load balancer by using the **Azure portal**[32], the **Azure CLI**[33], or **Azure PowerShell**[34].

Create a network load balancer

Exercise

Using Azure PowerShell

1. On the menu on the upper-right corner of the Azure portal, select the **Cloud Shell** icon .
2. Create a resource group by using **New-AzureRmResourceGroup**[35]. The following example creates a resource group named **myResourceGroupLB** in the **EastUS** location:

New-AzureRmResourceGroup -ResourceGroupName "myResourceGroupLB" -Location "EastUS"

1. Note that to access your app on the internet, you need a public IP address for the load balancer. Create a public IP address by using **New-AzureRmPublicIpAddress**[36]. The following example creates a public IP address named **myPublicIP** in the **myResourceGroupLB** resource group:

$publicIP = New-AzureRmPublicIpAddress -ResourceGroupName "myResourceGroupLB" -Location "EastUS" -AllocationMethod "Static" -Name "myPublicIP"

1. Create a front-end (internet-facing) IP configuration by using **New-AzureRmLoadBalancerFrontendIpConfig**[37]. The following example creates a front-end IP configuration named **myFrontEnd** and attaches the **myPublicIP** IP address:

$frontendIP = New-AzureRmLoadBalancerFrontendIpConfig -Name "myFrontEnd" -PublicIpAddress $publicIP

1. Create a back-end address pool by using **New-AzureRmLoadBalancerBackendAddressPoolConfig**[38]. The VMs attach to this back-end pool in the remaining steps. The following example creates a back-end address pool named **myBackEndPool**:

$backendPool = New-AzureRmLoadBalancerBackendAddressPoolConfig -Name "myBackEndPool"

1. **Note** that it doesn't do any good to load balance a request to a service that isn't responding. The load balancer needs to monitor the status of the services. To allow the load balancer to monitor the status of your app, you use a health probe. The health probe dynamically adds or removes VMs from the

29 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview
30 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-outbound-connections
31 https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-distribution-mode
32 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-create-basic-load-balancer-portal
33 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-create-basic-load-balancer-cli
34 https://docs.microsoft.com/en-us/azure/load-balancer/quickstart-create-basic-load-balancer-powershell
35 https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/new-azurermresourcegroup
36 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermpublicipaddress
37 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermloadbalancerfrontendipconfig
38 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermloadbalancerbackendaddresspoolconfig

load balancer rotation based on their responses to health checks. By default, a VM is removed from the load balancer distribution after two consecutive failures at 15-second intervals. You create a health probe based on either a protocol or a specific health check page for your app.

The following example creates a TCP probe. You can also create custom HTTP probes for health checks that are more finely grained. When using a custom HTTP probe, you need to create a health check page, such as **healthcheck.aspx**. The probe needs to return an HTTP 200 OK response for the load balancer to keep the host in rotation.

To create a TCP health probe, use **Add-AzureRmLoadBalancerProbeConfig**[39].

The following example creates a health probe named **myHealthProbe** that monitors each VM on HTTP port 80:

$probe = New-AzureRmLoadBalancerProbeConfig -Name "myHealthProbe" -RequestPath health-check2.aspx -Protocol http -Port 80 -IntervalInSeconds 16 -ProbeCount 2

1. **Note** that a load balancer rule is used to define how traffic is distributed to the VMs. You define the front-end IP configuration for the incoming traffic and the back-end IP pool to receive the traffic along with the required source and destination ports. To make sure that only healthy VMs receive traffic, you also define the health probe to use.

Create a load balancer rule by using **Add-AzureRmLoadBalancerRuleConfig**[40]. The following example creates a load balancer rule named **myLoadBalancerRule** and balances traffic on TCP port 80:

$lbrule = New-AzureRmLoadBalancerRuleConfig -Name "myLoadBalancerRule" -FrontendIpConfiguration $frontendIP -BackendAddressPool $backendPool -Protocol Tcp -FrontendPort 80 -BackendPort 80 -Probe $probe

1. Create NAT rules by using **New-AzureRmLoadBalancerInboundNatRuleConfig**[41]. The following example creates NAT rules named **myLoadBalancerRDP1** and **myLoadBalancerRDP2** to allow Remote Desktop Protocol (RDP) connections to the back-end servers with ports 4221 and 4222:

$natrule1 = New-AzureRmLoadBalancerInboundNatRuleConfig `

-Name 'myLoadBalancerRDP1' `

-FrontendIpConfiguration $frontendIP `

-Protocol tcp `

-FrontendPort 4221 `

-BackendPort 3389

$natrule2 = New-AzureRmLoadBalancerInboundNatRuleConfig `

-Name 'myLoadBalancerRDP2' `

-FrontendIpConfiguration $frontendIP `

-Protocol tcp `

-FrontendPort 4222 `

39 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/add-azurermloadbalancerprobeconfig
40 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/add-azurermloadbalancerruleconfig
41 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermloadbalancerinboundnatruleconfig

-BackendPort 3389

1. Create a basic load balancer by using **New-AzureRmLoadBalancer**[42]. The following example creates a public basic load balancer named **myLoadBalancer** by using the front-end IP configuration, back-end pool, health probe, load balancing rule, and NAT rules you created in the preceding steps:

$lb = New-AzureRmLoadBalancer `

-ResourceGroupName 'myResourceGroupLB' `

-Name 'MyLoadBalancer' `

-Location 'eastus' `

-FrontendIpConfiguration $frontendIP `

-BackendAddressPool $backendPool `

-Probe $probe `

-LoadBalancingRule $lbrule `

-InboundNatRule $natrule1,$natrule2

Before you test the newly created load balancer, you need an introduction to some additional services to distribute traffic.

Configure Azure Traffic Manager

Azure Traffic Manager is a DNS-based traffic load balancer that enables you to optimally distribute traffic to services across global Azure regions while providing high availability and responsiveness.

Traffic Manager uses DNS to direct client requests to the most appropriate service endpoint based on a **traffic-routing method**[43] and the health of the endpoints. An endpoint is any internet-facing service hosted inside or outside of Azure. Traffic Manager provides a range of traffic-routing methods and **endpoint monitoring options**[44] to suit different application needs and automatic failover models. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

When a client attempts to connect to a service, it must first resolve the DNS name of the service to an IP address. The client then connects to that IP address to access the service.

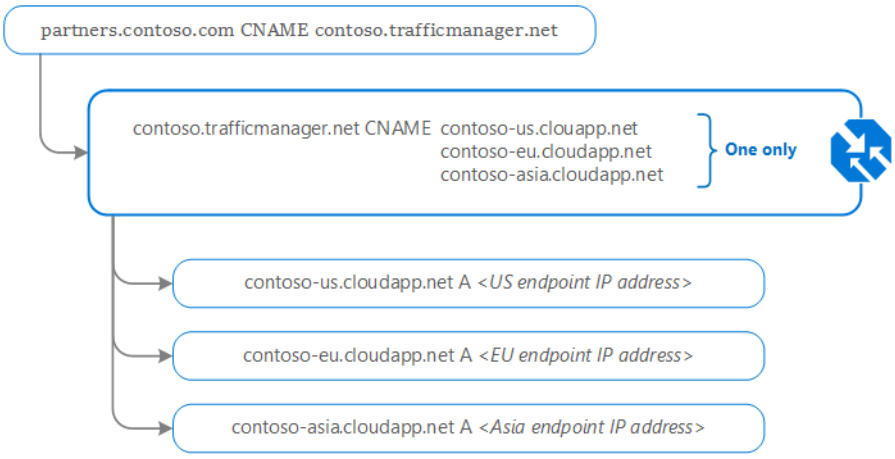
Traffic Manager example

Contoso Corporation has developed a new partner portal. The URL for this portal is https://partners.contoso.com/login.aspx. The application is hosted in three regions of Azure. To improve availability and maximize global performance, they use Traffic Manager to distribute client traffic to the closest available endpoint.

To achieve this configuration, they complete the following steps:

1. Deploy three instances of their service. The DNS names of these deployments are contoso-us.cloudapp.net, contoso-eu.cloudapp.net, and contoso-asia.cloudapp.net.
2. Create a Traffic Manager profile, named contoso.trafficmanager.net, and configure it to use the performance traffic-routing method across the three endpoints.
3. Configure their vanity URL, partners.contoso.com, to point to contoso.trafficmanager.net by using a DNS canonical name record (CNAME record).

42 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermloadbalancer
43 https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods
44 https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-monitoring



How clients connect by using Traffic Manager

When a client requests the page https://partners.contoso.com/login.aspx, the client performs the following steps to resolve the DNS name and establish a connection:

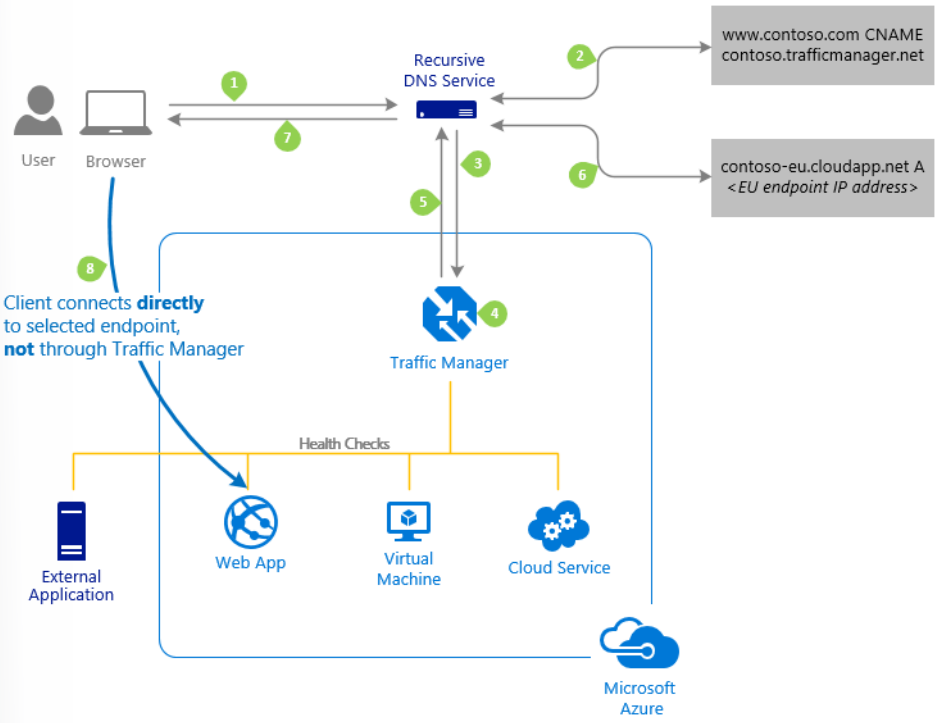
1. The client sends a DNS query to its configured recursive DNS service to resolve the name partners.contoso.com. A recursive DNS service, sometimes called a local DNS service, doesn't directly host DNS domains. Rather, the client offloads the work of contacting the various authoritative DNS services across the internet needed to resolve a DNS name.
2. To resolve the DNS name, the recursive DNS service finds the name servers for the contoso.com domain. It then contacts those name servers to request the partners.contoso.com DNS record. The contoso.com DNS servers return the CNAME record that points to contoso.trafficmanager.net.
3. The recursive DNS service finds the name servers for the trafficmanager.net domain, which Azure Traffic Manager provides. It then sends a request for the contoso.trafficmanager.net DNS record to those DNS servers.
4. The Traffic Manager name servers receive the request. They choose an endpoint based on:
5. - The configured state of each endpoint (disabled endpoints are not returned).

   - The current health of each endpoint as determined by the Traffic Manager health checks.

   - The chosen traffic-routing method.

6. The chosen endpoint is returned as another DNS CNAME record. In this case, suppose that contoso-us.cloudapp.net is returned.
7. The recursive DNS service finds the name servers for the cloudapp.net domain. It contacts those name servers to request the contoso-us.cloudapp.net DNS record. A DNS address record (A record) containing the IP address of the US-based service endpoint is returned.
8. The recursive DNS service consolidates the results and returns a single DNS response to the client.
9. The client receives the DNS results and connects to the specified IP address. The client connects to the application service endpoint directly and not through Traffic Manager. Because it's an HTTPS endpoint, the client performs the necessary SSL / Transport Layer Security (TLS) handshake and then makes an HTTP GET request for the /login.aspx page.

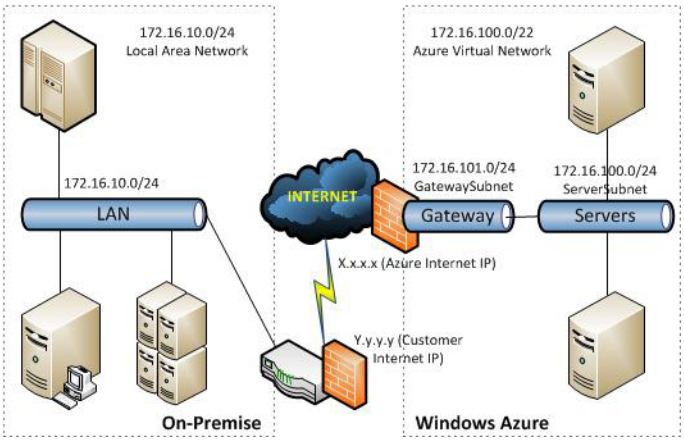
The following figure depicts these steps.



The recursive DNS service caches the DNS responses it receives. The DNS resolver on the client device also caches the result. Caching enables subsequent DNS queries to be answered more quickly by using data from the cache rather than by querying other name servers. The Time to Live (TTL) property of each DNS record determines the duration of the cache. Smaller values result in faster cache expiration and thus more round trips to the Traffic Manager name servers. Larger values mean that it might take longer to direct traffic away from a failed endpoint. Traffic Manager allows you to configure the TTL used in Traffic Manager DNS responses to be as low as zero seconds and as high as 2,147,483,647 seconds (the maximum range compliant with RFC 1035), enabling you to choose the value that best balances the needs of your application.

Exercise Configure Azure Virtual Network gateways

An Azure Virtual Network gateway serves as the cross-premises gateway connecting your workloads in Azure virtual networks to on-premises sites. It's required to connect to on-premises sites through Internet Protocol security (IPsec) site-to-site VPN (S2S VPN) tunnels or through ExpressRoute circuits. For IPsec / Internet Key Exchange (IKE) VPN tunnels, the gateways perform IKE handshakes and establish the IPsec S2S VPN tunnels between the virtual networks and on-premises sites. For ExpressRoute, the gateways advertise the prefixes in your virtual networks via the peering circuits, and they forward packets from your ExpressRoute circuits to your VMs inside your virtual networks.



Create a high-performance gateway

To create a gateway for a virtual network named **MyAzureVNET**, use the following Azure PowerShell cmdlet:

PS D:> New-AzureVNETGateway –Newname MyAzureVNET –GatewayType DynamicRouting –GatewaySKU HighPerformance

Note that DynamicRouting is the GatewayType for both the DynamicRouting gateway and the dedicated (ExpressRoute) gateway. Therefore, you can also use the cmdlet example to create a virtual network gateway to connect to an ExpressRoute circuit.

Test the Azure load balancer

1. Create a virtual network by using **New-AzureRmVirtualNetwork**[45]. The following example creates a virtual network named myVNET with mySubnet:

Create the subnet configuration.

```
$subnetConfig = New-AzureRmVirtualNetworkSubnetConfig `
-Name "mySubnet" `
-AddressPrefix 10.0.2.0/24
```

Create the virtual network.

```
$vNET = New-AzureRmVirtualNetwork `
-ResourceGroupName "myResourceGroupLB" `
-Location "EastUS" `
-Name "myVNET" `
-AddressPrefix 10.0.0.0/16 `
-Subnet $subnetConfig
```

1. Create a network security group to define inbound connections to your virtual network by creating a network security group rule for port 3389.

Create a network security group rule to allow RDP connections through port 3389 by using **New-AzureRmNetworkSecurityRuleConfig**[46].

45 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermvirtualnetwork
46 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermnetworksecurityruleconfig

```
$rule1 = New-AzureRmNetworkSecurityRuleConfig -Name 'myNetworkSecurityGroupRuleRDP' -Description 'Allow RDP' `
-Access Allow -Protocol Tcp -Direction Inbound -Priority 1000 `
-SourceAddressPrefix Internet -SourcePortRange * -DestinationAddressPrefix * `
-DestinationPortRange 3389
```

1. Create a network security group rule to allow inbound connections through port 80 by using **New-AzureRmNetworkSecurityRuleConfig**[47].

```
$rule2 = New-AzureRmNetworkSecurityRuleConfig `
-Name 'myNetworkSecurityGroupRuleHTTP' `
-Description 'Allow HTTP' -Access Allow -Protocol Tcp `
-Direction Inbound -Priority 2000 -SourceAddressPrefix Internet `
-SourcePortRange * -DestinationAddressPrefix * -DestinationPortRange 80
```

1. Create a network security group by using **New-AzureRmNetworkSecurityGroup**[48].

```
$nsg = New-AzureRmNetworkSecurityGroup -ResourceGroupName 'myResourceGroupLB' -Location 'EastUS' -Name 'myNetworkSecurityGroup' -SecurityRules $rule1,$rule2
```

1. Create virtual network adapters by using **New-AzureRmNetworkInterface**[49]. The following example creates two virtual network adapters (one for each VM you create for your app in the following steps). You can create additional virtual network adapters and VMs at any time and add them to the load balancer.

Create a network adapter for the first VM.

```
$nicVM1 = New-AzureRmNetworkInterface `
-ResourceGroupName 'myResourceGroupLB' `
-Location 'EastUS' `
-Name 'MyNic1' `
-LoadBalancerBackendAddressPool $backendPool `
-NetworkSecurityGroup $nsg `
-LoadBalancerInboundNatRule $natrule1 `
-Subnet $vNET.Subnets[0]
```

Create a network adapter for the second VM.

```
$nicVM2 = New-AzureRmNetworkInterface `
-ResourceGroupName 'myResourceGroupLB' `
-Location 'EastUS' `
-Name 'MyNic2' `
-LoadBalancerBackendAddressPool $backendPool `
-NetworkSecurityGroup $nsg `
```

47 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermnetworksecurityruleconfig
48 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermnetworksecuritygroup
49 https://docs.microsoft.com/en-us/powershell/module/azurerm.network/new-azurermnetworkinterface

```
-LoadBalancerInboundNatRule $natrule2 `
-Subnet $vNET.Subnets[0]
```

1. Create VMs for load balancing. To improve the high availability of your app, place your VMs in an availability set.

Create an availability set by using **New-AzureRmAvailabilitySet**[50]. The following example creates an availability set named **myAvailabilitySet**:

```
$availabilitySet = New-AzureRmAvailabilitySet `
-ResourceGroupName "myResourceGroupLB" `
-Name "myAvailabilitySet" `
-Location "EastUS" `
-Sku aligned `
-PlatformFaultDomainCount 2 `
-PlatformUpdateDomainCount 2
```

1. Set an administrator username and password for the VMs by using **Get-Credential**[51].

```
$cred = Get-Credential
```

1. Create the VMs by using **New-AzureRmVM**[52].
   The following example creates two VMs and the required virtual network components if they do not already exist. During the VM creation,
   the previously created network adapters are associated with the VMs, because they're assigned the same virtual network (myVNET) and subnet (mySubnet).

```
for ($i=1; $i -le 2; $i++)
{
New-AzureRmVm `
-ResourceGroupName "myResourceGroupLB" `
-Name "myVM$i" `
-Location "East US" `
-VirtualNetworkName "myVNET" `
-SubnetName "mySubnet" `
-SecurityGroupName "myNetworkSecurityGroup" `
-OpenPorts 80 `
-AvailabilitySetName "myAvailabilitySet" `
-Credential $cred `
-AsJob
}
```

50 https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermavailabilityset
51 https://msdn.microsoft.com/powershell/reference/5.1/microsoft.powershell.security/Get-Credential
52 https://docs.microsoft.com/en-us/powershell/module/azurerm.compute/new-azurermvm

The *-AsJob* parameter creates the VM as a background task, so the Azure PowerShell prompts return to you. You can get the details of background jobs by using the Job cmdlet. It takes a few minutes to create and configure the two VMs.

Install Microsoft Internet Information Services (IIS) with a custom webpage on the new VMs as follows:

1. Get the public IP address of the load balancer by using Get-AzureRmPublicIPAddress.

```
Get-AzureRmPublicIPAddress -ResourceGroupName "myResourceGroupLB" `
-Name "myPublicIP" | select IpAddress
```

1. Create a remote desktop connection to VM1 by using the public IP address that you got in the previous step. (Note that in the following command, mstsc refers to Microsoft Terminal Services Client.)

```
mstsc /v:PublicIpAddress:4221
```

1. Enter the credentials for VM1 to start the RDP session.
2. Launch Windows PowerShell on VM1, and then use the following commands to install IIS and update the default .htm file.

Install IIS.

```
Install-WindowsFeature -name Web-Server -IncludeManagementTools
```

Remove the default .htm file.

```
remove-item C:\inetpub\wwwroot\iisstart.htm
```

Add a custom .htm file.

```
Add-Content -Path "C:\inetpub\wwwroot\iisstart.htm" -Value $("Hello from" + $env:computername)
```

1. Close the RDP connection with myVM1.
2. Create an RDP connection with myVM2 by running the mstsc /v:PublicIpAddress:4222 command, and then repeat step 4 for VM2.

Finally, test the load balancer

Get the public IP address of your load balancer by using Get-AzureRmPublicIPAddress. The following example gets the IP address for **myPublicIP**, created earlier:

```
Get-AzureRmPublicIPAddress -ResourceGroupName "myResourceGroupLB" `
-Name "myPublicIP" | select IpAddress
```

You can now enter the public IP address into a web browser. The website is displayed, including the hostname of the VM that the load balancer distributed traffic to. To see the load balancer distribute traffic across both of the VMs running your app, you can force a refresh of your web browser.

Revert the Exercise

You can use the **Remove-AzureRmResourceGroup**[53] command to remove the resource groups, VMs, and all related resources when you no longer need them.

```
Remove-AzureRmResourceGroup -Name myResourceGroupLB
```

You can find additional Azure load balancer tutorials **here**[54].

53 https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/remove-azurermresourcegroup
54 https://docs.microsoft.com/en-us/azure/load-balancer/tutorial-load-balancer-basic-internal-portal

Secure the network

Using Network Security Groups

Network Security Groups (NSGs)

VMs that you create via the Resource Manager deployment model can have direct connectivity to the internet by using a public IP address that is directly assigned to the VMs. Only with the host firewall configured inside the VMs helps protect these VMs from the internet.

VMs that you create by using the classic deployment model communicate with internet resources through the cloud service that is assigned the public IP address, which is also known as the VIP. VMs that reside inside the cloud service share that VIP and establish communication with internet resources by using endpoints. If you remove the VM endpoints that map the public port and public IP address of the cloud service to the private port and private IP address of the VM, the VM becomes unreachable from the internet via the public IP address.

Network Security Groups (NSGs) help provide advanced security for the VMs you create via either deployment model (Resource Manager or classic). NSGs control inbound and outbound traffic passing through a network adapter (in the Resource Manager deployment model), a VM (in the classic deployment model), or a subnet (in both deployment models).

Network Security Group rules

NSGs contain rules that specify whether traffic will be approved or denied. Each rule is based on a source IP address, a source port, a destination IP address, and a destination port. Based on whether the traffic matches this combination, the rule either allows or denies the traffic. Each rule consists of the following properties:

- **Name**. This is a unique identifier for the rule.
- **Direction**. This specifies whether the traffic is inbound or outbound.
- **Priority**. If multiple rules match the traffic, rules with a higher priority apply.
- **Access**. This specifies whether the traffic is allowed or denied.
- **Source IP address prefix**. This prefix identifies where the traffic originated from. It can be based on a single IP address; a range of IP addresses in Classless Interdomain Routing (CIDR) notation; or the asterisk (*), which is a wildcard that matches all possible IP addresses.
- **Source port range**. This specifies source ports by using either a single port number from 1 through 65,535; a range of ports (for example, 200–400); or the asterisk (*) to denote all possible ports.
- **Destination IP address prefix**. This identifies the traffic destination based on a single IP address, a range of IP addresses in CIDR notation, or the asterisk (*) to match all possible IP addresses.
- **Destination port range**. This specifies destination ports by using either a single port number from 1 through 65,535; a range of ports (for example, 200–400); or the asterisk (*) to denote all possible ports .
- **Protocol**. This specifies a protocol that matches the rule. It can be UDP, TCP, or the asterisk (*).

Custom Network Security Group rules

Predefined default rules exist for inbound and outbound traffic. You can't delete these rules, but you can override them, because they have the lowest priority. The default rules allow all inbound and outbound traffic within a virtual network, allow outbound traffic towards the internet, and allow inbound traffic to

an Azure load balancer. A default rule with the lowest priority also exists in both the inbound and outbound sets of rules that denies all network communication.

When you create a custom rule, you can use default tags in the source and destination IP address prefixes to specify predefined categories of IP addresses. These default tags are:

- **Internet**. This tag represents internet IP addresses.
- **Virtual_network**. This tag identifies all IP addresses that the IP range for the virtual network defines. It also includes IP address ranges from on-premises networks when they are defined as local network to virtual network.
- **Azure_loadbalancer**. This tag specifies the default Azure load balancer destination.

Planning Network Security Groups

You can design NSGs to isolate virtual networks in security zones, like the model used by on-premises infrastructure does. You can apply NSGs to subnets, which allows you to create protected screened subnets, or DMZs, that can restrict traffic flow to all the machines residing within that subnet. With the classic deployment model, you can also assign NSGs to individual computers to control traffic that is both destined for and leaving the VM. With the Resource Manager deployment model, you can assign NSGs to a network adapter so that NSG rules control only the traffic that flows through that network adapter. If the VM has multiple network adapters, NSG rules won't automatically be applied to traffic that is designated for other network adapters.

You create NSGs as resources in a resource group, but you can share them with other resource groups in your subscription. If you create an NSG in the TestRG resource group, for example, you can use that NSG for a VM belonging to another resource group, such as ProductionRG.

When implementing NSGs, keep these important limits in mind:

- By default, you can create 100 NSGs per region per subscription. You can raise this limit to 400 by contacting Azure support.
- You can apply only one NSG to a VM, subnet, or network adapter.
- By default, you can have up to 200 rules in a single NSG. You can raise this limit to 500 by contacting Azure support.
- You can apply an NSG to multiple resources.

In the earlier exercise to test a load balancer, you created three NSGs. This is a good time to review that exercise.

For additional practice, **Microsoft Online Labs**[55] has a self-paced "Azure Networking Concepts" lab. Refer to "Scenario 2 - Building Network Security Groups." The lab covers the basics of creating virtual networks in the Azure portal, and you should be comfortable doing that before tackling more-advanced networking tasks in Azure.

Try the optional advanced configuration, where you create additional subnets to demonstrate building multiple subnets for network segmentation in a virtual network.

Note: Microsoft Online Labs periodically changes the available practical exercises. If you don't see this lab, please try another lab.

Using network virtual appliances

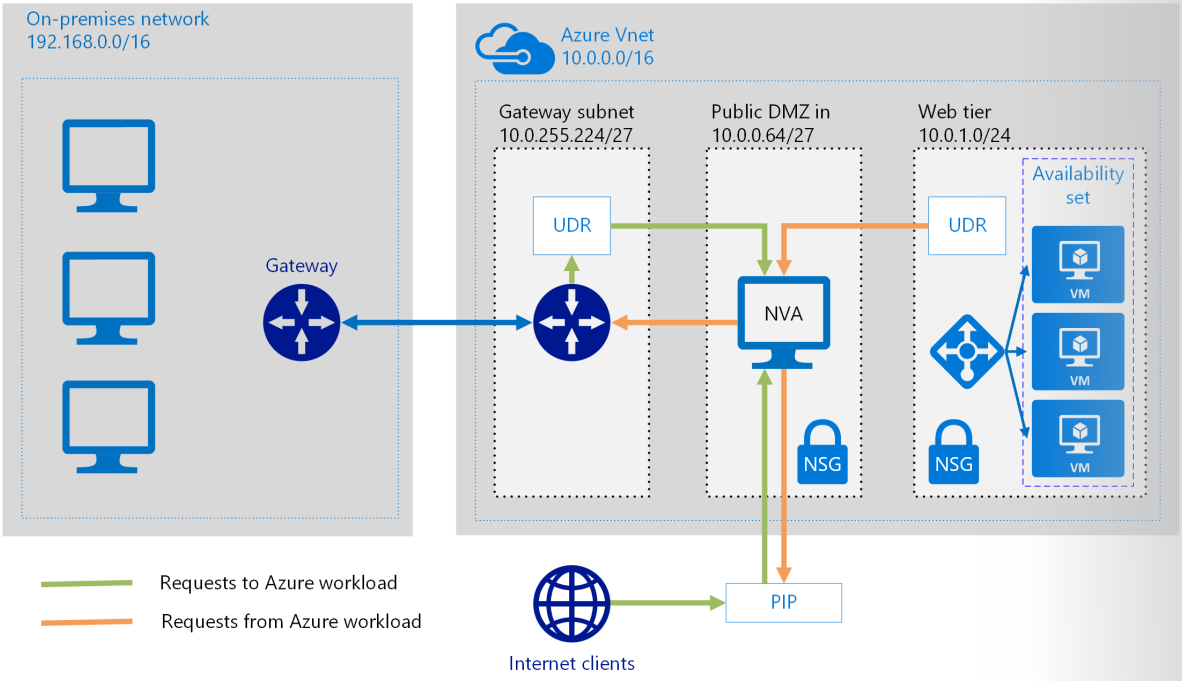
##Configure network virtual appliances

55 https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles

Before discussing network virtual appliances (NVAs), this section will present a basic summary of User Defined Routes (UDRs).

A UDR is a custom route in Azure that overrides Azure's default system routes or adds routes to a subnet's route table. In Azure, you create a route table and then associate that route table with zero or more virtual network subnets. Each subnet can have zero or one route table associated with it. If you create a route table and associate it to a subnet, Azure either combines its routes with the default routes that Azure adds to a subnet or overrides those default routes.

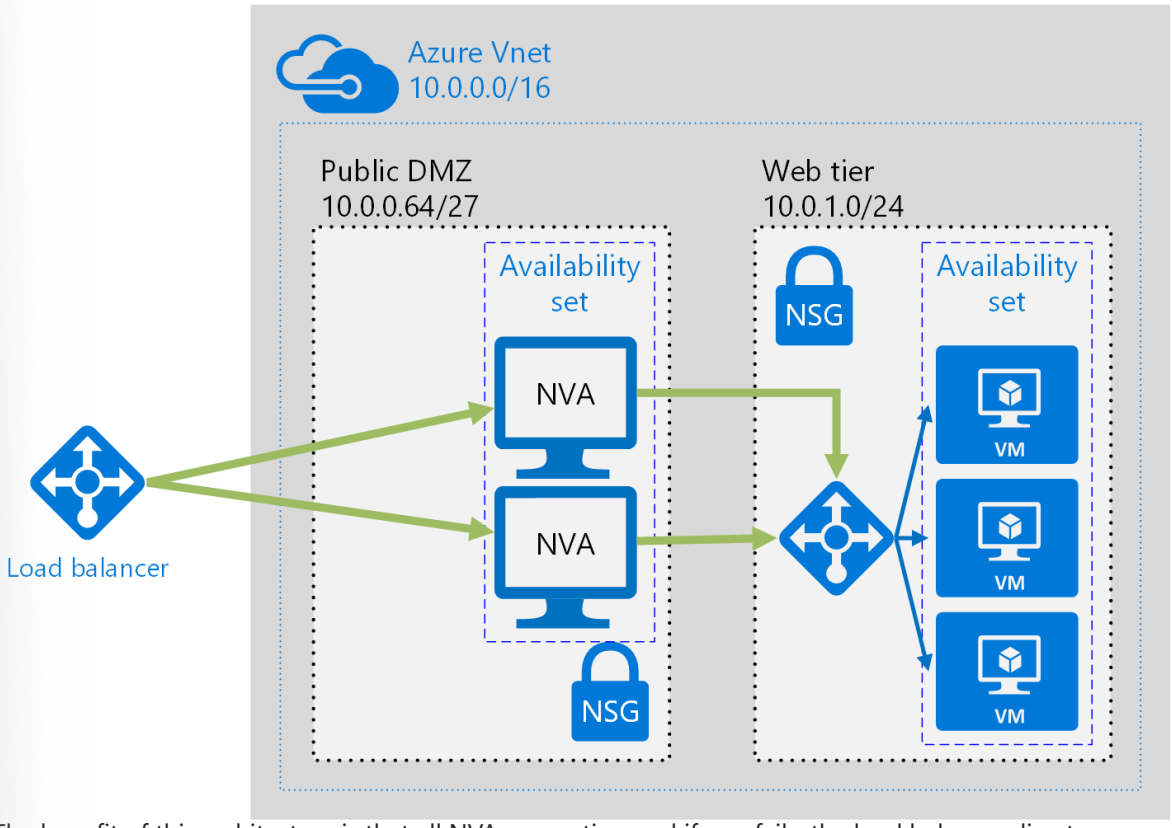
UDRs and NSGs help provide layer 3 and layer 4 (of the OSI model) security. NVAs help provide layer 7, application layer, security. You can deploy an NVA to a perimeter network in many architectures. For example, the following figure depicts the use of a single NVA for ingress.



In the preceding diagram, the NVA helps provide a secure network boundary by checking all inbound and outbound network traffic and then passing only the traffic that meets the network security rules. However, the fact that all network traffic passes through the NVA means that the NVA is a single point of failure in the network. If the NVA fails, no other path will exist for network traffic, and all the back-end subnets will become unavailable.

To make an NVA highly available, deploy more than one NVA into an availability set.

The following figure shows a high-availability architecture that implements an ingress perimeter network behind an internet-facing load balancer. This architecture is designed to provide connectivity to Azure workloads for layer 7 traffic, such as HTTP or HTTPS traffic.



The benefit of this architecture is that all NVAs are active, and if one fails, the load balancer directs network traffic to the other NVA. Both NVAs route traffic to the internal load balancer, so if one NVA is active, traffic will continue to flow. The NVAs are required to terminate SSL traffic intended for the web tier VMs. These NVAs can't be extended to handle on-premises traffic, because on-premises traffic requires another dedicated set of NVAs with their own network routes.

If you need this additional security look, refer to the **Azure Marketplace**[56] for supported appliances.

For additional configurations and detailed solutions, including sample code, refer to the files in the **GitHub repository**[57].

Configure forced tunneling for VPNs

Why do some cases require forced tunneling?

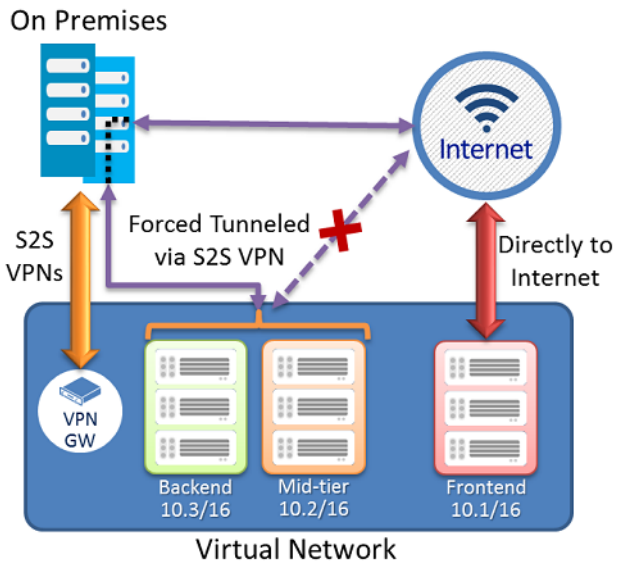
Forced tunneling lets you redirect, or force, all internet-bound traffic back to your on-premises location via a site-to-site VPN tunnel for inspection and auditing. This is a critical security requirement for most enterprise IT policies. Without forced tunneling, internet-bound traffic from your VMs in Azure always traverses from the Azure network infrastructure directly to the internet—without the option to allow you to inspect or audit the traffic. Unauthorized internet access potentially leads to information disclosure or other types of security breaches.

As stated earlier, Azure currently works with two deployment models: The Resource Manager deployment model and the classic deployment model. The two models aren't completely compatible with each other. The following exercise goes through configuring tunneling for virtual networks that were created via the

56 https://docs.microsoft.com/en-us/azure/security-center/security-center-incident?page=1&subcategories=appliances
57 https://aka.ms/ha-nva-fo

Resource Manager deployment model. If you want to configure forced tunneling in the classic deployment model, go **here**[58].

The following figure depicts how forced tunneling works.



In the preceding figure, the front-end subnet doesn't use forced tunneling. The workloads in the front-end subnet can continue to accept and respond to customer requests that come directly from the internet. The mid-tier and back-end subnets use forced tunneling. Any outbound connections from these two subnets to the internet are forced back to an on-premises site via one of the S2S VPN tunnels.

This allows you to restrict and inspect internet access from your VMs or cloud services in Azure while continuing to enable your multi-tier service architecture. If no internet-facing workloads exist in your VMs, you can also apply forced tunneling to the entire virtual network.

You configure forced tunneling in Azure via virtual network UDRs. Redirecting traffic to an on-premises site is expressed as a default route to the Azure VPN gateway. This example uses UDRs to create a routing table to first add a default route and then associate the routing table with your virtual network subnets to enable forced tunneling on those subnets. The following procedure helps you create a resource group and a virtual network. You'll then create a VPN gateway and configure forced tunneling.

Exercise

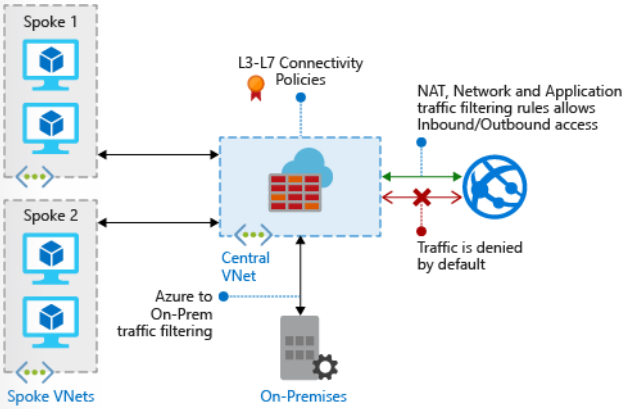
Create a VPN gateway and configure forced tunneling.

Use the instructions **here**[59]. This exercise will take about one and a half hours.

Create firewall rules

Azure Firewall is a managed, cloud-based network security service that helps protect your Azure Virtual Network resources. It's a fully stateful firewall in an SaaS model that has built-in high availability and unrestricted cloud scalability.

58 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-forced-tunneling
59 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection



You can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks. Azure Firewall uses a static public IP address for your virtual network resources, allowing outside firewalls to identify traffic originating from your virtual network. The service is fully integrated with Azure Monitor for logging and analytics.

Some of the key features of Azure Firewall are:

- Built-in high availability. You don't need additional load balancers.
- Unrestricted cloud scalability. You don't need to budget for peak traffic times.
- Application FQDN filtering rules. You can limit outbound HTTP and HTTPS traffic to a specified list of FQDNs, including wildcards. This feature doesn't require SSL termination.
- Network traffic filtering rules. You can centrally create allow or deny network filtering rules by source and destination IP addresses, port, and protocol. Azure Firewall is fully stateful, so it can distinguish legitimate packets for different types of connections. It enforces and logs rules across multiple subscriptions and virtual networks.
- FQDN tags. FQDN tags make it easy for you to allow well-known Azure service network traffic through your firewall. For example, if you want to allow Windows Update network traffic through your firewall, you create an application rule and include the Windows Update tag. Then, network traffic from Windows Update can flow through your firewall.
- Outbound SNAT support. SNAT translates all outbound virtual network traffic IP addresses to the Azure Firewall public IP address. You can identify and allow traffic originating from your virtual network to remote internet destinations.
- Inbound Destination Network Address Translation (DNAT) support. DNAT translates inbound network traffic to your firewall public IP address and filters that traffic to the private IP addresses in your virtual networks.
- Azure Monitor logging. All events are integrated with Azure Monitor, allowing you to archive logs to a storage account, stream events to your event hub, or send events to Azure Monitor logs.

Manage and configure Azure Firewall

You can find a list of the current Azure Firewall commands **here**[60].

For example, here's an Azure PowerShell command to create a network firewall:

Az network firewall create –name

60 https://docs.microsoft.com/en-us/cli/azure/ext/azure-firewall/network/firewall?view=azure-cli-latest

–resource-group

–location

–tags

Exercise

Deploy and configure Azure Firewall by using the Azure portal

Controlling outbound network access is an important part of an overall network security plan. For example, you might want to limit access to websites or to the outbound IP addresses and ports.

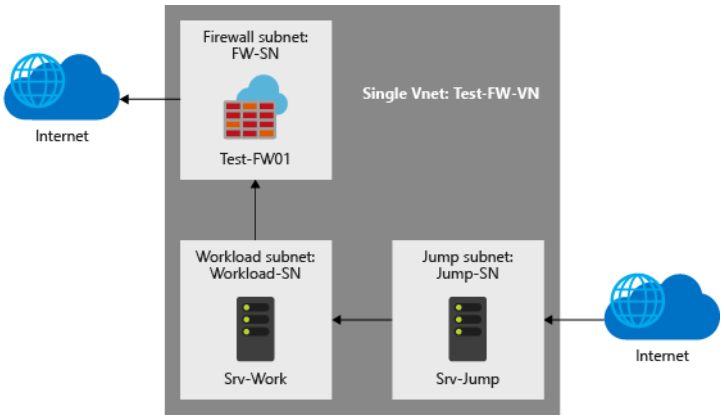
Azure Firewall is one way you can control outbound network access from an Azure subnet. With Azure Firewall, you can configure:

- Application rules defining the FQDNs that can be accessed from a subnet.
- Network rules that define the source address, protocol, destination port, and destination address.

Network traffic is subjected to the configured firewall rules when you route your network traffic to the firewall as the subnet default gateway.

For production deployments, a we recommend a **hub-spoke model**[61], where the firewall exists in its own virtual network, and workload servers exist in peered virtual networks in the same region with one or more subnets. In this exercise, you'll create a simplified virtual network with three subnets for easy deployment:

- **AzureFirewallSubnet**. This subnet contains the firewall.
- **Workload-SN**. This subnet contains the workload server. This subnet's network traffic goes through the firewall.
- **Jump-SN**. This subnet contains the jump server. The jump server has a public IP address you can connect to by using Microsoft Remote Desktop. From there, you can connect to the workload server (by again using Remote Desktop).



In this exercise, you'll learn how to:

- Set up a test network environment.
- Deploy a firewall.
- Create a default route.
- Configure an application to allow access to msn.com.

61 https://docs.microsoft.com/azure/architecture/reference-architectures/hybrid-networking/hub-spoke

- Configure a network rule to allow access to external DNS servers.
- Test the firewall.

You can find the exercise instructions **here**[62].

Configure admin access on the network

Your Azure VMs will need to be securely accessed over the internet or across a dedicated wide area network (WAN) link. Several ways exist to help secure access to Windows and Linux VMs running in Azure:

- Secure Shell (SSH)
- RDP
- Privileged Access Workstations (PAWs) and a VPN

SSH

With an SSH key pair, you can create VMs in Azure that use SSH keys for authentication—eliminating the need for passwords to sign in. First, you'll learn how to quickly generate and use an SSH public-private key file pair for Linux VMs. VMs you create by using SSH keys are configured with passwords disabled by default, which greatly increases the difficulty of brute-force guessing attacks.

Note: Azure currently supports SSH protocol 2 (SSH-2) RSA public-private key pairs with a minimum length of 2048 bits. Azure doesn't support other key formats, such as ED25519 and Elliptic Curve Digital Signature Algorithm (ECDSA).

Create the SSH key pair

Use the **ssh-keygen** command to generate SSH public and private key files. By default, Azure creates these files in the **~/.ssh** directory. You can specify a different location and an optional passphrase to access the private key file. If an SSH key pair with the same name exists in the specified location, Azure overwrites those files.

The following command uses the Bash shell to create an SSH key pair that uses RSA encryption and has a bit length of 2048:

ssh-keygen -t rsa -b 2048

If you use the Azure CLI to create your VM via the **az vm create**[63] command (which you did in an earlier exercise), you can generate SSH public and private key files by using the **–generate-ssh-keys** option. Azure stores the key files in the ~/.ssh directory unless otherwise specified via the **–ssh-dest-key-path** option. The **–generate-ssh-keys** option doesn't overwrite existing key files but instead returns an error. In the following command for the Azure CLI, replace **VMname and RGname** with your own values:

az vm create –name VMname –resource-group RGname –generate-ssh-keys

To create a Linux VM that uses SSH keys for authentication, specify your SSH public key when creating the VM by using the Azure portal, Azure CLI, or Resource Manager templates.

Create a Linux VM in Azure by using Azure PowerShell

1. Launch Azure Cloud Shell.
2. Create an SSH key pair:

62 https://docs.microsoft.com/en-us/azure/architecture/guide/design-principles/scale-out
63 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/convert-unmanaged-to-managed-disks

ssh-keygen -t rsa -b 2048

1. Create a resource group (remember that a resource group is a logical container in which Azure resources are deployed and managed):

New-AzResourceGroup -Name “myResourceGroup” -Location "WestUS"

1. Create a virtual network, subnet, and public IP address:

Create a subnet configuration.

$subnetConfig = New-AzVirtualNetworkSubnetConfig `

-Name “mySubnet” `

-AddressPrefix 192.168.1.0/24

Create a virtual network.

$vnet = New-AzVirtualNetwork `

-ResourceGroupName “myResourceGroup” `

-Location “WestUS” `

-Name “myVNET” `

-AddressPrefix 192.168.0.0/16 `

-Subnet $subnetConfig

Create a public IP address, and specify a DNS name.

$pip = New-AzPublicIpAddress `

-ResourceGroupName “myResourceGroup” `

-Location “WestUS” `

-AllocationMethod Static `

-IdleTimeoutInMinutes 4 `

-Name “mypublicdns$(Get-Random)”

1. Create an Azure NSG and a traffic rule. The NSG helps secure the VM via inbound and outbound rules. The following example creates an inbound rule for TCP port 22 that allows SSH connections. To allow incoming web traffic, the example an also creates an inbound rule for TCP port 80:

Create an inbound NSG rule for port 22.

$nsgRuleSSH = New-AzNetworkSecurityRuleConfig `

-Name “myNetworkSecurityGroupRuleSSH” `

-Protocol “Tcp” `

-Direction “Inbound” `

-Priority 1000 `

-SourceAddressPrefix * `

```
-SourcePortRange * `
-DestinationAddressPrefix * `
-DestinationPortRange 22 `
-Access "Allow"
```

Create an inbound NSG rule for port 80.

```
$nsgRuleWeb = New-AzNetworkSecurityRuleConfig `
-Name "myNetworkSecurityGroupRuleWWW" `
-Protocol "Tcp" `
-Direction "Inbound" `
-Priority 1001 `
-SourceAddressPrefix * `
-SourcePortRange * `
-DestinationAddressPrefix * `
-DestinationPortRange 80 `
-Access "Allow"
```

Create an NSG.

```
$nsg = New-AzNetworkSecurityGroup `
-ResourceGroupName "myResourceGroup" `
-Location "WestUS" `
-Name "myNetworkSecurityGroup" `
-SecurityRules $nsgRuleSSH,$nsgRuleWeb
```

Notice that at this point, you have arranged the network and built a container for deploying the VM into.

1. Create a virtual network adapter by using **New-AzNetworkInterface**. The virtual network adapter connects the VM to a subnet, NSG, and public IP address:

Create a virtual network adapter and associate it with a public IP address and NSG.

```
$nic = New-AzNetworkInterface `
-Name "myNic" `
-ResourceGroupName "myResourceGroup" `
-Location "WestUS" `
-SubnetId $vnet.Subnets[0].Id `
-PublicIpAddressId $pip.Id `
```

```
-NetworkSecurityGroupId $nsg.Id
```

1. Create a VM. Note that to create a VM by using Azure PowerShell, you create a configuration that has settings like the image to use, size, and authentication options. Then, you use the configuration to build the VM. Define the SSH credentials, OS information, and VM size. In this example, the SSH key is stored in **~/.ssh/id_rsa.pub**:

```
# Define a credential object.
$securePassword = ConvertTo-SecureString ' ' -AsPlainText -Force
$cred = New-Object System.Management.Automation.PSCredential ("azureuser", $securePassword)

# Create a VM configuration.
$vmConfig = New-AzVMConfig `
-VMName "myVM" `
-VMSize "Standard_D1" | `
Set-AzVMOperatingSystem `
-Linux `
-ComputerName "myVM" `
-Credential $cred `
-DisablePasswordAuthentication | `
Set-AzVMSourceImage `
-PublisherName "Canonical" `
-Offer "UbuntuServer" `
-Skus "16.04-LTS" `
-Version "latest" | `
Add-AzVMNetworkInterface `
-Id $nic.Id

# Configure the SSH key.
$sshPublicKey = cat ~/.ssh/id_rsa.pub
Add-AzVMSshPublicKey `
-VM $vmconfig `
-KeyData $sshPublicKey `
-Path "/home/azureuser/.ssh/authorized_keys"
```

1. Combine the previous configuration definitions to create a VM by using **New-AzVM**[64] (it will take a few minutes for your VM to be deployed):

```
New-AzVM `
```

64 https://docs.microsoft.com/powershell/module/az.compute/new-azvm

-ResourceGroupName “myResourceGroup” `

-Location eastus -VM $vmConfig

Connect to the VM

Create an SSH connection with the VM by using the public IP address. To see the public IP address of the VM, use the **Get-AzPublicIpAddress** cmdlet:

Get-AzPublicIpAddress -ResourceGroupName “myResourceGroup” | Select "IpAddress"

Using the same shell, you used to create your SSH key pair (like Azure Cloud Shell or your local Bash shell) paste the SSH connection command into the shell to create an SSH session, and use the public IP address you got earlier—for example:

ssh azureuser@10.111.12.123

When prompted, enter the user name of **azureuser**. If you’re using a passphrase with your SSH keys, enter that when prompted.

Clean up

When you no longer need the resource group, VM, and related resources, use the **Remove-AzResource-Group** cmdlet to remove them:

Remove-AzResourceGroup -Name “myResourceGroup”

To generate and use SSH keys on a computer running Windows to create and connect to a Linux VM in Azure, go **here**[65].

RDP

With RDP, you can access the desktop of a role running in Azure. You can use a Remote Desktop connection to troubleshoot and diagnose problems with your application while it’s running.

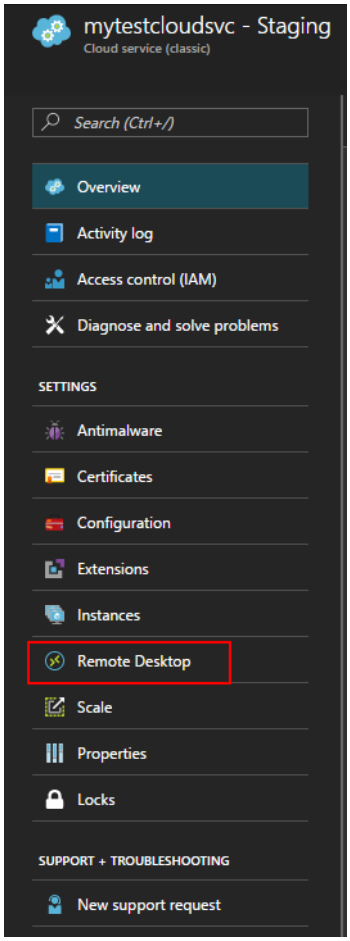
You can enable a Remote Desktop connection in your role during development by including the Remote Desktop modules in your service definition, or you can enable Remote Desktop through the Remote Desktop Extension. We recommend using the Remote Desktop Extension, because you can then enable Remote Desktop even after the application is deployed without having to redeploy your application.

The Azure portal uses the Remote Desktop Extension approach, so you can enable Remote Desktop even after the application is deployed. The Remote Desktop settings for your cloud service allow you to enable Remote Desktop, change the local Administrator account used to connect to the VMs, set the certificate used in authentication, and set the expiration date.

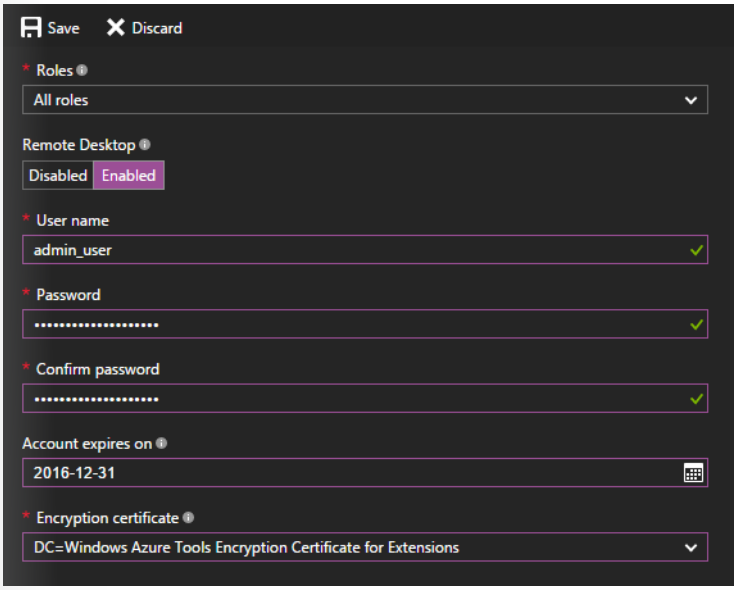
Configure Remote Desktop from the Azure portal

1. Select Cloud Services, select the name of the cloud service, and then select Remote Desktop.

65 https://docs.microsoft.com/en-us/azure/virtual-machines/linux/ssh-from-windows



1. Choose whether you want to enable Remote Desktop for an individual role or all roles, and then select **Enabled**.
2. Fill in the required boxes for the user name, password, expiration date, and certificate.



1. In **Roles**, select either the role you want to update or **All** for all roles.

5.When you finish your configuration updates, select **Save**. It will take a few moments before your role instances are ready to receive connections.

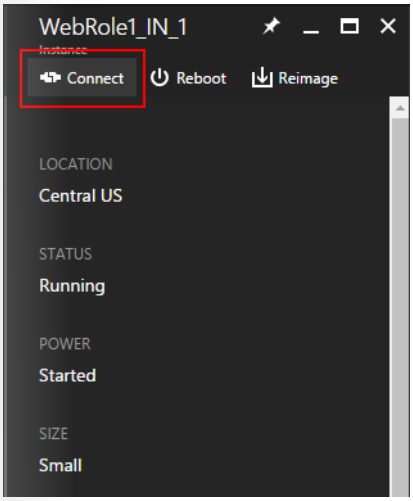
Connect to the role instances via RDP

After Remote Desktop is enabled on the roles, you can initiate a connection directly from the Azure portal:

1.Select **Instances** to open the **Instances** settings.

2.Select a role instance that has Remote Desktop configured.

3.Select **Connect** to download an RDP file for the role instance.



4.Select Open, and then select Connect to start the Remote Desktop connection.

Note:

If your cloud service exists behind an NSG, you might need to create rules that allow traffic on ports 3389 and 20000. Remote Desktop uses port 3389. Cloud service instances are load balanced, so you can't

directly control which instance to connect to. The **RemoteForwarder and RemoteAccess** agents manage RDP traffic and allow the client to send an RDP cookie and specify an individual instance to connect to. These agents require port 20000 to be open, but that port might be blocked if you have an NSG.

Using Privileged Access Workstations (PAWs) and a VPN

Because the vast majority of attacks target the user, the endpoint becomes one of the primary points of attack. A malicious hacker who compromises the endpoint can use the user's credentials to gain access to the organization's data. Most endpoint attacks take advantage of the fact that users are administrators in their local workstations.

A PAW provides a dedicated OS for sensitive tasks that helps provide protection from internet attacks and threat vectors. Separating these sensitive tasks and accounts from the daily use workstations and devices provides very strong protection from phishing attacks; application and OS vulnerabilities; various impersonation attacks; and credential theft attacks, such as keystroke logging, Pass-the-Hash attacks, and pass-the-ticket attacks.

Best practice	Solution
Use a security-enhanced management workstation to help protect sensitive accounts, tasks, and data.	Use a PAWs to reduce the attack surface of workstations. These security-enhanced management workstations can help you mitigate some of these attacks and make data safer.
Get powerful endpoint protection.	Enforce security policies across all devices that are used to consume data, regardless of the data location (cloud or on-premises).

PAW reference[66]

To help ensure that Azure VPN access originates from an organization's PAW system, configure the Azure firewall to allow RPD sessions that originate only from the assigned range of PAW IP addresses.

Configure Azure DDoS Protection

A denial of service attack (DoS) is an attack that has the goal of preventing access to services or systems. If the attack originates from one location, it is called a DoS. If the attack originates from multiple networks and systems, it is called distributed denial of service (DDoS).

Before learning more about DDoS, you need to know what botnets are. Botnets are collections of internet-connected systems that an individual controls and uses without their owners' knowledge. Botnet owners use them to perform various actions of their choosing.

Often, they use them for spamming, data storage, DDoS, or various other actions that are up to the person in control of the botnet. In the past, botnets were made up just of compromised computers, but now, botnets are also made up of Internet of Things (IoT) devices. Malicious hackers can get these poorly secured security cameras, digital video recorders, thermostats, and other internet-connected devices under their control.

So, DDoS is a collection of attack types aimed at disrupting the availability of a target. These attacks involve a coordinated effort that uses multiple internet-connected systems to launch many network

66 https://docs.microsoft.com/en-us/windows-server/identity/securing-privileged-access/privileged-access-workstations

requests against DNS, web services, email, and more. Pretty much any application that the malicious hacker can access might become the target of a DDoS. The malicious hacker's goal is to overwhelm system resources on targeted servers so they can no longer process legitimate traffic, effectively making the system inaccessible.

A DDoS generally involves many systems sending traffic to targets as part of a botnet. In most cases, the owners of the systems in a botnet don't know that their devices are compromised and participating in an attack. Botnets are becoming a bigger problem than before because of the increasing numbers of connected devices.

Designing and building for DDoS resiliency requires planning and designing for a variety of failure modes. The following table lists the best practices for building DDoS-resilient services in Azure.

Best practice 1

Ensure that security is a priority throughout the entire lifecycle of an application, from design and implementation to deployment and operations.
Applications might have bugs that allow a relatively low volume of requests to use a lot of resources, resulting in a service outage.

Solution 1

To help protect a service running in Azure, understand your application architecture, and focus on the **five pillars of software quality.**[67] You should know typical traffic volumes, the connectivity model between the application and other applications, and the service endpoints that are exposed to the public internet.

Helping ensure that an application is resilient enough to handle a DoS targeted at the application itself is most important. Security and privacy features are built in to the Azure platform, beginning with the **Microsoft Security Development Lifecycle (SDL)**[68]. The SDL addresses security at every development phase and ensures that Azure is continually updated to make it even more secure.

Best practice 2

Design your applications to **scale horizontally**[69] to meet the demands of an amplified load—specifically, in the event of a DDoS. If your application depends on a single instance of a service, it creates a single point of failure.
Provisioning multiple instances makes your system more resilient and more scalable.

Solution 2

For Azure App Service, select an **App Service plan**[70] that offers multiple instances.

For Azure Cloud Services, configure each of your roles to use **multiple instances**[71].

For Azure Virtual Machines, ensure that your VM architecture includes more than one VM and that each VM is included in an

67 https://docs.microsoft.com/en-us/azure/architecture/guide/pillars
68 https://www.microsoft.com/en-us/securityengineering/sdl/
69 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
70 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
71 https://docs.microsoft.com/en-us/cli/azure/vm

availability set[72]. We recommend using
virtual machine scale sets[73] for autoscaling capabilities.

Best practice 3

Layer security defenses in an application to reduce the chance of a successful attack. Implement security-enhanced designs for your applications by using the built-in capabilities of the Azure platform.

Solution 3

Be aware that the risk of attack increases with the size, or surface area, of the application. You can reduce the surface area by using whitelisting to close down the exposed IP address space and listening ports that aren't needed on the load balancers (for Azure Load Balancer and Azure Application Gateway).

You can also use NSGs to reduce the attack surface. You can use **service tags**[74] and **application security groups**[75] as a natural extension of an application's structure to minimize complexity for creating security rules and configuring network security.

Azure DDoS Protection

Azure DDoS Protection combined with application design best practices helps defend against DDoS. Azure DDoS Protection provides the following service tiers:

- Basic. This service tier is automatically enabled as part of the Azure platform. It includes always-on traffic monitoring and the real-time mitigation of common network-level attacks, providing the same defenses that Microsoft Online Services uses. The entire scale of the Azure global network can be used to distribute and mitigate attack traffic across regions. This service tier provides powerful protection for Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6) Azure public IP addresses.
- Standard. This service tier provides additional mitigation capabilities over the Basic service tier that are tuned specifically to Azure Virtual Network resources. DDoS Protection Standard is simple to enable and requires no application changes. Protection policies are tuned through dedicated traffic monitoring and machine learning algorithms. Policies are applied to public IP addresses associated to resources deployed in virtual networks, such as Azure Load Balancer, Azure Application Gateway, and Azure Service Fabric instances, but this protection doesn't apply to Azure App Service Environment. Azure Monitor views make real-time telemetry available during an attack and for history. Rich attack mitigation analytics are available via diagnostic settings. You can add application layer protection through the **Azure Application Gateway web application firewall**[76] or by installing a third-party firewall from Azure Marketplace. This service tier helps protect IPv4 Azure public IP addresses.

DDoS Protection Standard can mitigate the following types of attacks:

- Volumetric attacks. The goal of these attacks is to flood the network layer with a substantial amount of seemingly legitimate traffic. They include UDP floods, amplification floods, and other spoofed-packet floods. DDoS Protection Standard mitigates these potentially multiple-gigabyte attacks by automatically absorbing and scrubbing them with Azure's global network scale.
- Protocol attacks. These attacks render a target inaccessible by exploiting a weakness in the layer 3 and layer 4 protocol stack. They include SYN floods, reflection attacks, and other protocol attacks. DDoS

72 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability
73 https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/overview
74 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/managed-disks-overview
75 https://docs.microsoft.com/en-us/azure/security-center/security-center-alerts-type
76 https://docs.microsoft.com/en-us/cli/azure/security?toc=%2fazure%2fvirtual-network%2ftoc.json

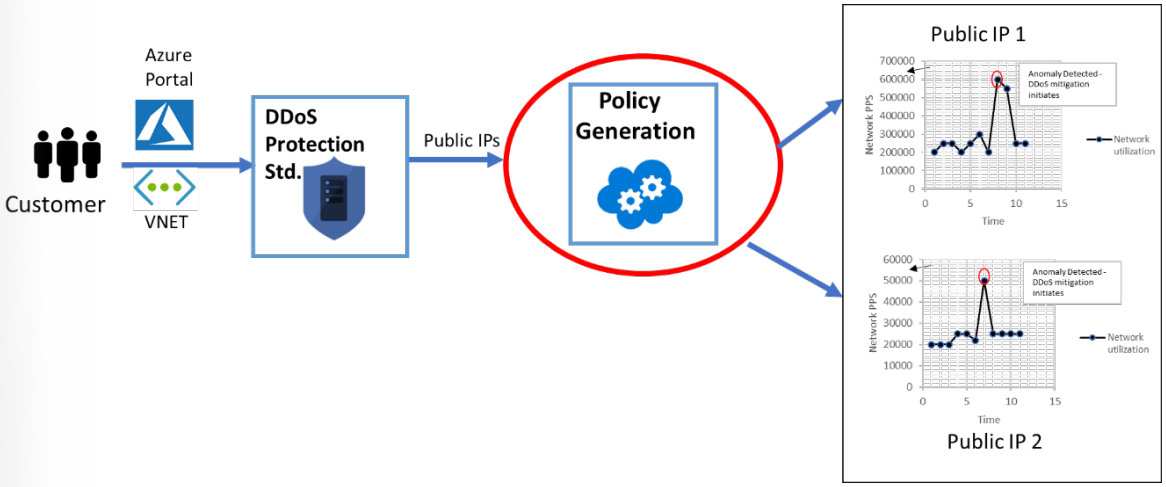
Protection Standard mitigates these attacks, differentiating between malicious and legitimate traffic, by interacting with the client and blocking malicious traffic.

- Resource layer (application layer) attacks. These attacks target web application packets to disrupt the transmission of data between hosts. The attacks include HTTP protocol violations, SQL injection, cross-site scripting, and other layer 7 attacks. Use the Azure Application Gateway web application firewall with DDoS Protection Standard to help defend against these attacks. Azure Marketplace also offers third-party web application firewalls.

DDoS Protection Standard helps protect resources in a virtual network, including public IP addresses associated with VMs, load balancers, and application gateways. When coupled with the Application Gateway web application firewall, DDoS Protection Standard helps provide full layer 3 to layer 7 mitigation capabilities.

How Azure DDoS Protection works

DDoS Protection Standard monitors actual traffic utilization and constantly compares it against the thresholds defined in the DDoS policy. When the traffic threshold is exceeded, DDoS mitigation is automatically initiated. When traffic returns to a level below the threshold, the mitigation is removed.



During mitigation, DDoS Protection redirects traffic sent to the protected resource and performs several checks, including:

- Helping ensure that packets conform to internet specifications and aren't malformed.
- Interacting with the client to determine if the traffic might be a spoofed packet (for example, using SYN Auth or SYN Cookie or dropping a packet for the source to retransmit it).
- Using rate-limit packets if it can't perform any other enforcement method.

DDoS Protection blocks attack traffic and forwards the remaining traffic to its intended destination. Within a few minutes of attack detection, you'll be notified with Azure Monitor metrics. By configuring logging on DDoS Protection Standard telemetry, you can write the logs to available options for future analysis. Azure Monitor retains metric data for DDoS Protection Standard for 30 days.

Configuration example for Azure DDoS Protection

Before completing any steps in this tutorial, sign in to the Azure portal at https://portal.azure.com with an account assigned to either the **Network Contributor role**[77] or a **custom role**[78] that is assigned the appropriate permissions.

Create a DDoS protection plan

A DDoS protection plan defines a set of virtual networks that have DDoS Protection Standard enabled across subscriptions. You can configure one DDoS protection plan for your organization and link virtual networks from multiple subscriptions to the same plan. The DDoS protection plan is associated with a subscription that you select during the plan creation. That subscription incurs the monthly recurring bill for the plan and any overage charges, in case the number of protected public IP addresses exceeds 100.

Most organizations don't require more than one plan. You can't move a plan between subscriptions. If you want to change the subscription for a plan, delete the existing plan, and then create a new one. To create a DDoS protection plan:

1. In the upper-left corner of the Azure portal, select **Create a resource**.
2. earch for **DDoS**. When **DDoS protection plan** appears in the search results, select it.
3. Select **Create**.
4. Enter or select your own values, or use the example values in the following table, and then select **Create**.

Setting	Value
Name	myDDosProtectionPlan
Subscription	Your subscription
Resource group	myResourceGroup (after selecting **Create new**
Location	East US

Enable DDoS for an existing virtual network

1. If you don't have an existing DDoS protection plan, create one by completing the steps in **Create a DDoS protection plan**[79].
2. In the upper-left corner of the Azure portal, select **Create a resource**.
3. In the **Search resources, services, and docs** box at the top of the portal, enter the name of the virtual network that you want to enable DDoS Protection Standard for. When the name of the virtual network appears in the search results, select it.
4. Under **SETTINGS**, select **DDoS protection**.
5. Select **Standard**. Under **DDoS protection plan**, select either an existing DDoS protection plan or the plan you created in step 1, and then select **Save**.
   The plan you select can be in the same or a different subscription than the virtual network, but both subscriptions need to be associated to the same Azure AD tenant.

Here are some additional exercises:

- Enable DDoS for a new virtual network, located **here**[80].
- Disable DDoS for a virtual network, located **here**[81].

77 https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles?toc=%2fazure%2fvirtual-network%2ftoc.json
78 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/ps-template?toc=%2fazure%2fvirtual-network%2ftoc.json
79 https://docs.microsoft.com/en-us/azure/security/azure-security-antimalware
80 https://docs.microsoft.com/en-us/azure/azure-resource-manager/management-groups-create
81 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection

- Work with DDoS protection plans, located **here**[82].
- Configure alerts for DDoS protection metrics, located **here**[83].
- Use DDoS protection telemetry, located **here**[84].
- View DDoS mitigation policies, located **here**[85].
- Configure DDoS attack analytics, located **here**[86].
- Configure DDoS attack mitigation reports, located **here**[87].
- Configure DDoS attack mitigation flow logs, located **here**[88].
- Validate DDoS detection, located **here**[89].
- Understand permissions, located **here**[90].

82 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
83 https://azuremarketplace.microsoft.com/en-us/marketplace/apps/category/networking
84 https://docs.microsoft.com/en-us/cli/azure/account
85 https://docs.microsoft.com/en-us/azure/app-service/overview-hosting-plans
86 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
87 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
88 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
89 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability
90 https://docs.microsoft.com/en-us/azure/firewall/tutorial-firewall-deploy-portal

Implementing host security

Configure endpoint security

Computer systems that interact directly with users are considered endpoint systems. Systems on devices, such as laptops, smartphones, tablets, and computers, all need to be secured to help prevent them from acting as gateways for security attacks on an organization's networked systems.

Earlier, this module discussed the shared responsibilities of helping secure services in Azure. IaaS involves more customer responsibility than PaaS and SaaS did, and Azure Security Center provides the tools you need to harden your network, help secure your services, and stay on top of your security posture.

First step: Help protect against malware

Install antimalware to help identify and remove viruses, spyware, and other malicious software. You can install **Microsoft Antimalware**[91] or an endpoint protection solution from a Microsoft Partner.

Next, integrate your antimalware solution with Azure Security Center to monitor the status of the anti-malware protection. Security Center reports this on the **Endpoint protection issues** blade. Security Center highlights issues, such as detected threats and insufficient protection, which might make your VMs and computers vulnerable to malware threats. By using the information on Endpoint protection issues, you can make a plan to address any identified issues.

Exercise

Enable and configure antimalware for VMs

To enable and configure Microsoft Antimalware for Azure VMs by using the Azure portal while provisioning a VM, complete the following steps:

1. Sign in to the Azure portal at **https://portal.azure.com**[92].
2. To create a new VM, navigate to **Virtual machines**, select **Add**, and then select **Windows Server**.
3. Enter the requested information. Enter a name, username, and password, and then either create a new resource group or select an existing one.
4. Select **OK**.
5. Select a VM size.
6. Select the version of Windows Server that you want to use.
7. Select **Review + Create**.
8. In the next section, make the appropriate choices for your needs, and then select the **Extensions** section.
9. Select **Add extension**
10. Under **New resource**, select **Microsoft Antimalware**.
11. Select **Create**.
12. In the **Install extension** section, you can configure files, locations, process exclusions, and other scan options. Select**OK**.
13. Select **OK**.

91 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
92 https://portal.azure.com/

14. Back in the **Settings** section, select **OK**.

15. On the **Create** screen, select **OK**.

Here are other methods for deploying Microsoft Antimalware:

- For deployment via Visual Studio VM configuration, go **here**[93].
- For deployment via Azure PowerShell cmdlets, go **here**[94].
- To enable and configure antimalware monitoring by using Azure PowerShell cmdlets, go **here**[95].

Second step: Monitor the status of antimalware

Azure Security Center monitors the status of antimalware protection and reports this on the **Endpoint protection issues** blade. Security Center notes issues, such as detected threats and insufficient protection, that might make your VMs and computers vulnerable to malware threats. By using the information on **Endpoint protection issues**, you can make a plan to address any identified issues.

Security Center reports the following endpoint protection issues:

- Endpoint protection not installed on Azure VMs. A supported antimalware solution isn't installed on these Azure VMs.
- Endpoint protection not installed on non-Azure computers. A supported antimalware solution isn't installed on these non-Azure computers.
- Endpoint protection health issues:
-
```
  - **Signature out of date**. An antimalware solution is installed on
these VMs and computers, but the solution doesn't have the latest antimal-
ware signatures.

  - **No real time protection**. An antimalware solution is installed on
these VMs and computers, but it isn't configured for real-time protection.
The service might be disabled, or Security Center might be unable to obtain
the status because the solution isn't supported.

  - **Not reporting**. An antimalware solution is installed but not report-
ing data.

  - **Unknown**. An antimalware solution is installed, but either its
status is unknown or it's reporting an unknown error.
```

Security Center can also install antimalware solutions on Azure and non-Azure VMs.

Configure VM security by using templates or VM-level policies

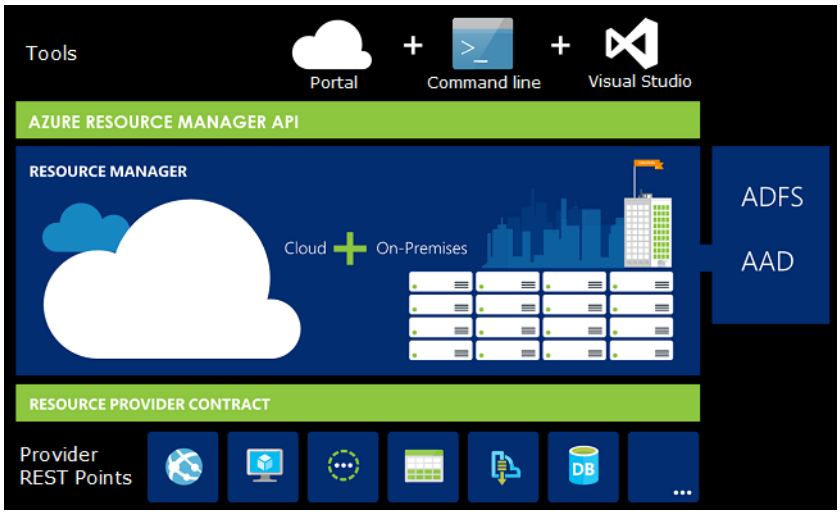
Before diving into configuring VM policies and templates, you need to understand the features and functionality of Azure Resource Manager.

Resource Manager is the deployment and management service for your Azure subscription. It provides a consistent management layer that allows you to create, update, and delete resources in your Azure

93 https://docs.microsoft.com/en-us/azure/security/azure-security-antimalware
94 https://docs.microsoft.com/en-us/azure/security/azure-security-antimalware
95 https://docs.microsoft.com/en-us/azure/virtual-machines/linux/tutorial-monitoring

subscription. You can use its access control, auditing, and tagging features to help secure and organize your resources after deployment.

When you take actions through the portal, Azure PowerShell, the Azure CLI, REST APIs, or client SDKs, the Resource Manager API handles your request. Because the same API handles all requests, you get consistent results and capabilities in all the different tools. The following figure depicts this point.



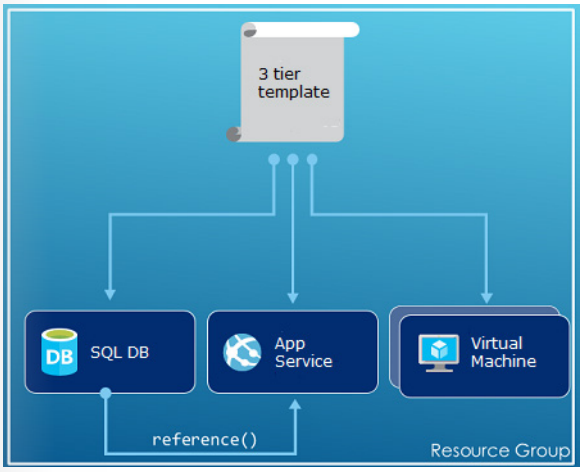
When you create something in Azure, the request goes through the Resource Manager API. If it's successful, a resource is created and assigned to a resource group. Examples of resources include VMs, web apps, and Azure SQL databases. Typically, some resources are related to each other. In that case, use Resource Manager to deploy these resources into the same resource group. You can then deploy, update, manage, and monitor them together.

Here are some additional terms to know when using Resource Manager:

- **Resource provider**. A service that supplies Azure resources. For example, a common resource provider is Microsoft.Compute, which supplies the VM resource. Microsoft.Storage is another common resource provider.
- **Resource Manager template**. A JSON file that defines one or more resources to deploy to a resource group or subscription. You can use the template to consistently and repeatedly deploy the resources.
- **Declarative syntax**. Syntax that lets you state, "Here's what I intend to create" without having to write the sequence of programming commands to create it. The Resource Manager template is an example of declarative syntax. In the file, you define the properties for the infrastructure to deploy to Azure.

You can use the Resource Manager template to define your VMs. After they are defined you can easily deploy and redeploy them. We recommend periodically redeploying your VMs to force the deployment of a freshly updated and security-enhanced VM OS.

How you define templates and resource groups is entirely up to you and how you want to manage your solution. For example, you can deploy a three-tier application through a single template to a single resource group, as the following figure depicts.



To learn about the format of the template and how you construct one, refer to **Understand the structure and syntax of Azure Resource Manager Templates**[96]. To learn the JSON syntax for resources types, refer to **Define resources in Azure Resource Manager templates**[97].

Resource Manager processes the template like any other request. It parses the template and converts its syntax into REST API operations for the appropriate resource providers.

Access control is an additional security feature of Resource Manager that you can apply to all services in your resource group, because role-based access control (RBAC) is natively integrated into the management platform.

Exercise

Take a few minutes to try **this exercise**[98], in which you'll deploy a Resource Manager template by using Azure PowerShell. The template you create deploys a single VM running Windows Server in a new virtual network with a single subnet.

In this exercise, you'll:

- Launch Azure Cloud Shell.
- Create the required resource group.
- Create the template files to deploy resources.
- Create a storage account and upload files.
- Deploy the template.

✓ Don't worry about having to type the JSON code for the Resource Manager template. Simply use the copy functionality that the exercise provides.

Harden VMs in Azure

Azure Security Center helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. Security Center helps you safeguard VM data in Azure

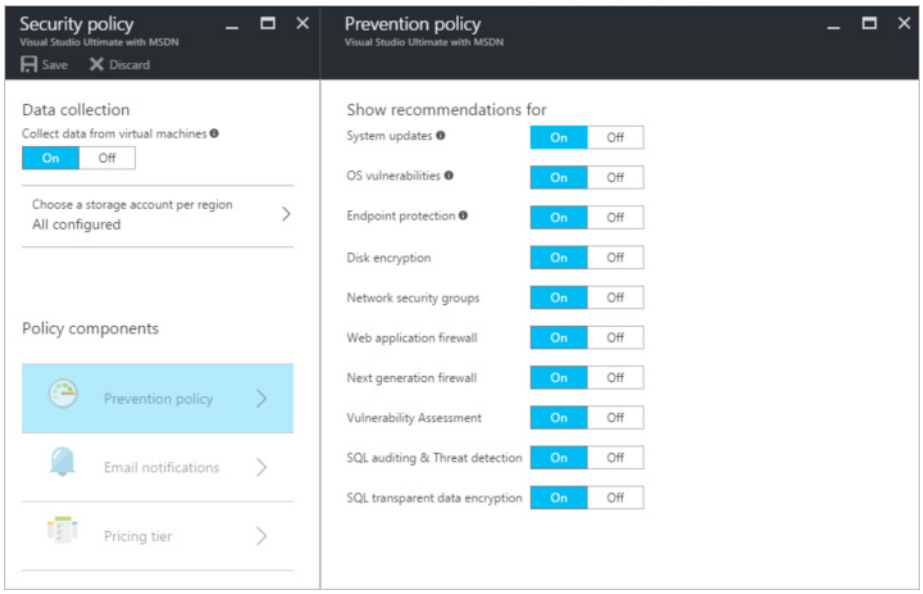
96 https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-authoring-templates
97 https://docs.microsoft.com/en-us/azure/templates/
98 https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-choose-me

by providing visibility into the security settings of your VMs. When Security Center helps safeguard your VMs, the following capabilities are available:

- OS security settings with the recommended configuration rules
- System security updates and critical updates that are missing
- Endpoint protection recommendations
- Disk encryption validation
- Vulnerability assessment and remediation
- Threat detection

Set security policies to manage vulnerabilities for VMs

You need to enable data collection so that Azure Security Center can gather the information it needs to provide recommendations and alerts based on the security policy you configure. In the following figure, data collection has been turned on.



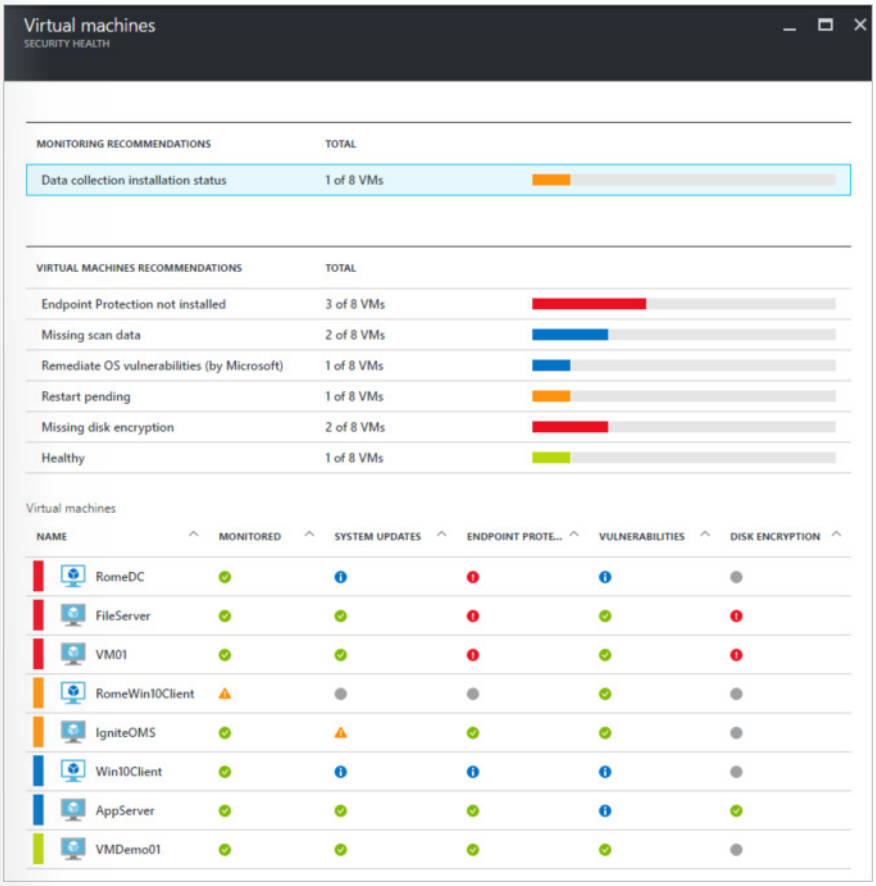
A security policy defines the set of controls recommended for resources within the specified subscription or resource group. Before enabling a security policy, you need to enable data collection. Security Center collects data from your VMs to assess their security state, provide security recommendations, and alert you to threats. In Security Center, you define policies for your Azure subscriptions or resource groups according to your company's security needs and the types of applications or sensitivity of data in each subscription.

Security Center analyzes the security state of your Azure resources. When Security Center identifies potential security vulnerabilities, it creates recommendations. The recommendations guide you through the process of configuring the needed controls.

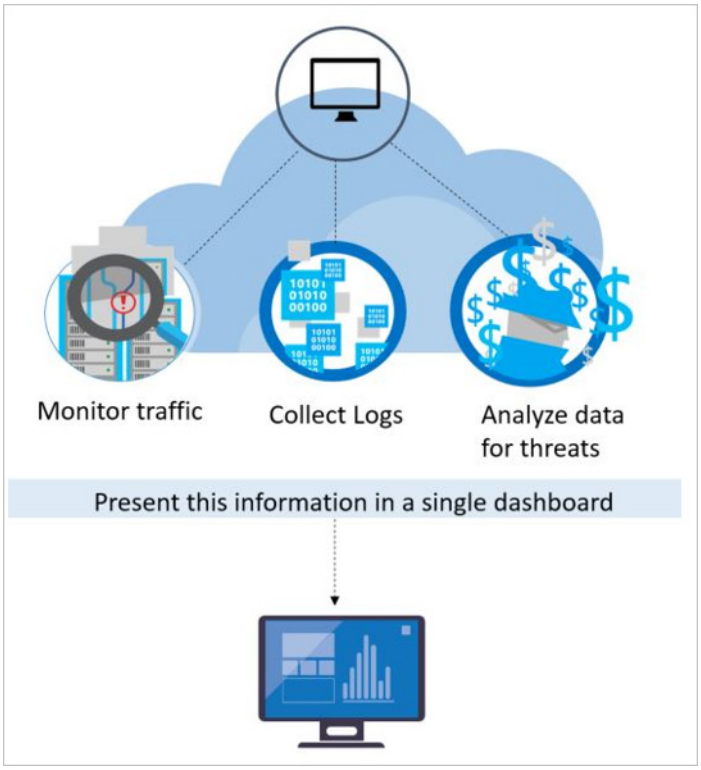
After setting a security policy, Security Center analyzes the security state of your resources to identify potential vulnerabilities. It depicts recommendations in a table format, where each line represents one recommendation. The table **here**[99] has examples of recommendations for Azure VMs and what each will do if you apply it. When you select a recommendation, Security Center provides information about how you can implement that recommendation.

99 https://docs.microsoft.com/en-us/azure/governance/policy/how-to/get-compliance-data

Security Center also monitors and analyzes the enabled security policies to identify potential vulnerabilities. On the **Resource security health** blade, you can check the security state of your resources along with any issues. When you select **Virtual machines** in **Resource security health**, the **Virtual machines** blade opens with recommendations for your VMs, as the following figure depicts.



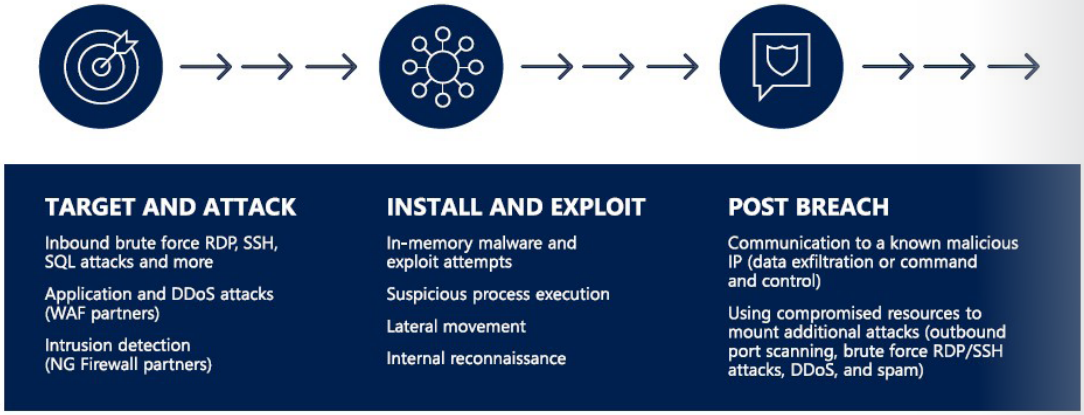
Security Center threat detection works by automatically collecting security information from your Azure resources, the network, and connected partner solutions. It analyzes this information, often correlating information from multiple sources, to identify threats. Security Center prioritizes alerts along with recommendations on how to remediate the threats.



Security Center employs advanced security analytics that go far beyond signature-based approaches. Security Center takes advantage of breakthroughs in big data and machine learning technologies to evaluate events across the entire cloud fabric—detecting threats that would be impossible to identify via manual approaches and predicting the evolution of attacks. These security analytics include:

- Integrated threat intelligence. Seeks known malicious hackers by taking advantage of global threat intelligence from Microsoft products and services, the Microsoft Digital Crimes Unit, the Microsoft Security Response Center, and external feeds.
- Behavioral analytics. Applies known patterns to discover malicious behavior.
- Anomaly detection. Uses statistical profiling to build a historical baseline. It sends alerts on deviations from established baselines that conform to potential attack vectors.

Using these analytics, Security Center can help disrupt the kill chain by adding detection in different phases of the kill chain, as the following figure depicts.



The preceding figure depicts some common alerts for each phase, and several more **types of alerts**[100] exist. Security Center also correlates alerts and creates a **security incident**[101]. Security incidents give you a better view of which alerts belong to the same attack campaign.

Configure system updates in Azure

Azure Update Management is a service included as part of your Azure subscription. With Update Management, you can assess your update status across your environment and manage your Windows Server and Linux server updates from a single location—for both your on-premises and Azure environments.

Update Management is available at no additional cost (you pay only for the log data that Azure Log Analytics stores), and you can easily enable it for Azure and on-premises VMs. To try it, navigate to your **VM** tab in Azure, and then enable Update Management for one or more of your VMs. You can also enable Update Management for VMs directly from your Azure Automation account.

Azure Update Management overview

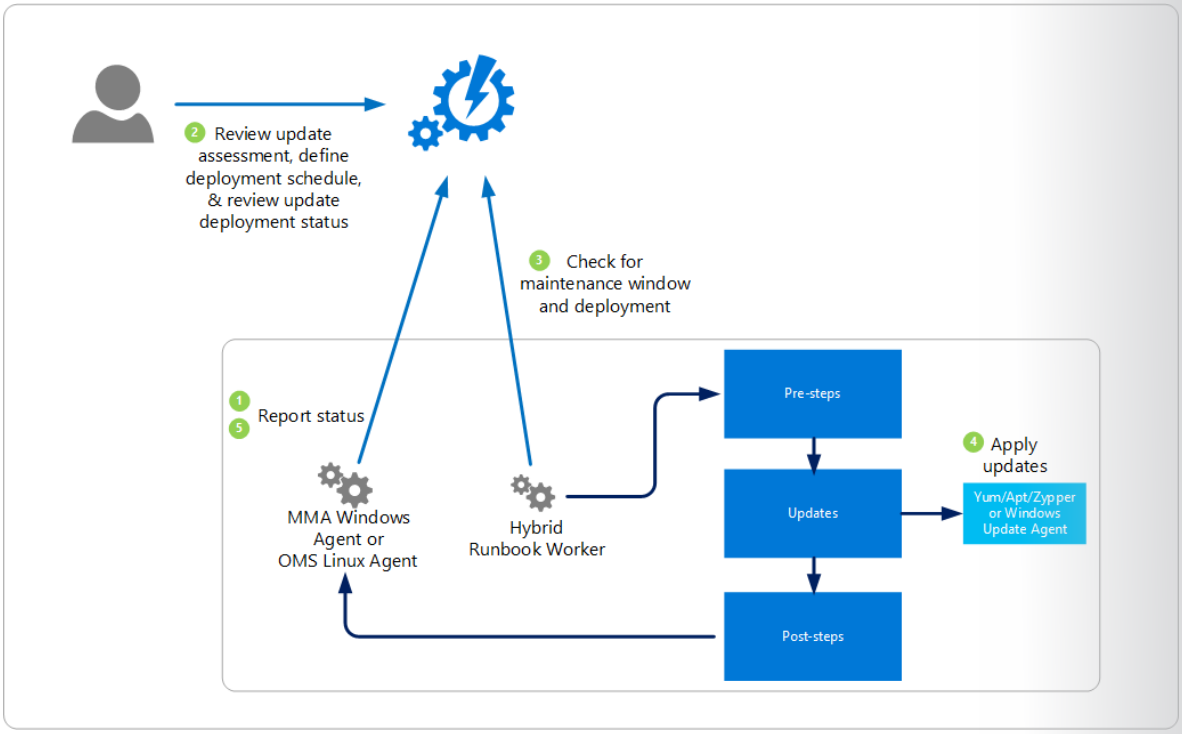
Computers that Update Management manages use the following configurations to perform assessment and update deployments:

- Microsoft Monitoring Agent (MMA) for Windows or Linux
- Desired State Configuration (DSC) in Windows PowerShell for Linux
- Hybrid Runbook Worker in Azure Automation
- Microsoft Update or Windows Server Update Services (WSUS) for Windows computers

Azure Automation uses runbooks to install updates. You can't view these runbooks, and they don't require any configuration. When an update deployment is created, it creates a schedule that starts a master update runbook at the specified time for the included computers. The master runbook starts a child runbook on each agent to install the required updates.

The following diagram is a conceptual depiction of the behavior and data flow together with how the solution assesses and applies security updates to all connected Windows Server and Linux computers in a workspace.

100 https://docs.microsoft.com/en-us/azure/governance/policy/overview
101 https://docs.microsoft.com/en-us/azure/virtual-network/security-overview



Details of the update process can be found **here**[102].

Exercise

Starting a runbook from the Azure portal

1. In the Azure portal, select **Automation**, and then select the name of an automation account.
2. On the **Hub** menu, select **Runbooks**.
3. On the Runbooks page, select a runbook, and then select **Start**.
4. If the runbook has parameters, you'll be prompted to provide values in text boxes, one for each parameter. For more details, refer to "Runbook parameters" **here**[103].
5. On the **Job** page, view the status of the runbook job.

Harding devices connected to Azure

Windows 10, Windows Server 2019, and Windows Server 2016 include key security features. They are Windows Defender Credential Guard, Windows Defender Device Guard, and Windows Defender Application Control.

Windows Defender Credential Guard

Introduced in Windows 10 Enterprise and Windows Server 2016, Windows Defender Credential Guard uses virtualization-based security enhancement to isolate secrets so that only privileged system software can access them. Unauthorized access to these secrets might lead to credential theft attacks, such as Pass-the-Hash or pass-the-ticket attacks. Windows Defender Credential Guard helps prevent these attacks by helping protect Integrated Windows Authentication (NTLM) password hashes, Kerberos authentication ticket-granting tickets, and credentials that applications store as domain credentials.

102 https://docs.microsoft.com/en-us/azure/automation/manage-update-multi
103 https://docs.microsoft.com/en-us/azure/automation/automation-starting-a-runbook

By enabling Windows Defender Credential Guard, you get the following features and solutions:

- Hardware security enhancement. NTLM, Kerberos, and Credential Manager take advantage of platform security features, including Secure Boot and virtualization, to help protect credentials.
- Virtualization-based security enhancement. NTLM-derived credentials, Kerberos-derived credentials, and other secrets run in a protected environment that is isolated from the running operating system.
- Better protection against advanced persistent threats. When virtualization-based security enhancement helps protect Credential Manager domain credentials, NTLM-derived credentials, and Kerberos-derived credentials, the credential theft attack techniques and tools that many targeted attacks use are blocked. Malware running in the OS with administrative privileges can't extract secrets that virtualization-based security helps protect. Although Windows Defender Credential Guard provides powerful mitigation, persistent threat attacks will likely shift to new attack techniques, so you should also incorporate Windows Defender Device Guard and other security strategies and architectures.

Windows Defender Device Guard and Windows Defender Application Control

The configuration state of Windows Defender Device Guard was originally designed with a specific security idea in mind. Although no direct dependencies existed between the two main OS features of the Windows Defender Device Guard configuration—that is, between configurable code integrity and Hypervisor-protected code integrity (HVCI)—the discussion intentionally focused on the Windows Defender Device Guard lockdown state that can be achieved when they're deployed together.

However, the use of the term device guard to describe this configuration state has unintentionally left many IT pros with the impression that the two features are inexorably linked and can't be separately deployed. Additionally, because HVCI relies on security based on Windows virtualization, it comes with additional hardware, firmware, and kernel driver compatibility requirements that some older systems can't meet.

As a result, many IT pros assumed that because some systems couldn't use HVCI, they couldn't use configurable code integrity, either. But configurable code integrity has no specific hardware or software requirements other than running Windows 10, which means that many IT pros were wrongly denied the benefits of this powerful application control capability.

Since the initial release of Windows 10, the world has witnessed numerous hacking and malware attacks where application control alone might have prevented the attack altogether. Configurable code integrity is now documented as an independent technology within the Microsoft security stack and given a name of its own: Windows Defender Application Control.

Application control is a crucial line of defense for helping protect enterprises given today's threat landscape, and it has an inherent advantage over traditional antivirus solutions. Specifically, application control moves away from the traditional application trust model, in which all applications are assumed trustworthy by default, to one where applications must earn trust to run. Many organizations understand this and frequently cite application control as one of the most effective means for addressing the threat of malware based on executable files (such as .exe and .dll files).

Windows Defender Application Control helps mitigate these types of threats by restricting the applications that users can run and the code that runs in the system core, or kernel. Policies in Windows Defender Application Control also block unsigned scripts and MSIs, and Windows PowerShell runs in Constrained language mode.

Does this mean the Windows Defender Device Guard configuration state is going away? Not at all. The term device guard will continue to describe the fully locked down state achieved using Windows Defender Application Control, HVCI, and hardware and firmware security features. It will also allow Microsoft to work with its original equipment manufacturer (OEM) partners to identify specifications for devices that are device guard capable—so that joint customers can easily purchase devices that meet all the hardware

and firmware requirements of the original locked down scenario of Windows Defender Device Guard for Windows 10 devices.

Implement platform security enhancements

Configure custom domains for PaaS

Why use custom domains?

Every domain name in Azure AD is either an initial domain name or a custom domain name. Azure AD comes with an initial domain name in the form company.onmicrosoft.com. This third-level domain name—for example, contoso.onmicrosoft.com—is established when the directory is created, typically by the admin who creates the directory.

The use of custom domains helps ensure that your internal and external URLs are the same. The benefits are as follows:

- Your users will have an easier experience. They don't need to learn different internal and external URLs and track where they are—because the same URL will work for them no matter where they are.
- The links contained in applications will work without additional configurations. If you have hard coded internal links in your applications, they might not work when selected if the internal link isn't published in the Azure AD Application Proxy application and if the link isn't externally resolvable. When your URLs are the same, you avoid this problem. To learn more about options if you're not able to use custom domains, refer to the documentation on translating inline links.
- Some configurations will work only if you have custom domains. For example, you need custom domains for applications that use SAML, such as when you use Active Directory Federation Services but can't use Web Services Federation (WS-Federation). For more details, refer to the documentation on claims-aware applications.
- You can build user confidence. Using an msappproxy.net domain might be unfamiliar for your users, but a custom domain can help build their confidence when accessing your applications.

If you want to make sure your internal users aren't directed through Azure AD Application Proxy, you need to set up a split DNS infrastructure.

You use a split DNS infrastructure to direct internal hosts to an internal domain name server for name resolution and external hosts to an external domain name server for name resolution.

If you can't make the internal and external URLs match, the recommendation isn't as strong, although you might still get some benefits.

Configure a custom domain

Prerequisites

Before you configure a custom domain, prepare the following requirements:

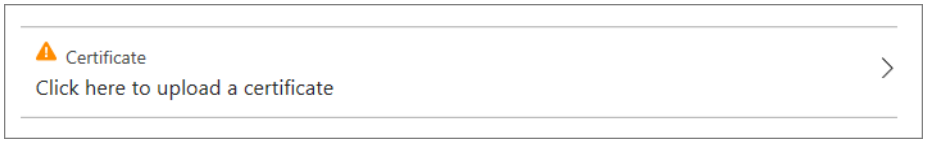
- A verified domain added to Azure AD
- A custom certificate for the domain in the form of a .pfx file
- An on-premises app published through Azure AD Application Proxy

To use or manage a custom domain purchased through the Azure portal, you need an Azure App Service app (web app, mobile app, API app, or function app) that is running in any non-free SKU. (Custom domains aren't available for free apps.)

Configure the domain

1. Sign in to the Azure portal.

2. Navigate to **Azure Active Directory > Enterprise applications > All applications**, and then select the app you want to manage.
3. Select **Application Proxy**.
4. In the **External URL** list, select your custom domain. If the list doesn't have your domain, it hasn't been verified yet.
5. If the list doesn't have the domain you want, **add it as a verified domain**[104].
6. Select **Save**.
7. Note that the **Certificate** option that was disabled becomes enabled. Select it.



If you already uploaded a certificate for this domain, the Certificate option displays the certificate information.

1. Upload the .pfx file containing the certificate, and then enter the password for the certificate.
2. Select **Save** to save your changes.

10.Add a **DNS record**[105] that redirects the new external URL to the msappproxy.net domain.

For more information, refer to "FAQ on App Service Certificates and Custom Domains" at **https://social.msdn.microsoft.com/Forums/en-US/f3e6faeb-5ed4-435a-adaa-987d5db43b80/faq-on-app-service-certificates-and-custom-domains**.

Configure update domains

Sometimes, you need to update your app, or Microsoft needs to update the host that your VMs are running on. Note that with IaaS VMs, Microsoft doesn't automatically update your VMs. You have complete control (and responsibility) for that. But, if Microsoft identifies a serious security vulnerability and creates an update, it's in Microsoft's interest to apply the update to the host of your VM as soon as possible. How is that done without taking your service offline? The answer is by using update domains. This is similar to the fault domain method, but instead of an accidental failure occurring, a purposeful move to take down one or more of your servers occurs. To make sure your service doesn't go offline because of an update, this method goes through your update domains one after the other.

To provide redundancy to your application, we recommend that you group two or more VMs in an availability set. This configuration within a datacenter helps ensure that during a planned or unplanned maintenance event, at least one VM will be available, and it meets the 99.95 percent Azure SLA.

The underlying Azure platform assigns an update domain and a fault domain to each VM in your availability set. For any particular availability set, the platform assigns 5 update domains that aren't user configurable by default (you can then increase Resource Manager deployments to provide up to 20 update domains). These update domains indicate groups of VMs and underlying physical hardware that can be restarted at the same time. When more than 5 VMs are configured within a single availability set, the sixth VM is placed into the same update domain as the first VM, the seventh VM in the same update domain as the second VM, and so on. The order of update domains being restarted might not proceed sequentially during planned maintenance events, but only one update domain is restarted at a time. A

104 https://docs.microsoft.com/en-us/azure/governance/policy/assign-policy-azurecli
105 https://docs.microsoft.com/en-us/azure/security-center/security-center-virtual-machine

restarted update domain is given 30 minutes to recover before maintenance is initiated on a different update domain.

Azure assigns both fault domains and update domains in the order that it discovers them as they are provisioned. The following figure depicts a collection of VMs in a single availability set.

NAME	STATUS	SIZE	UPDATE DOMAIN	FAULT DOMAIN
Srv0	Running	Standard_A0	0	0
Srv1	Running	Standard_A0	1	1
Srv2	Running	Standard_A0	2	0
Srv3	Running	Standard_A0	3	1
Srv4	Running	Standard_A0	4	0
Srv5	Running	Standard_A0	0	1
Srv6	Running	Standard_A0	1	0
Srv7	Running	Standard_A0	2	1
Srv8	Running	Standard_A0	3	0
Srv9	Running	Standard_A0	4	1
Srv10	Running	Standard_A0	1	0

If you are currently using VMs with unmanaged disks, we recommend that you **convert VMs in availability sets to use managed disks**[106].

Managed disks[107] provide better reliability for availability sets by helping ensure that the disks of VMs in an availability set are sufficiently isolated from each other to avoid single points of failure. It does this by automatically placing the disks in different storage fault domains (or storage clusters) and aligning them with the VM fault domains. If a storage fault domain fails because of a hardware or software failure, only the VM instance with disks in the storage fault domain will fail.

Additional best practices

- **Use scheduled events**[108] to proactively respond to VM impacting events.
- Configure the application tiers into **separate availability sets**[109].
- Combine **a load balancer**[110] with availability sets.
- Use **availability zones**[111] to help protect against datacenter-level failures.

Implement Azure functions updates for serverless computing

As discussed earlier in this module, **serverless computing** is the abstraction of servers, infrastructure, and operating systems. When you build serverless apps you don't need to provision and manage any servers. Serverless computing is driven by the reaction to events and triggers happening in near-real-time—in the cloud. As a fully managed service, server management and capacity planning are invisible to the developer and billing is based just on resources consumed or the actual time your code is running.

106 https://azure.microsoft.com/mediahandler/files/resourcefiles/azure-functions-serverless-platform-security/Microsoft Serverless Platform.pdf
107 https://en.wikipedia.org/wiki/Webhook
108 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability
109 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability
110 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
111 https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources

Azure Functions are an example of serverless applications. These functions can power a single-page app. The app calls functions using the **WebHook**[112] URL, saves user data, and decides what data to display. Or, do simple customizations, such as changing ad targeting by calling a function and passing it user profile information.

Updating your Azure Functions

There are multiple ways to update Azure Functions. One of the simplest and direct way is by using a **PUT statement**.

Performing a PUT operation on a specific Azure User Defined Function (UDF) resource replaces the entire UDF resource. All user settable properties, including the id and the body, must be submitted in the body to perform the replacement.

An example is show **here**[113].

Another method for deploying and updating your function is by using App Service continuous integration. Functions integrates with BitBucket, Dropbox, GitHub, and Azure DevOps. This enables a workflow where function code updates made by using one of these integrated services trigger deployment to Azure. Continuous deployment is a great option for projects where multiple and frequent contributions are being integrated. It also lets you maintain source control on your functions code. The following deployment sources are currently supported:

- Bitbucket
- Dropbox
- External repository (Git or Mercurial)
- Git local repository
- GitHub
- OneDrive
- Azure DevOps

Deployments are configured on a per-function app basis. After continuous deployment is enabled, access to function code in the portal is set to read-only.

To be able to deploy from Azure DevOps, you must first link your Azure DevOps organization with your Azure subscription. For more information, see **Set up billing for your Azure DevOps organization**[114].

The procedure for setting up continuous deployment from the supported deployment sources for existing Azure functions can be found **here**[115].

Implement platform updates

We recommend that you can use Update Management in Azure Automation to manage OS updates for your Windows and Linux computers that are deployed in Azure, in on-premises environments, or in other cloud environments. You can quickly assess the status of available updates on all agent computers and manage the process of installing required updates for servers.

112 https://en.wikipedia.org/wiki/Webhook
113 https://docs.microsoft.com/en-us/rest/api/cosmos-db/replace-a-user-defined-function
114 https://docs.microsoft.com/azure/devops/organizations/billing/set-up-billing-for-your-organization-vs?view=vsts
115 https://docs.microsoft.com/en-us/azure/azure-functions/functions-continuous-deployment

You can enable **Update Management for VMs**[116] directly from your Azure Automation account. You can also enable Update Management for a VM from the VM page in the Azure portal. This scenario is available for Linux and Windows VMs.

Computers that Update Management manages use the following configurations to perform assessment and update deployments:

- Microsoft Monitoring Agent (MMA) for Windows or Linux
- DSC in Windows PowerShell for Linux
- Hybrid Runbook Worker in Azure Automation
- Microsoft Update or WSUS for Windows computers

Note: If you use Windows Update, leave the automatic Windows Update setting enabled.

Exercise

For an exercise to enable Update Management, go **here**[117].

Configure Security issues for serverless computing

An enterprise uses serverless architectures to both build and deploy software and services without the need for in-house physical or virtual servers. Serverless computing moves the responsibility for server management from the application owner to the platform provider. This helps eliminate security issues like servers with known security vulnerabilities that haven't been updated and makes DoS more of a billing issue than a security issue.

However, security issues and challenges to consider still exist in serverless computing.

This module discussed the shared responsibility model was earlier. Serverless computing shifts even more responsibility to the cloud provider.

You're still responsible for your application code. Poorly written code is not secure. You're also responsible for data management, data encryption, identity management, authentication and authorization, and the configuration of services and RBAC.

Application-level vulnerabilities (for example, cross-site scripting and SQL injection) continue to be severe if exploited. Mitigation techniques (for example, input validation and programmatic database access) are as critical as ever.

Application dependencies are like exploited server dependencies. They're widespread and downloaded frequently by developers. It's hard to track which packages you're using. And, they're often vulnerable with new vulnerabilities disclosed regularly.

It seems counterintuitive that serverless computing has vulnerabilities to large attack surfaces and increased complexity, but it does. Events from HTTPS APIs, message queues, cloud storage, and IoT device communications are the main source of data consumption for serverless functions. When such messages introduce complex protocols and message structures, the possibility of an attack increases.

Standard application-layer protection mechanisms, such as web application firewalls, can't inspect these kinds of attacks if they can't do sophisticated application-layer inspection. And for serverless functions, you can't put an application-layer inspection device in front of the compute device that's hosting the serverless function.

116 https://docs.microsoft.com/en-us/azure/automation/manage-update-multi
117 https://docs.microsoft.com/en-us/azure/security/azure-security-antimalware

Problems related to transparency occur when debugging a nested service call fails. Third-party services and data in transit make debugging code extremely challenging.

You can use correlation IDs to track the request source, but you need to apply them in every service in a microservice architecture. No mitigation exists for this, so that's why you need to apply such correlation IDs in every service.

Imagining and monitoring serverless architectures is still more complex than doing so for standard software environments. To address this concern, consider each function to be its own security perimeter. Each function needs to sanitize inputs and outputs, help protect its data, and help secure its code and dependencies.

The move to serverless computing increases application fragmentation because of the number of moving functions and services. This fragmentation along with the creation of new services might lead to solutions that are both hard to manage and missing insights and usage patterns that would otherwise be useful when you're evaluating the application for possible security events or indicators of compromise.

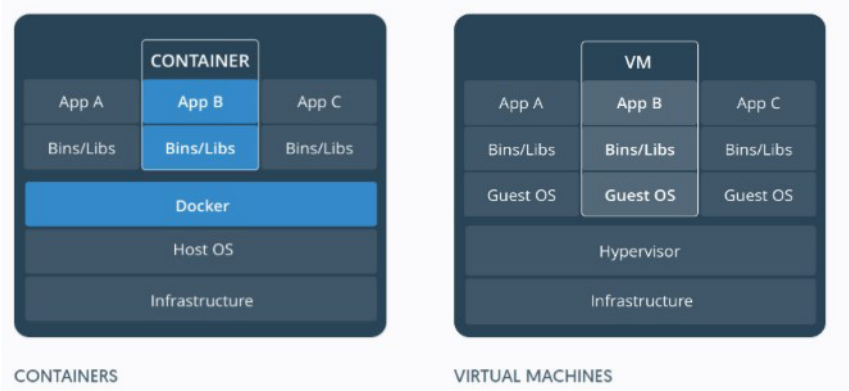
Another security issue with serverless computing is the fire-and-forget convenience of these low-cost functions. These functions are easy to deploy and cost nothing if not invoked, so deleting old functions isn't a pressing issue. This "function litter" increases the attack surface and can become an invitation to malicious hackers.

For a programmer's perspective on the security of Azure serverless computing, refer to the **Azure Functions and Serverless Platform Security**[118] white paper.

Security considerations for containers

First, here's a brief refresher on what a container is.

A container image is a lightweight, standalone, executable package that includes everything needed to run an application. It includes the app, system tools, system libraries, settings for the application, and more. Containers differ from hypervisor-based VMs in that they run on a single copy of an OS that's running on the hardware. The following figure compares containers and VMs.



When an app is containerized, the app and the components needed to run it are combined in a single image. Containers are then created from this image as needed. Companies can use images as a baseline and build upon them to create other images, making subsequent image creation faster. Also, multiple containers can share the same image, which allows containers to start very quickly and use fewer resources. Although containers have existed for decades, tools like Docker have popularized the approach of container images and container management, making them viable in today's datacenter. Historically, con-

118 https://www.microsoft.com/handsonlabs/SelfPacedLabs

tainer technology was exclusively implemented on Linux-based OSs. But in the past few years, other modern OSs, like Windows Server, have embraced container technology. In turn, the container development community has embraced Windows Server.

Security concerns for containers

Containers are not inherently vulnerable. But as with all IT technologies, it's important to adhere to strict guidelines and procedures for security. The following guidelines and procedures address operational practices more than technical practices.

For example, privilege escalation, repository validation, and image signing represent special threats in the container world. These are new constructs, and people unfamiliar with them might not know the proper way to govern and work with them.

Networking in a container deployment is another special area that needs to be addressed in security scenarios. Unlike VMs, containers have open network traffic across services and a shared kernel—which is a serious security concern. However, you can make the case that VMs are less secure than containers, because breaking applications into microservices with well-defined interfaces and limited, packaged services reduces the overall attack surface. To use containers more safely, you need to be aware of the potential security issues and the major tools and techniques for helping secure container-based systems.

Kernel exploits

Unlike in a VM, all containers and the host share the kernel. This sharing magnifies the importance of any vulnerabilities in the kernel.

Denial of service attacks

All containers share kernel resources. If one container can monopolize access to certain resources—including memory and user IDs—it can deprive the other containers on the host. The result is a DoS, whereby legitimate users are unable to access part or all of the system.

For example, repeatedly opening sockets will slow down the entire host machine and eventually cause it to stop working.

Container breakouts

A malicious hacker who gains access to a container shouldn't be able to gain access to other containers or the host. By default, the container namespace doesn't include users, so any process that breaks out of the container will have the same privileges on the host as it did in the container. Anyone signed in as a root user in the container will thus have root access on the host. You need to prepare for potential privilege escalation attacks—whereby a user gains elevated privileges, such as those of the root user.

Poisoned images

How do you know that the images you're using have improved safety, haven't been tampered with, and come from where they claim to come from? If a malicious hacker can trick you into running an image, both the host and your data are will be at risk. Similarly, you want to be sure that the images you're running are up to date and don't contain versions of software with known vulnerabilities.

Compromising secrets

When a container accesses a database or service, it will likely require a secret, such as an API key or a username and password. A malicious hacker who can get access to this secret will also have access to the service. This problem becomes more acute in a microservice architecture where containers are constantly stopping and starting as compared to an architecture with small numbers of long-lived VMs.

The following security measures, when implemented well and managed effectively, can help you secure and protect your container ecosystem:

- Use vulnerability management as part of your container development lifecycle.

- Scan for vulnerabilities before pushing images to the registry.
- Continue scanning in the registry.
- Map image vulnerabilities to running containers.
- Ensure that your environment uses only approved images.
- Permit only approved registries.
- Help ensure the integrity of images throughout the lifecycle.
- Enforce the principle of least privilege in the runtime.
- Reduce the container attack surface by removing unneeded privileges.
- Whitelist files and executable files that the container is allowed to access or run.
- Enforce network segmentation on running containers.
- Monitor container activity and user access.
- Monitor container resource activity.
- Log all container administrative user access for auditing.

Summary

Enterprises should keep up with container technology to see how they can take advantage of it. As with all nascent technologies, caveats to be alert for—and best practices and policies to adhere to—exist for a container deployment. The Azure platform has fully managed container services that are integrated with Identity & Access Management capabilities: authentication, authorization, and single sign-on or federation. These services also provide encrypted communications to help secure data in transit. And, they fully support network controls to help lock kernel resources.

A comprehensive security program uses a mixture of Azure and partner solutions to fill in gaps, and it covers much more than just the container runtime environment. Container technology is maturing, but most of what you need to deliver and manage container security is available in the Azure platform today.

For more details about container security in Azure resources, refer to the following Microsoft .NET resources:

- **Management with Azure Container Registry**[119]
- **Management with multiple container images**[120]
- **Manage Azure Container Service with Kubernetes orchestration**[121]

119 https://azure.microsoft.com/en-us/resources/samples/aci-dotnet-create-container-groups-using-private-registry/
120 https://azure.microsoft.com/en-us/resources/samples/aci-dotnet-create-container-groups/
121 https://github.com/Azure-Samples/acs-dotnet-manage-azure-container-service-with-kubernetes-orchestrator

Implement subscription security

Create Azure resource locks

Management locks help you prevent the accidental deletion or modification of your Azure resources. You can manage these locks from within the Azure portal. To view, add, or delete locks, go to the **RESOURCE MANAGEMENT** section of any resource's settings blade.

You can set the lock level to CanNotDelete or ReadOnly (in the portal, the locks are called Delete and Read-only, respectively):

- **CanNotDelete**. This means authorized users can still read and modify a resource, but they can't delete the resource.
- **ReadOnly**. This means authorized users can read a resource, but they can't delete or update the resource. Applying this lock is similar to restricting all authorized users to the permissions granted by the **Reader** role.

When you apply a lock at a parent scope, all resources within that scope inherit the same lock. Even resources you add later inherit the lock from the parent. The most restrictive lock in the inheritance takes precedence.

Unlike with RBAC, you use management locks to apply a restriction across all users and roles.

Exercise

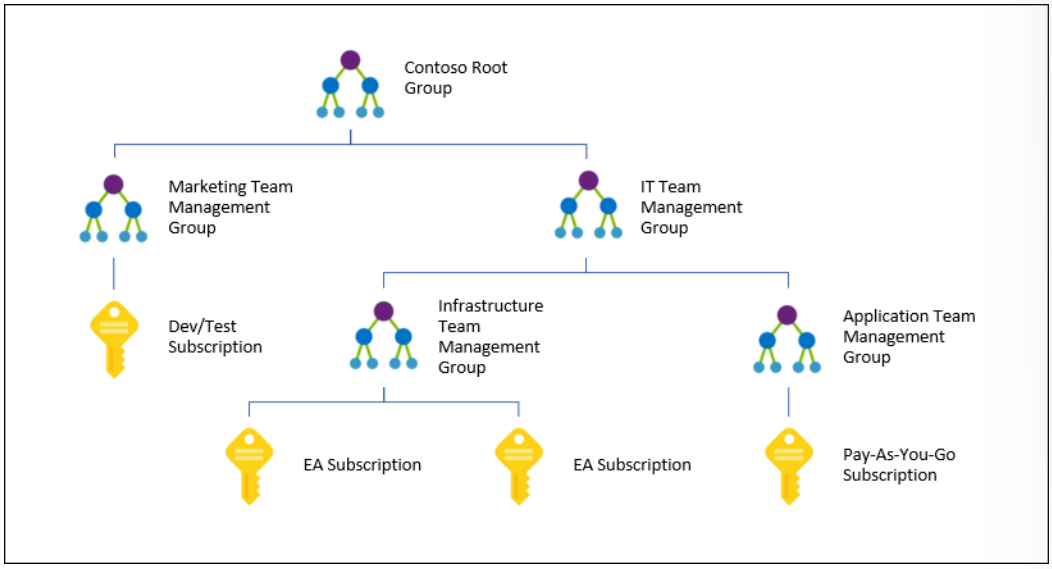
Create locks by using the Azure portal. Go **here**[122].

Configure subscription-level policies in Azure Policy

If your organization has several subscriptions, you might need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups provide a level of scope above that of subscriptions. You organize subscriptions into containers called management groups and apply your governance conditions to the management groups. Management groups enable:

- Organizational alignment for your Azure subscriptions through custom hierarchies and grouping.
- The targeting of policies and spending budgets across subscriptions and inheritance down the hierarchies.
- Compliance and cost reporting by organization (business or teams).

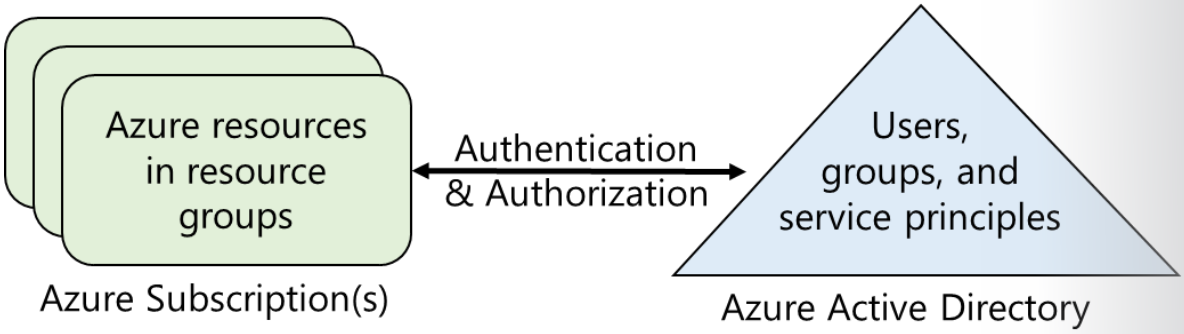
122 https://docs.microsoft.com/en-us/azure/virtual-network/security-overview



All subscriptions within a management group automatically inherit the conditions applied to the management group. For example, you can apply policies to a management group that limit the regions available for VM creation. This policy will apply to all management groups, subscriptions, and resources under that management group by allowing VMs to be created only in those regions.

An Azure subscription is a logical unit of Azure services that is linked to an Azure account. Billing for Azure services is done on a per-subscription basis. If your account is the only account associated with a subscription, you're responsible for the billing.

Subscriptions help you organize access to cloud service resources. They also help you control how resource usage is reported, billed, and paid for. Each subscription can have a different billing and payment setup, so you can have different subscriptions and different plans by department, project, regional office, and so on. Every cloud service belongs to a subscription, and programmatic operations might require the subscription ID.



Azure accounts

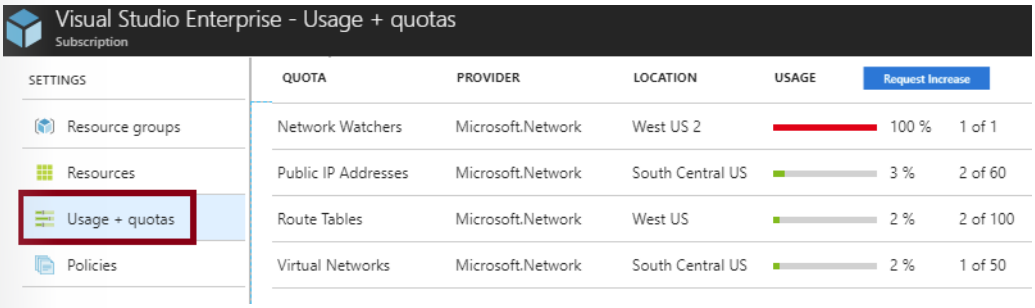
Subscriptions have accounts. An Azure account is simply an identity in either Azure AD a directory that is trusted by Azure AD, such as a work or school organization. If you don't belong to one of these organizations, you can sign up for an Azure account by using your Microsoft account, which is also trusted by Azure AD.

Getting access to resources

Every Azure subscription is associated with Azure AD. Users and services that access the resources of the subscription first need to authenticate with Azure AD.

Typically, to grant a user access to your Azure resources, you add them to the Azure AD directory associated with your subscription. The user then has access to all the resources in your subscription. This is an all-or-nothing operation that might give that user access to more resources than you anticipated.

Azure provides the ability to view the number of resources of each network resource type that you've deployed in your subscription and what your subscription limits are. The ability to view resource usage against limits is helpful for tracking current usage and planning for future use. The following figure has two public IP addresses in the **South Central US** location, and the limit is 60.



The limits displayed are the limits for your subscription. If you need to increase a default limit, select **Request Increase**. You then complete and submit a support request. All resources have a maximum limit listed in Azure **limits**[123]. If your current limit is already at the maximum number, you can't increase it.

 You can also check your resource limits via Azure PowerShell and the Azure CLI. You manage the Azure PowerShell subscriptions by using the **az account** cmdlets. For the list of the cmdlets, got **here**[124].

To manage your security posture by using Azure Security Center, use the **az security** cmdlets **here**[125].

For more information about organizing your resources with Azure management groups, refer to **https://docs.microsoft.com/en-us/azure/azure-resource-manager/management-groups-overview**[126].

For more information about creating management groups for resource organization and management, refer to **https://docs.microsoft.com/en-us/azure/azure-resource-manager/management-groups-create?toc=/azure/billing/TOC.json**[127].

Configure resource-level access policies

Azure Policy is a service in Azure that you use to create, assign, and manage policies. These policies enforce different rules and effects over your resources to help those resources stay compliant with your corporate standards and SLAs. Azure Policy meets this need by evaluating your resources for noncompliance with assigned policies.

For example, you might have a policy to allow only a certain SKU size of VMs in your environment. After this policy is implemented, Azure Policy evaluates new and existing resources for compliance. With the right type of policy, you can bring existing resources into compliance.

If you think that this seems very similar to RBAC, note that a few key differences exist. RBAC focuses on user actions at different scopes. You might be added to the contributor role for a resource group,

123 https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits?toc=%2fazure%2fnetworking%2ftoc.json
124 https://docs.microsoft.com/en-us/azure/azure-resource-manager/management-groups-overview?view=azure-cli-latest
125 https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-forced-tunneling-rm?view=azure-cli-latest
126 https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-web-application-firewall-overview
127 https://docs.microsoft.com/en-us/azure/governance/policy/tutorials/create-and-manage?toc=/azure/billing/TOC.json

allowing you to make changes to that resource group. Azure Policy focuses on resource properties during deployment and for existing resources. Azure Policy controls properties such as the types or locations of resources. Unlike RBAC, Azure Policy is a default allow and explicit deny system.

Azure Policy has several permissions, known as operations, in two resource providers:

- **Microsoft.Authorization**[128]
- **Microsoft.PolicyInsights**[129]

Many built-in roles grant permissions to Azure Policy resources. The **Resource Policy Contributor** role includes most Azure Policy operations. Owner has full rights. Both Contributor and Reader can use all read Azure Policy operations, but Contributor can also trigger remediation.

If none of the built-in roles have the required permissions, create a **custom role**[130].

Azure Policy now provides compliance evaluation for all assignments regardless of pricing tier. If your assignments don't show the compliance data, please ensure that the subscription is registered with the **Microsoft.PolicyInsights** resource provider.

Policy definition

The process of creating and implementing a policy in Azure Policy begins with creating a policy definition. Every policy definition has conditions under which it will be enforced. And, it has a defined effect that takes place if the conditions are met.

Azure Policy offers several built-in policies that are available by default. You can find the default policies **here**[131]. One of the default resource-level access polices is **Allowed Resource Type**, which defines the resource types that you can deploy. Its effect is to deny all resources that don't belong to this defined list.

To implement these policy definitions (both built-in and custom ones), you need to assign them. You can assign any of these policies through the Azure portal, Azure PowerShell, or the Azure CLI.

Policy evaluation happens with several actions, such as policy assignment or policy updates. For a complete list, refer to **Policy evaluation triggers**[132].

To learn more about the structures of policy definitions, review **Policy Definition Structure**[133].

Sample exercises for policy compliance are:

- Assign a policy by using the **Azure portal**[134].
- Assign a policy by using **Azure PowerShell**[135].
- Assign a policy by using the **Azure CLI**[136].

Exercise

Create and manage policies to enforce compliance

Understanding how to create and manage policies in Azure is important for staying compliant with your corporate standards and SLAs. In this tutorial, you'll learn to use Azure Policy to do some of the more common tasks related to creating, assigning, and managing policies across your organization, such as:

- Assigning a policy to enforce a condition for resources you'll create in the future.

128 https://docs.microsoft.com/en-us/azure/governance/policy/concepts/definition-structure
129 https://docs.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations
130 https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles
131 https://docs.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations
132 https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability
133 https://docs.microsoft.com/en-us/azure/virtual-network/manage-ddos-protection
134 https://docs.microsoft.com/en-us/azure/governance/policy/assign-policy-portal
135 https://docs.microsoft.com/en-us/azure/governance/policy/assign-policy-powershell
136 https://docs.microsoft.com/en-us/azure/dns/dns-operations-recordsets-portal

- Creating and assigning an initiative definition to track compliance for multiple resources.
- Resolving a noncompliant or denied resource.
- Implementing a new policy across an organization.

Go **here**[137].

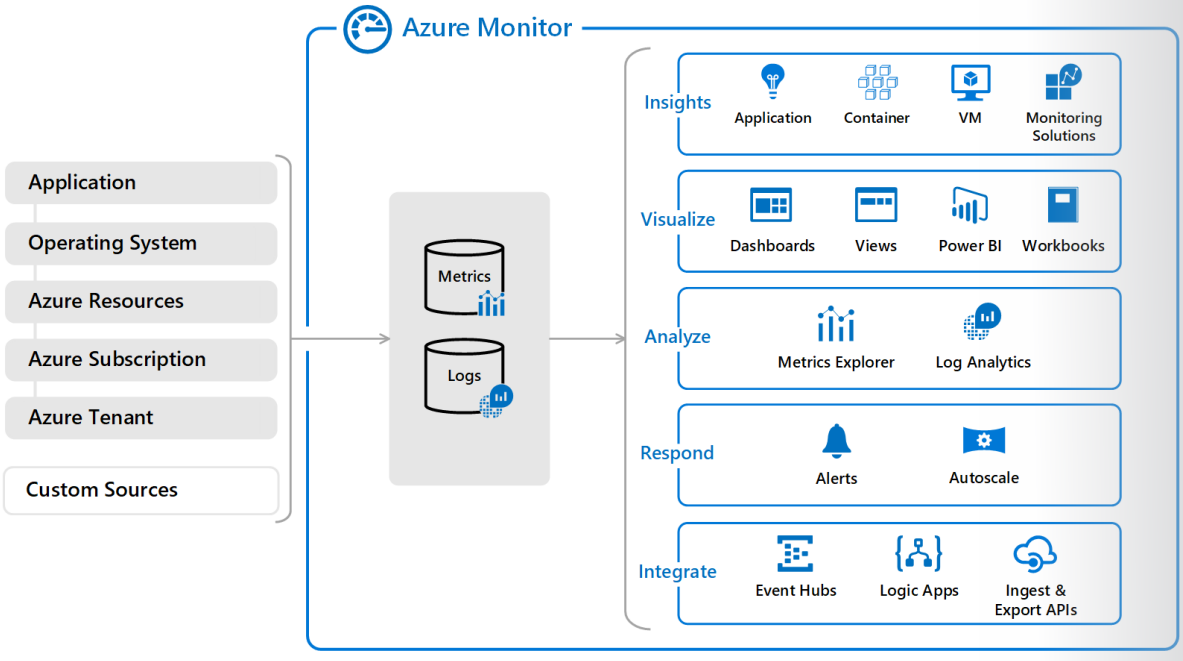
137 https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/add-custom-domain

Module 3 Manage Security Operations

Configure Security Services

Azure Monitor

Earlier, this course discussed Microsoft Azure Monitor. The following high-level diagram depicts the two fundamental data types that Azure Monitor uses.



On the left side of the figure are the sources of monitoring data that populate these data stores. On the right side are the different functions that Azure Monitor performs with this collected data, such as analysis, alerting, and streaming to external systems.

All data that Azure Monitor collects fits into one of two fundamental types: **metrics or logs**[1]. **Metrics**[2] are numerical values that describe aspects of a system at a point in time. They are lightweight and capable of supporting near-real-time scenarios. **Logs**[3] contain different kinds of data organized into records with different sets of properties for each type. Logs store telemetry, such as events and traces, and performance data so that it can all be combined for analysis.

For many Azure resources, you'll find the data that Azure Monitor collects right in the resource's **Overview** page in the Azure portal. Check out any virtual machine (VM). for example, and you'll notice several charts displaying performance metrics. Select any of the graphs to open the data in **Metrics Explorer**[4], which allows you to chart the values of multiple metrics over time. You can view the charts interactively or pin them to a dashboard to view them with other visualizations.

Metrics Explorer

You can analyze log data that Azure Monitor collects by using queries to quickly retrieve, consolidate, and analyze the collected data. You can create and test queries by using **log analytics**[5] in the Azure portal and then either directly analyze the data by using these tools or save queries for use with **visualizations**[6] or **alert rules**[7].

This module will discuss streaming the collected monitor data to external Security Information and Event Management (SIEM) solutions via Azure Security Center. The forwarding or streaming is typically done directly from monitored resources through Azure Event Hubs.

You can get guidance for the different kinds of monitoring data at **Stream Azure monitoring data to an event hub for consumption by an external tool**[8].

Exporting data to a SIEM

Processed events that Azure Security Center produces are published to the **Azure activity log**[9], one of the log types available through Azure Monitor. Azure Monitor offers a consolidated pipeline for routing any of your monitoring data into a SIEM tool. This is done by streaming that data to an event hub, where it can then be pulled into a partner tool.

This pipe uses the Azure Monitor single pipeline for getting access to the monitoring data from your Azure environment. This allows
you to easily set up SIEMs and monitoring tools to consume the data. Currently, the exposed security data from Azure Security Center
to a SIEM consists of **security alerts**[10].

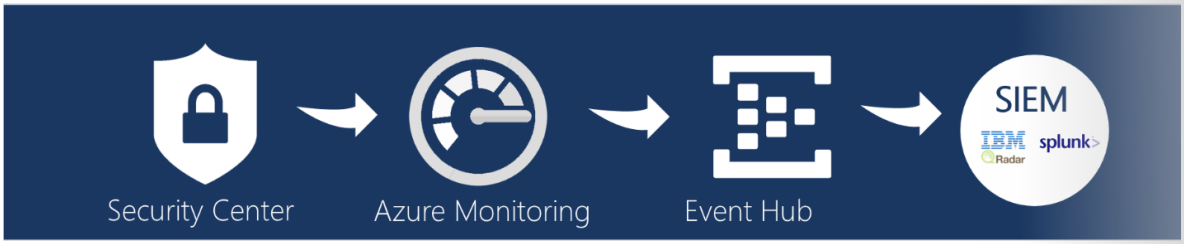
1 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-collection
2 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-collection
3 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-collection
4 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/metrics-charts
5 https://docs.microsoft.com/en-us/azure/azure-monitor/log-query/portals
6 https://docs.microsoft.com/en-us/azure/azure-monitor/visualizations
7 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/alerts-overview
8 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/stream-monitoring-data-event-hubs
9 https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs
10 https://docs.microsoft.com/en-us/azure/security-center/security-center-managing-and-responding-alerts

Azure Security Center security alerts

Security Center automatically collects, analyzes, and integrates log data from your Azure resources; the network; and connected partner solutions, like firewall and endpoint protection solutions, to detect real threats and reduce false positives. Security Center displays a list of prioritized security alerts along with the information you need to quickly investigate the problem and recommendations for how to remediate an attack.

The following sections describe how you can configure data to be streamed to an event hub. The steps assume that you already have Azure Security Center configured in your Azure subscription.

High-level overview



Azure Event Hubs

Azure Event Hubs is a streaming platform and event ingestion service that can transform and store data by using any real-time analytics provider or batching/storage adapters. Use Event Hubs to stream log data from Azure Monitor to a partner SIEM and monitoring tools.

Create an event hub

- First, **create an event hub namespace**[11]. This namespace and event hub will be the destination for all your monitoring data.

Stream the Azure activity log to the event hub

- Refer to the **stream activity log to event hubs**[12] article.
  Note that routing your monitoring data to an event hub via Azure Monitor allows you to easily integrate with a partner SIEM and monitoring tools. Here's the list of **supported SIEMs**[13].

The value of a Security Information and Event Management tool -02

Security Operations (SecOps) teams are inundated with a high volume of alerts and spend far too much time in tasks
like infrastructure set up and maintenance. As a result, many legitimate threats go unnoticed. According to the
"Cybersecurity Jobs Report 2018-2021" by Cybersecurity Ventures, An expected shortfall of 3.5M security professionals
by 2021 will further increase the challenges for security operations teams. Alert fatigue is real. Security analysts
face a huge burden of triage as they not only have to sift through a sea of alerts, but also correlate alerts from
different products manually or using a traditional correlation engine.

11 https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-create
12 https://docs.microsoft.com/en-us/azure/security-center/security-center-partner-integration
13 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/stream-monitoring-data-event-hubs

Microsoft Azure Sentinel

Microsoft Azure Sentinel is a cloud native SIEM tool that is built on Azure. It offers nearly limitless cloud scale
and speed to address your security needs. Think of Azure Sentinel as the first **SIEM-as-a-service** that brings the power
of the cloud and artificial intelligence to help security operations teams efficiently identify and stop cyber-attacks
before they cause harm. Azure Sentinel enriches your investigation and detection by providing both Microsoft's threat
intelligence stream as well as external threat intelligence streams.

Azure Sentinel integrates with Microsoft 365 solution and correlates millions of signals from different products
such as Azure Identity Protection, Microsoft Cloud App Security, and soon Azure Advanced Threat Protection,
Windows Advanced Threat Protection, O365 Advanced Threat Protection, Intune, and Azure Information Protection.

Azure Sentinel provides a single dashboard in which to perform investigation and remediation and provide
automation to move smoothly between detection and remediation. Azure Sentinel will help save you time performing
complex investigation.

Exercise

The Story:

CONTOSO LTD is a global trading company based in the United States but conducting business globally. The
CTO has been under increasing pressure from the board to digitally transform their operations and services
and close capital-intensive data center operations. As such, she directed her team to start using Azure
about three months ago for testing purposes and to deploy some low-risk production workloads. CONTOSO's
security team has not been part of the project and has not been monitoring the workloads for threats.

You are the director of CONTOSO LTD's small information security team. On Friday just before 5 PM, the CTO
calls you and says there might be a problem. "You know that Azure project we've been talking about. Well,
we kicked it off about three months ago. And here's the thing, we're seeing some strange things and are
worried we've been compromised. Sorry for not bringing you in sooner but I need you to look into it ASAP."

You sign-on to your Azure corporate AAD account and see that the CTOs team did configure Azure ATP, and Windows
Defender Advanced Threat Protection. You have been reviewing the preview documentation for Azure Sentinel and
think it would work to consolidate the data from these multiple sources. With the consolidation you can get
a unified look into their deployment.

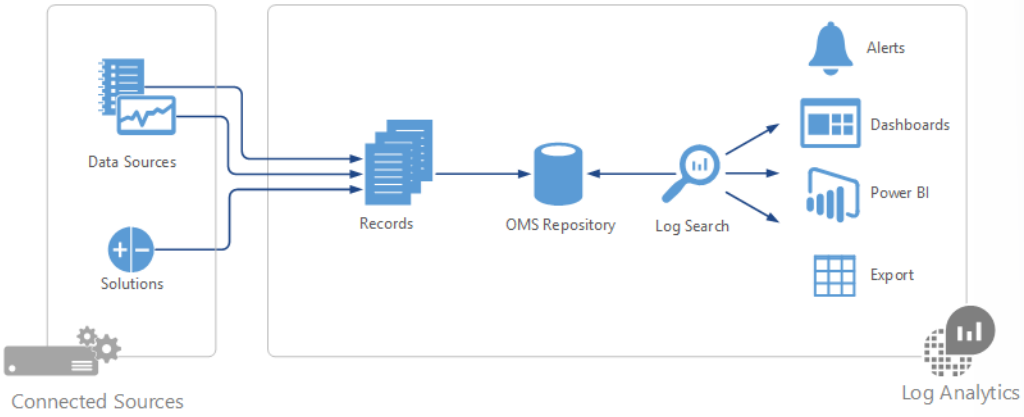
Configure Azure Sentinel

Azure Sentinel is powered by Azure Log Analytics and supports both cloud and on-premise workloads across both Windows and Linux.

To enable Azure Sentinel

1. Active Azure Subscription and sign in to the **Azure portal**[14]
2. Create an Azure Log Analytics workspace

Azure Log Analytics is a service that helps you collect and analyze data that's generated by resources in your cloud
and on-premises environments. It gives you real-time insights by using integrated search and custom dashboards to
readily analyze millions of records across all your workloads and servers regardless of their physical locations.



At the center of Log Analytics is the Log Analytics workspace, which is hosted in Azure. Log Analytics collects data in the workspace from connected sources by configuring data sources and adding solutions to your subscription. Data sources and solutions each create different record types, each with its own set of properties. But you can still analyze sources and solutions together in queries to the workspace. This capability allows you to use the same tools and methods to work with a variety of data collected by a variety of sources.

The connected sources are the computers and other resources that generate the data that Log Analytics collects. The sources can include agents installed on Windows and Linux computers that directly connect or agents that exist in a connected **System Center Operations Manager management group**[15]. Log Analytics can also collect data from an Azure storage account.

The data sources consist of the various kinds of data collected from each connected source. These sources include events and performance data from Windows and Linux agents in addition to sources such as Microsoft Internet Information Services (IIS) logs and custom text logs. You configure each data source that you want to collect, and the configuration is automatically delivered to each connected source.

Four ways exist to **collect logs and metrics for Azure services**[16]:

- From Azure Diagnostics directly to Log Analytics
- From Azure Diagnostics to Azure storage to Log Analytics

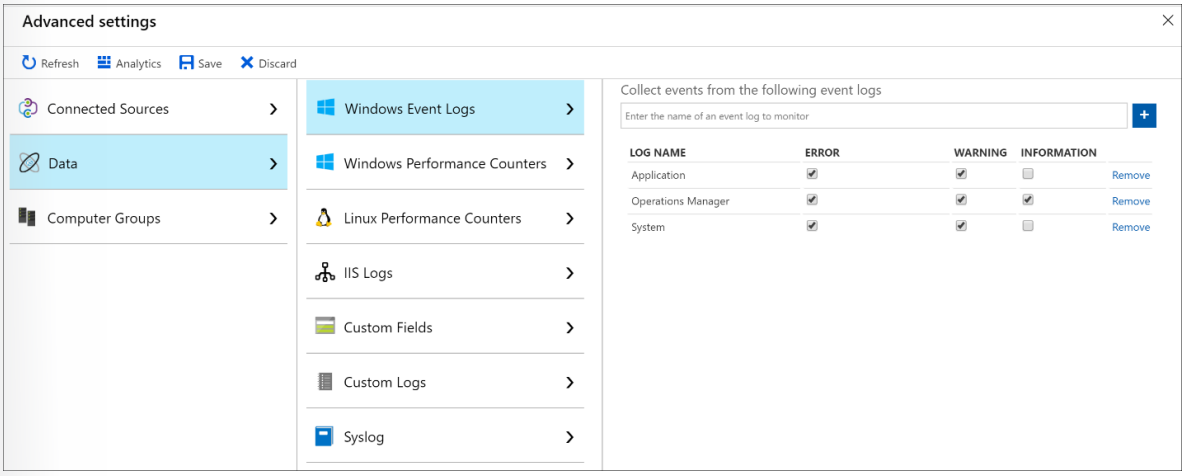
14 https://portal.azure.com/
15 https://docs.microsoft.com/azure/log-analytics/log-analytics-om-agents
16 https://docs.microsoft.com/azure/log-analytics/log-analytics-azure-storage

- Via connectors for Azure services
- Via scripts to collect and then post data into Log Analytics

This **table**[17] provides a summary of the agent data sources that are currently available in Azure Monitor.

Configure data sources

You configure data sources from the Data menu in Advanced settings for the workspace. Any configuration is delivered to all connected sources in your workspace.
Currently, you can't exclude any agents from this configuration. The following figure depicts this Data menu.



To configure data sources:

1.In the Azure portal, click **All services**. In the list of resources, type **Log Analytics**. As you begin typing, the list filters based on your input.
Select **Log Analytics workspaces**.

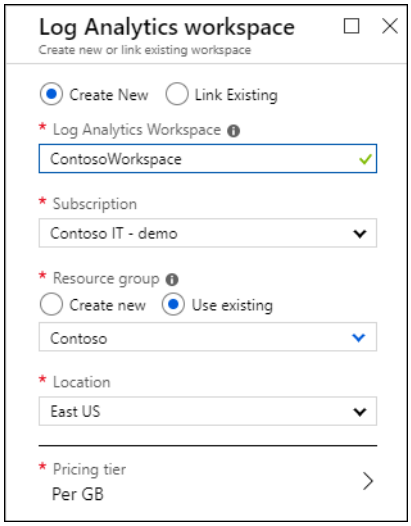
2.Click **Add**, then select choices for the following items:

a. Provide a name for the new **Log Analytics workspace**, such as ContosoWorkspace.

b. Select a **Subscription** to link to by selecting from the drop-down list if the default selected is not appropriate.

c. For **Resource Group**, choose to use an existing resource group already setup or create a new one.

17 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/agent-data-sources



3.After providing the required information on the Log Analytics Workspace pane, click **OK**. While the information is verified
and the workspace is created, you can track its progress under **Notifications** from the menu.

4.To enable Azure Sentinel, you need **contributor permissions**[18] to the subscription in which the Azure Sentinel workspace resides.

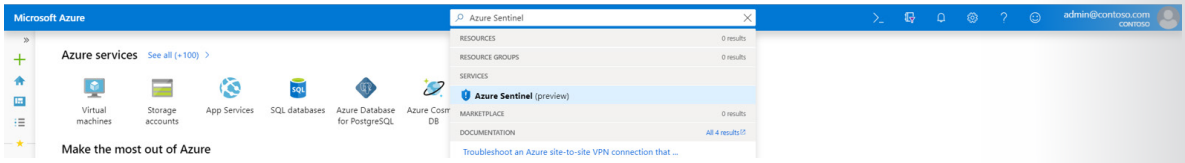
5.To use Azure Sentinel, you need either contributor or viewer permissions on the resource group that the workspace belongs to.

6.Additional permissions may be needed to connect specific data sources.

Enable Azure Sentinel

1.While in the Azure Portal, select the subscription in which Azure Sentinel is to be created. In our story, it would be the CTOs pilot team's subscription.

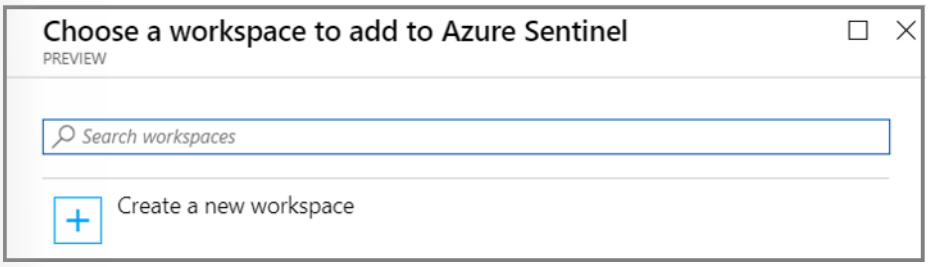
2.Search for Azure Sentinel



3.Click **+Add**

4.Select the workspace you want to use (ContosoWorkspace). You can run Azure Sentinel on more than one workspace, but the data is isolated to a single works.

18 https://docs.microsoft.com/en-us/azure/role-based-access-control/role-assignments-portal



5.Click **Add Azure Sentinel**.

Connect data sources

Azure Sentinel creates the connection to services and apps by connecting to the service and forwarding the events and logs to Azure Sentinel. For machines and virtual machines, you can install the Azure Sentinel agent that collects the logs and forwards them to Azure Sentinel. For Firewalls and proxies, Azure Sentinel utilizes a Linux Syslog server. The agent is installed on it and from which the agent collects the log files and forwards them to Azure Sentinel.

1.Click **Data collection**.

2.There is a tile for each data source you can connect.

For example, click Azure Active Directory. If you connect this data source, you stream all the logs from Azure AD into Azure Sentinel. You can select what type of logs you want to get - sign-in logs and/or audit logs.

At the bottom, Azure Sentinel provides recommendations for which dashboards you should install for each connector so you can immediately get interesting insights across your data.

In our example we would add AAD, ATP. Follow the installation instructions or **refer to the relevant connection guide**[19] for more information.

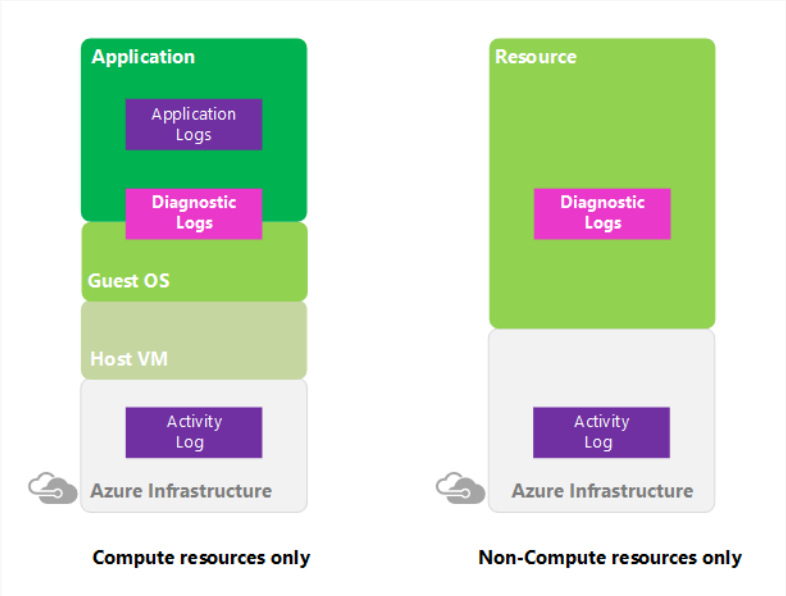
After your data sources are connected, your data starts streaming into Azure Sentinel and is ready for you to start working with. You can view the logs in the built-in dashboards and start building queries in Log Analytics to investigate the data.

Configure diagnostic logging and log retention

Azure Monitor diagnostic logs are logs produced by an Azure service that provide rich, frequently collected data about the operation of that service. Azure Monitor makes two types of diagnostic logs available:

- **Tenant logs**. These logs come from tenant-level services that exist outside an Azure subscription, such as Azure Active Directory (Azure AD).
- **Resource logs**. These logs come from Azure services that deploy resources within an Azure subscription, such as Network Security Groups (NSGs) or storage accounts.

19 https://docs.microsoft.com/en-us/azure/sentinel/connect-data-sources



The content of these logs varies by Azure service and resource type. For example, NSG rule counters and Azure Key Vault audits are two types of diagnostic logs.

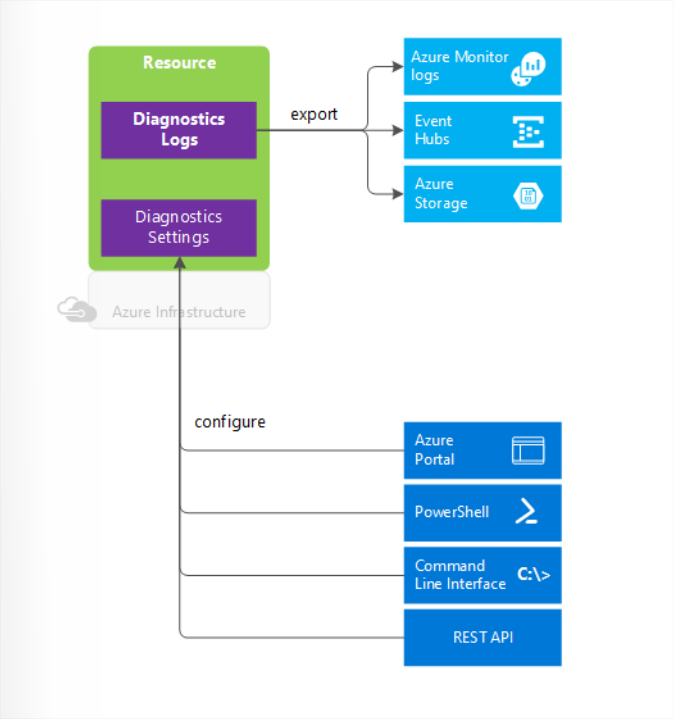
These logs differ from the **activity log**[20]. The activity log provides insight into the operations, such as creating a VM or deleting a logic app, that Azure Resource Manager performed on resources in your subscription using. The activity log is a subscription-level log. Resource-level diagnostic logs provide insight into operations that were performed within that resource itself, such as getting a secret from a key vault.

These logs also differ from guest operating system (OS)–level diagnostic logs. Guest OS diagnostic logs are those collected by an agent running inside a VM or other supported resource type. Resource-level diagnostic logs require no agent and capture resource-specific data from the Azure platform itself, whereas guest OS–level diagnostic logs capture data from the OS and applications running on a VM.

Uses for diagnostic logs

Here are some of the things you can do with diagnostic logs:

20 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/activity-logs-overview



- Save them to a **storage account**[21] for auditing or manual inspection. You can specify the retention time (in days) by using resource diagnostic settings.
- **Stream them to event hubs**[22] for ingestion by a third-party service or custom analytics solution, such as Power BI.
- Analyze them with **Azure Monitor**[23], such that the data is immediately written to Azure Monitor with no need to first write the data to storage.

Exercise

To enable the collection of diagnostic logs, go **here**[24].

Configure vulnerability scanning and policies

Azure Security Center provides you with a centralized view of your Azure resources and their active security state. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions. These solutions are delivered through the following capabilities:

- Prevention. Security Center monitors the state of your Azure resources based on how you configure the security policies for your organization, applications, and data.
- Detection. Security Center automatically collects and analyzes security data from your Azure resources; the network; and partner solutions, like antimalware and firewalls. It takes advantage of global threat intelligence from Microsoft products and services, the Microsoft Digital Crimes Unit, the Microsoft Security Response Center, and external feeds.

21 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/archive-diagnostic-logs
22 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/diagnostic-logs-stream-event-hubs
23 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/collect-azure-metrics-logs
24 https://docs.microsoft.com/en-us/azure/azure-monitor/platform/diagnostic-logs-overview#how-to-enable-collection-of-diagnostic-logs

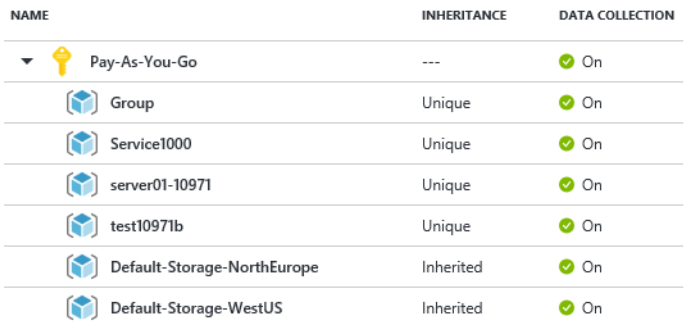
- Response. Security Center provides you with prioritized security and incident alerts. It offers insights into the source of an attack and any impacted resources along with suggestions for how to stop the current attack and help prevent future attacks.

Security policies

A security policy defines the set of controls that are recommended for resources within the specified subscription or resource group. In Security Center, you define policies for your Azure subscriptions or resource groups according to your company's security needs and the types of applications or sensitivity of data in each subscription.

For example, resources used for development or testing might have different security requirements than resources used for production applications. Likewise, applications that use regulated data, like personally identifiable information, might require a higher level of security. Security policies that are enabled in Azure Security Center drive security recommendations and monitoring to help you identify potential vulnerabilities and mitigate threats.

When you enable Security Center and data collection, all the security policies are enabled by default. The policies are inherited from the subscription down to the resource groups. However, you can control the security policies individually at the resource group level. In the following figure, some of the resource groups have inheritance turned on and some are set to be unique (which means that their security policy settings might differ from those of the subscription).

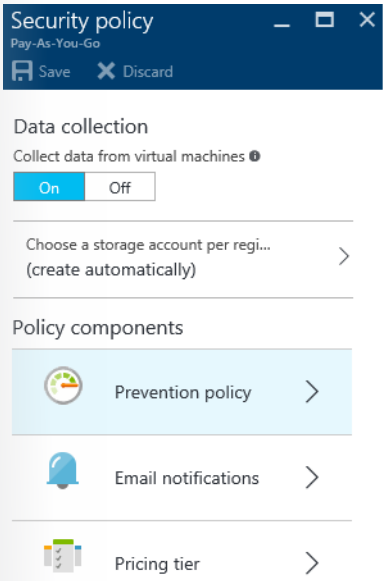


NAME	INHERITANCE	DATA COLLECTION
Pay-As-You-Go	---	On
Group	Unique	On
Service1000	Unique	On
server01-10971	Unique	On
test10971b	Unique	On
Default-Storage-NorthEurope	Inherited	On
Default-Storage-WestUS	Inherited	On

To modify a security policy at the subscription level or resource group level, you need to be an Owner or Contributor for that subscription.

At the subscription level, you have two configuration items:

- **Data collection**. Set this to On or Off for the VMs in the subscription. Data collection includes the daily scanning of all supported VMs for security monitoring and recommendations. It also includes the collection of security events for analysis and threat detection.
- **Storage account**. Set this to where to store the security data. If you don't choose a storage account for each region, one will be created for you. For security reasons, the data that's collected is logically isolated from other customers' data.



Additionally, three policy components exist:

- **Prevention policy**. Recommendations for different features, such as system updates and disk encryption.
- **Email notifications**. Security contact details and email notifications for high security alerts.
- **Pricing tier**. Information about the available pricing tiers.

The vulnerability assessment in Azure Security Center is part of the Security Center VM recommendations. If Security Center doesn't find a vulnerability assessment solution installed on your VM, it recommends that you install one. A partner agent, after being deployed, starts reporting vulnerability data to the partner's management platform. In turn, the partner's management platform provides vulnerability and health monitoring data back to Security Center. You can quickly identify vulnerable VMs on the Security Center dashboard. Switch to the partner management console directly from Security Center for additional reports and information.

Here are some important notes regarding the vulnerability assessment capability:

- Currently, vulnerability assessments are available from Qualys and Rapid7.
- You can install the vulnerability assessment solution on multiple VMs. The VMs need to belong to the same subscription.

Exercise

Implement a vulnerability assessment recommendation.

When Security Center identifies one or more supported VMs that are missing a vulnerability assessment solution, it triggers a VM recommendation.
This exercise shows how to access and apply this recommendation for multiple VMs that exist in the same subscription.
The exercise is **here**[25].

25 https://docs.microsoft.com/en-us/azure/security-center/security-center-vulnerability-assessment-recommendations

Configure security policies using Azure Security Center

Configure endpoint protection

Focusing just on the endpoint recommendation, what does Azure Security Center report as issues?

By using the information under **Endpoint protection issues**, you can identify a plan to address any issues identified.

Security Center reports the following endpoint protection issues:

- **Endpoint protection not installed on Azure VMs**. A supported antimalware solution isn't installed on these Azure VMs.
- **Endpoint protection not installed on non-Azure computers**. A supported antimalware solution isn't installed on these non-Azure computers.
- Endpoint protection health issues:
  - **Signature out of date**. An antimalware solution is installed on these VMs and computers, but the solution doesn't have the latest antimalware signatures.
  - **No real time protection**. An antimalware solution is installed on these VMs and computers, but it isn't configured for real-time protection. The service might be disabled, or Security Center might be unable to obtain the status because the solution isn't supported.
  - **Not reporting**. An antimalware solution is installed but not reporting data.
  - **Unknown**. An antimalware solution is installed, but either its status is unknown or it's reporting an unknown error.

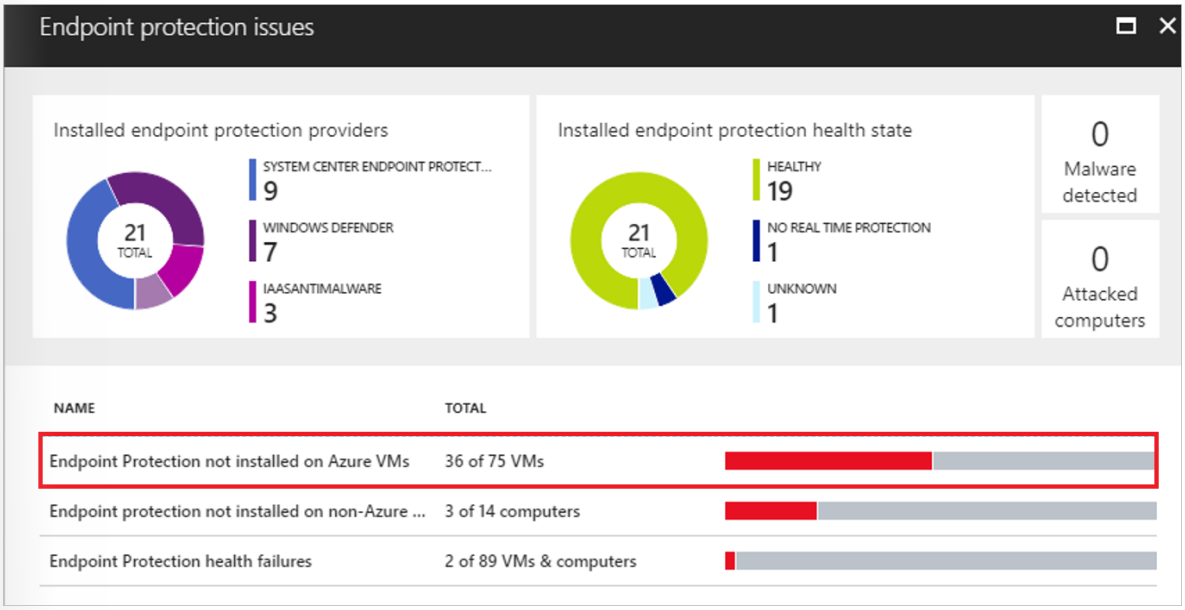
Implement the recommendation

Security Center presents the endpoint protection issues as a recommendation. If your environment is determined to be vulnerable to antimalware threats, this recommendation will be displayed under **Recommendations** and under **Compute**. To see the **Endpoint protection** issues dashboard, follow the **Compute** workflow.

The following example uses Compute and explains how to install antimalware on Azure VMs.

Install antimalware on Azure VMs

1. Under the Security Center main menu or **Overview**, select **Compute**.
2. Under **Compute**, select **Endpoint protection issues**. The **Endpoint protection issues** dashboard opens.



1. Note that the top of the dashboard displays:
   - **Installed endpoint protection providers**. Lists the different providers that Security Center identified.
   - **Installed endpoint protection health state**. Displays the health state of the VMs and computers that have an endpoint protection solution installed. The chart shows the number of VMs and computers that are healthy and the number considered to have insufficient protection.
   - **Malware detected**. Displays the number of VMs and computers where Security Center is reporting detected malware.
   - **Attacked computers**. Displays the number of VMs and computers where Security Center is reporting attacks by malware.

The bottom of the dashboard displays a list of endpoint protection issues that includes:

```
* The total number of VMs and computers impacted by the issue.

* A bar aggregating the number of VMs and computers impacted by the issue.
The color of the bar identifies the priority:

- Red means high priority, so the issue should be addressed immediately.

- Orange means medium priority, so the issue should be addressed as soon as
possible.
```

1. Select **Endpoint protection not installed on Azure VMs**.
2. Note that under **Endpoint protection not installed on Azure VMs** is a list of Azure VMs that don't have antimalware installed. You can choose to install antimalware on all VMs in the list or select individual VMs to install antimalware on by selecting the specific VM.
3. Under **Select Endpoint protection**, select the endpoint protection solution you want to use. In this example, select **Microsoft Antimalware**.

4. Note that additional information about the endpoint protection solution is displayed. Select **Create**.

Exercise

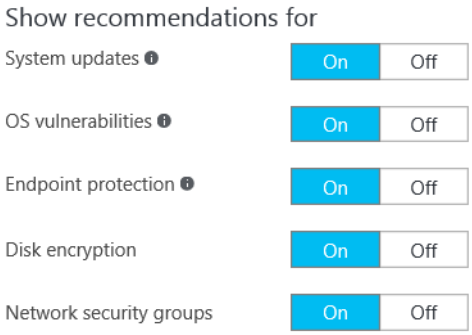
Install antimalware on non-Azure computers.

For instructions, go **here**[26]. Suggestion: try it without looking at the instructions.

Configure centralized policy management by using Azure Security Center

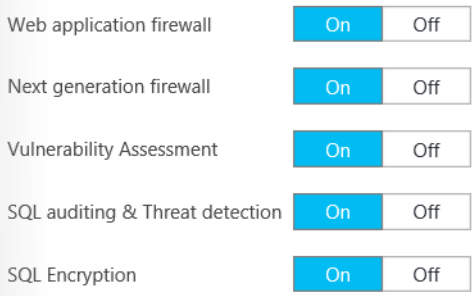
By default, all prevention policies are turned on. Prevention policies and recommendations are tied to each other. In other words, if you enable a prevention policy, such as OS vulnerabilities, that enables recommendations for that policy. In most situations, you want to enable all policies even though some might be more important to you than others, depending on the Azure resource you've deployed.

The following is a generated list of the types of recommendations. The recommendations help provide full visibility into the security health of your environment.



- **System updates**. Retrieves a daily list of available security updates and critical updates from Windows Update or Windows Server Update Services (WSUS).
- **OS vulnerabilities**. Analyzes OS configurations daily to determine issues that might make the VM vulnerable to attack.
- **Endpoint protection**. Recommends endpoint protection to be provisioned for all Windows VMs to help identify and remove viruses, spyware, and other malicious software.
- **Disk encryption**. Recommends enabling disk encryption in all VMs to enhance data protection at rest.
- **Network security groups**. Recommends that NSGs be configured to control inbound and outbound traffic to VMs that have public endpoints. In addition to checking that an NSG has been configured, this policy assesses inbound security rules.

26 https://docs.microsoft.com/en-us/azure/security-center/security-center-install-endpoint-protection



- **Web application firewall**. Extends network protections beyond NSGs, which are built into Azure. Security Center will discover deployments for which a next generation firewall is recommended and allow you to provision a virtual appliance.
- **Next Generation firewall**. Azure Security Center may recommend that you add a partner's next generation firewall (NGFW) from a Microsoft partner to increase your security protections.
- **Vulnerability Assessment**. Recommends that you install a vulnerability assessment solution on your VM.
- **SQL auditing & Threat detection**. Recommends that you enable the auditing of access to Azure SQL Database for compliance and advanced threat detection—for investigation purposes.
- **SQL Encryption**. Recommends that you enable encryption at rest for your Azure SQL database, associated backups, and transaction log files. This helps prevent your data from being readable even if it's breached.

Configure just-in-time VM access by using Azure Security Center

Just-in-time (JIT) virtual machine (VM) access can be used to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy access to connect to VMs when needed.

Typical attack

Brute force attacks commonly target management ports as a means to gain access to a VM. If successful, an attacker can take control over the VM and establish a foothold into your environment. A brute-force attack consists of checking all possible usernames or passwords until the correct one is found. (This isn't the most sophisticated form of attack, but it's relatively simple to perform.)

One way to reduce exposure to a brute force attack is to limit the amount of time that a port is open. Management ports do not need to be open at all times. They only need to be open while you are connected to the VM, for example to perform management or maintenance tasks. When just-in-time is enabled, Security Center uses network security group (NSG) rules, which restrict access to management ports so they cannot be targeted by attackers.

Blunting RDP brute-force attacks

To blunt RDP brute-force attacks, you can take multiple measures, such as:

- Disabling the public IP address and using one of these connection methods:
  - Point-to-site virtual private network (VPN)
  - Site-to-site VPN
  - Azure ExpressRoute

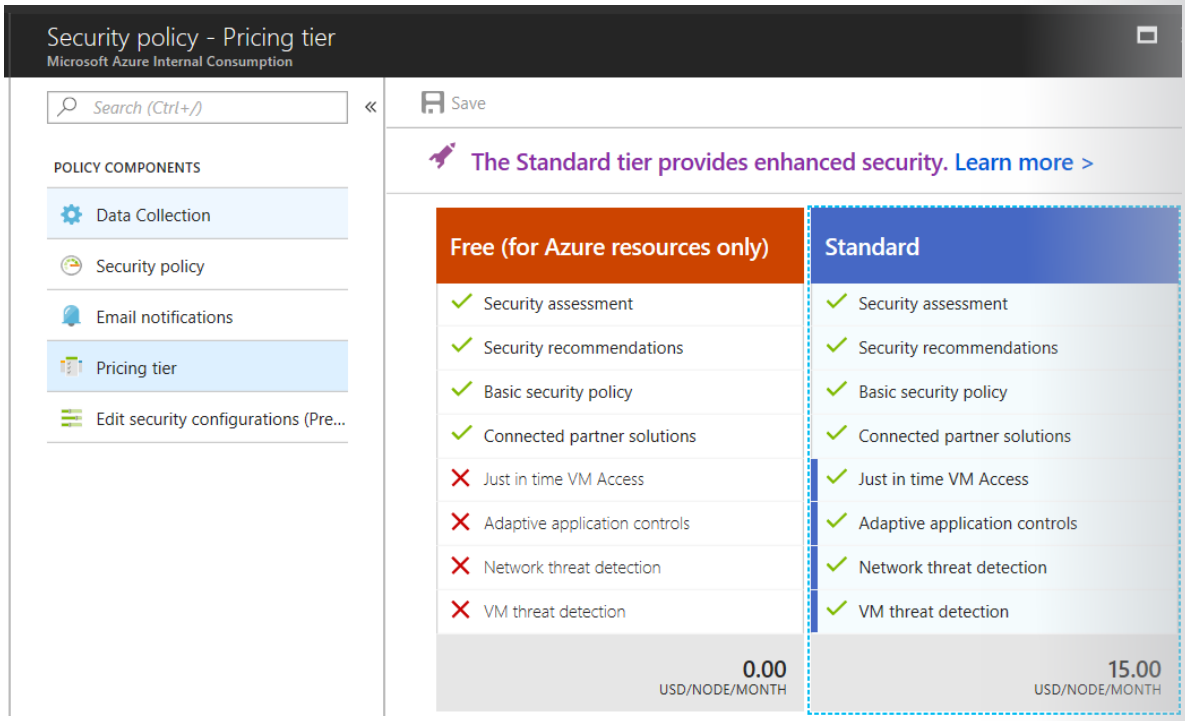
- Requiring two-factor authentication
- Using complex passwords
- Limiting the amount of time that the ports are open

Azure Security Center implements the last method by using just-in-time (JIT) VM Access, which allows you to help secure remote access to one or more VMs. JIT VM Access reduces the exposure time of privileges and increases your visibility into the use of privileged accounts.

When you enable JIT VM Access for your VMs, you can create a policy that determines the ports to help protect, how long ports should remain open, and the approved IP addresses that can access these ports. The policy helps you stay in control of what users can do when they request access. Requests are logged in the Azure activity log, so you can easily monitor and audit access. The policy will also help you quickly identify the existing VMs that have JIT VM Access enabled and the VMs where JIT VM Access is recommended.

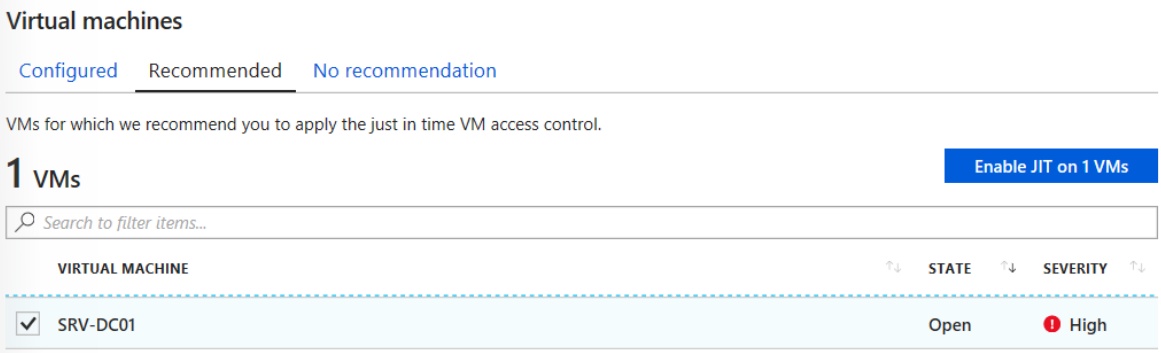
How JIT VM Access works

1. Note that you need to be in the Standard pricing tier of Azure Security Center.



1. Enable JIT VM Access for the selected Azure VMs.[27]

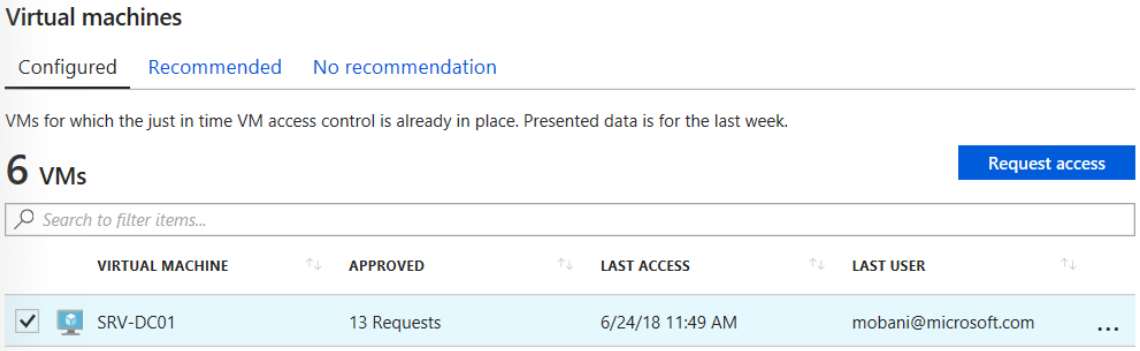
27 https://msdnshared.blob.core.windows.net/media/2018/06/RDP02.png



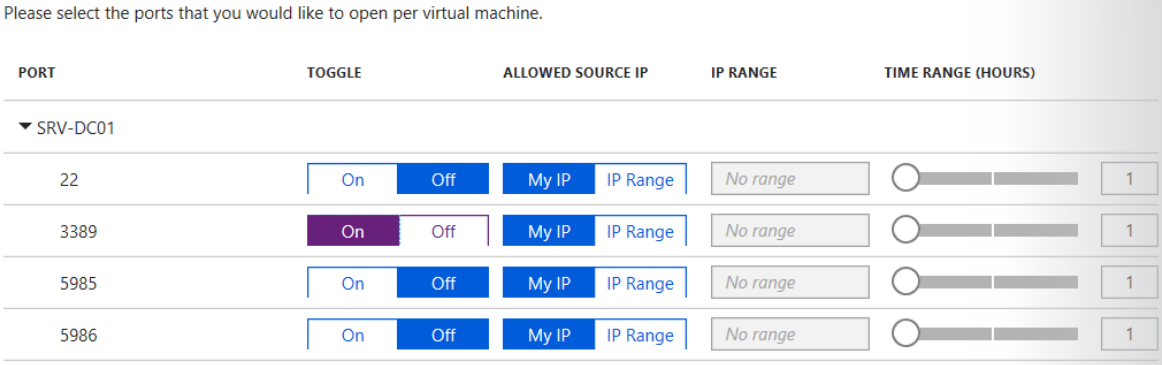
1. Note that at this point, the NSG of the VMs is updated with rules to block remote management access.

1001	SecurityCenter-JITRule_-1055335910...	3389	Any	Any	10.1.0.4	Deny
1002	SecurityCenter-JITRule_-1055335910...	5985	Any	Any	10.1.0.4	Deny
1003	SecurityCenter-JITRule_-1055335910...	5986	Any	Any	10.1.0.4	Deny
1004	SecurityCenter-JITRule_-1135386826...	22	Any	Any	10.1.0.6	Deny

1. Use Azure Security Center to request access to a VM by using one of the protocols in the policy.



1. Note that the required NSG is updated to allow inbound access for that protocol.



1. Note that Admins and developers can remotely get in to the VM.
2. Note that Security Center will remove the allow rule in the NSG after the predetermined amount of time specified in the policy.
3. Note that to send such a request, the user who requests access to the VM needs to have write access to the VMs in Azure role-based access control (RBAC).
4. Note that all access requests are logged and can be reviewed in the activity log.

Manage security alerts

Create and customize alerts

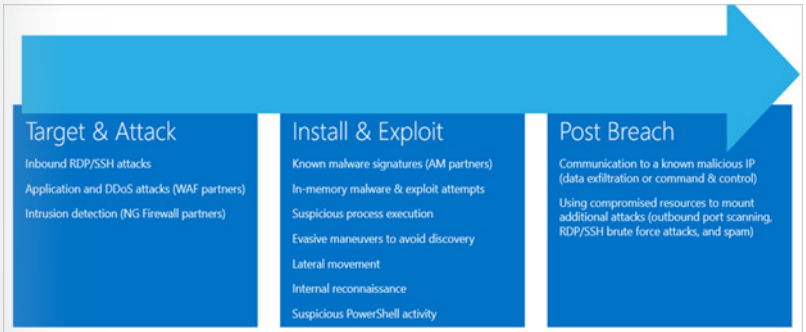
Azure Security Center automatically collects, analyzes, and fuses log data from your Azure resources; the network; and partner solutions, like antimalware and firewalls. When threats are detected, a security alert is created. Examples include the detection of:

- Compromised VMs communicating with known malicious IP addresses.
- Advanced malware detected by Windows error reporting.
- Brute-force attacks against VMs.
- Security alerts from integrated partner security solutions, such as antimalware or web application firewalls.

Both Security Center and partner solutions can generate security alerts. Security Center logs individual security alerts. But it also uses big data and machine learning technologies to combine individual alerts into incidents. An incident is a collection of related individual alerts. Note that this combining of related alerts into incidents is an advanced capability of Security Center and requires at least the Standard tier for Azure Security Center.

Alert types

Azure Security Center provides a variety of alerts that align with the stages of the kill chain. The kill chain consists of three phases: target and attack, install and exploit, and post breach. Different types of attacks are associated with each stage, and they target different subsystems. For example, the target and attack phase include inbound RDP/SSH attacks and intrusion detection.



The alerts contain valuable information about what triggered the alert, the resources targeted, and the source of the attack. The information included in an alert varies based on the type of analytics used to detect the threat. Incidents might also contain additional contextual information that might be useful during the investigation of a threat. To address attacks during these stages, Security Center has categories of alerts:

- Virtual machine behavioral analysis
- Network analysis
- SQL database and SQL Data Warehouse analysis
- Contextual information

For a detailed list of the predefined alerts in Security Center, go **here**[28]. These alerts are triggered when either a threat or suspicious activity takes place. For some scenarios, you might want to create a custom alert to address the specific needs of your environment.

Custom alert rules in Security Center allow you to define new security alerts based on data that's already been collected from your environment. You can create queries and use the results of these queries as criteria for the custom rule. When the criteria are matched, the rule runs. You can use computers security events, a partner's security solution logs, or data ingested via APIs to create your custom queries.

Exercise

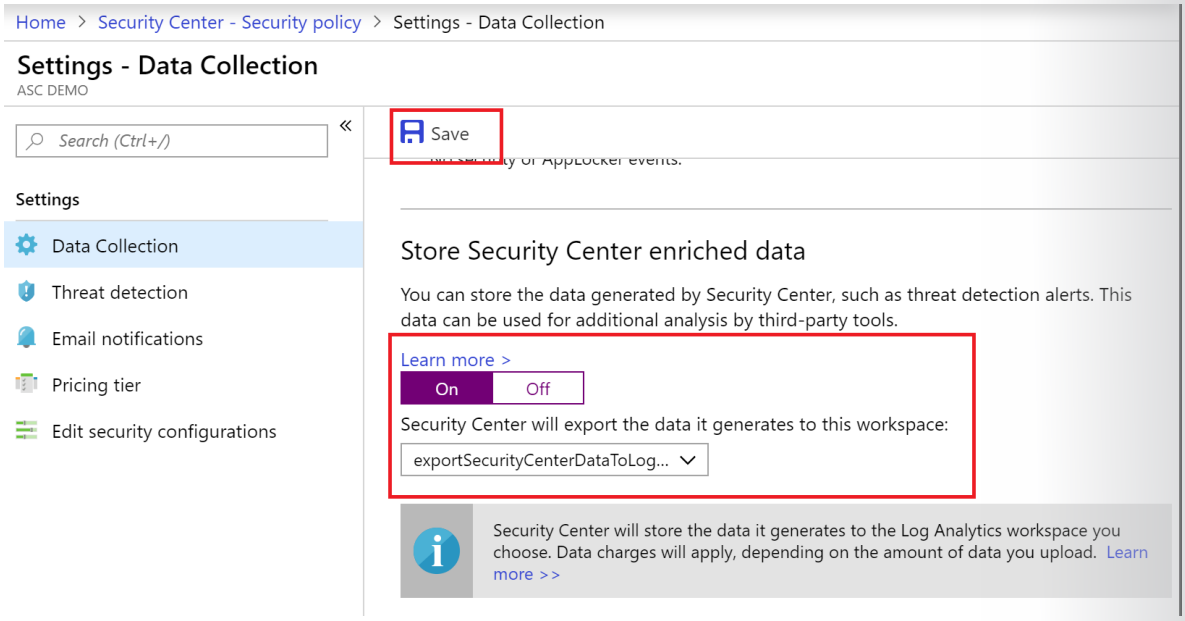
Create a custom alert rule in Security Center. Note that you need write permission in the workspace that you select to store your custom alert.

For the exercise instructions, go **here**[29].

If you need to export the alerts generated by Security Center to a location that other tools can use, the Export to Log Analytics feature provides the solution. For this process to work, you need a Log Analytics workspace. We recommend that you use the same Log Analytics workspace that you defined in the data collection settings when defining one for collecting data.

Export the alerts

1. Select **Security Policy**.
2. For the subscription you want to use, select **Edit Settings**.
3. Select **Data Collection**.
4. Scroll down to **Store Security Center enriched data**, and then select **On**.



1. Select the workspace you want to export the alerts to, and then select **Save**.

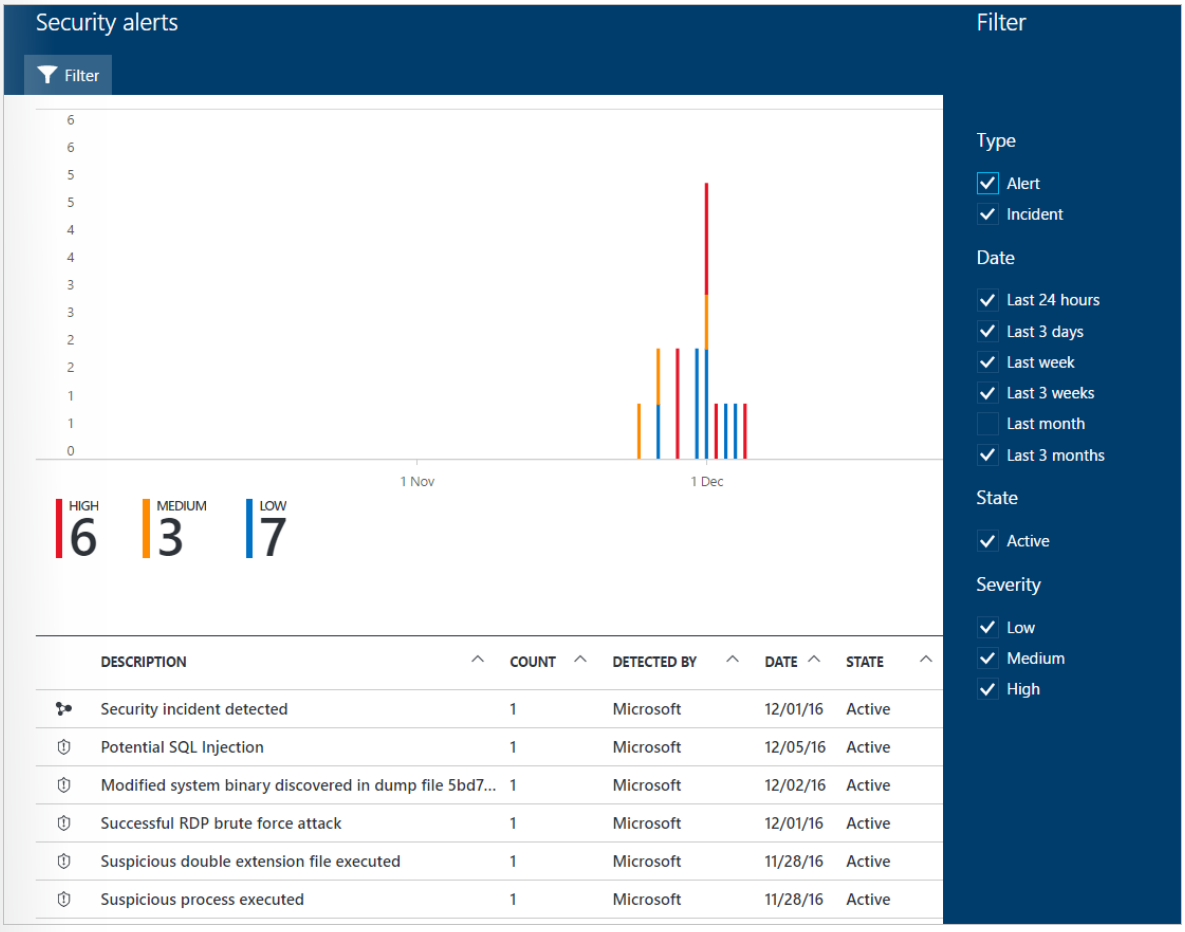
28 https://docs.microsoft.com/azure/security-center/security-center-managing-and-responding-alerts
29 https://docs.microsoft.com/en-us/azure/security-center/security-center-custom-alert

Managing security alerts

In the Azure portal, on the **Overview** page for Security Center, an at-a-glance view of your environment is displayed. The **Detection** area of the **Overview** page displays a graph of your current alerts, colored by the severity level (high, medium, or low).

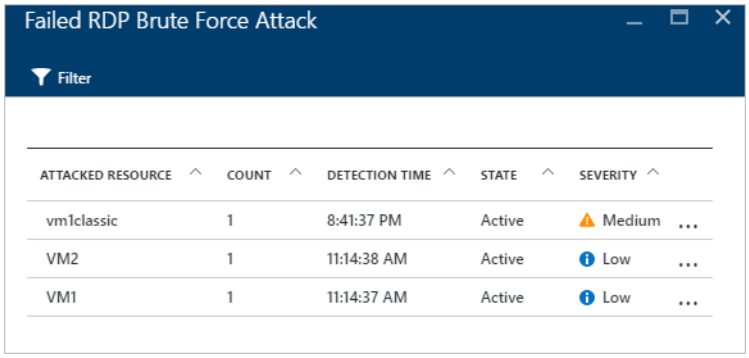
The bottom part of the blade has the details for each alert. To sort them, select the column you want to sort by. You can filter alerts based on the date, state, and severity. Filtering alerts might be useful for scenarios where you need to narrow the scope of the security alerts. For example, you might you want to address security alerts that occurred in the last 24 hours if you're investigating a potential breach in the system.

Select **Filter** on the **Security alerts** page. The **Filter** area opens, and you select the date, state, and severity values you want to see.

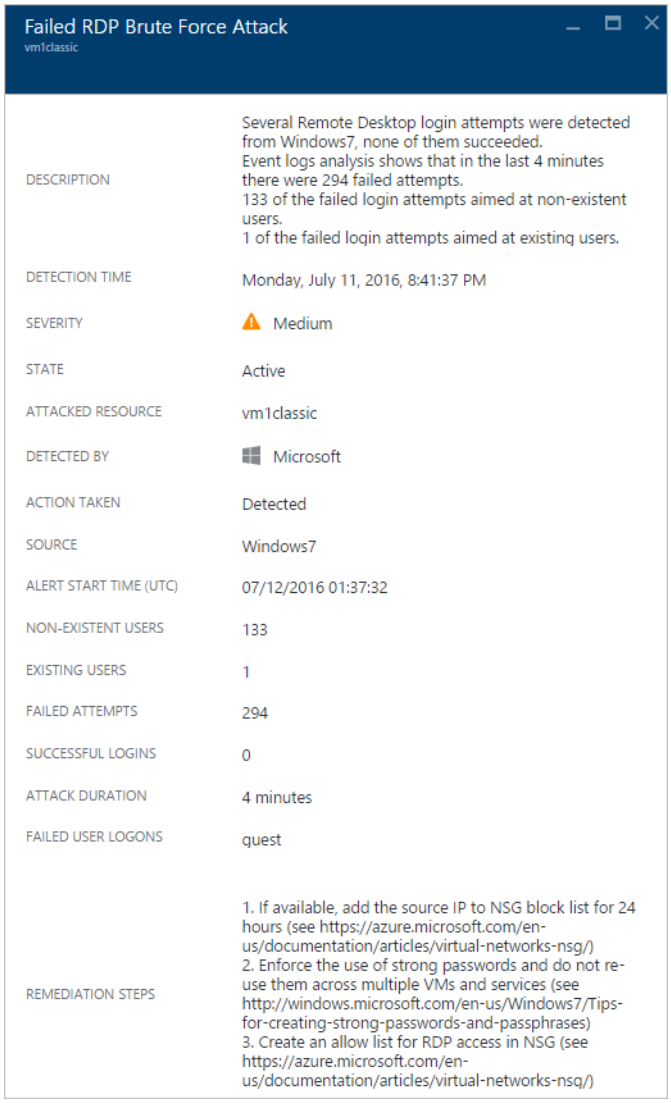


Working with alerts

Select a security alert to learn more about the events that triggered it and what steps, if any, you need to take to remediate an attack. Security alerts are grouped by type and date. Selecting a security alert opens a blade containing a list of the grouped alerts, as the following figure depicts.



In this case, the alerts that were triggered refer to suspicious RDP activity. The first column lists which resources were attacked, the second displays how many times the resource was attacked, the third displays the time of the attack, the fourth displays the state of the alert, and the fifth displays the severity of the attack. After reviewing this information, select a resource that was attacked.



The **DESCRIPTION** area has more details about this event. These details offer insight into what triggered the security alert, the target resource, the source IP address (when applicable), and recommendations

about how to remediate the event. In some cases, the source IP address is empty (not available), because not all Windows security event logs include the IP address.

The remediation steps suggested by Security Center vary according to the security alert. In some cases, you might have to use other Azure capabilities to implement the recommended remediation. For example, the remediation for this attack is to blacklist the IP address that generated this attack by using a network access control list (ACL) or an NSG rule.

From this page, you can also start an investigation to better understand the timeline of the attack, how the attack took place, which systems were potentially compromised, and which credentials were used, and you can get a graphical representation of the entire attack chain.

After you identify the compromised system, you can run security playbooks (which the next lesson covers) that were previously created. A security playbook is a collection of procedures that can be executed from Security Center after a certain playbook is triggered from a selected alert.

Configure a playbook for a security event by using Azure Security Center

A security playbook can help automate and orchestrate your response to a specific security alert detected by Security Center. Security playbooks in Security Center are based on **Azure Logic Apps**[30], which means that you can use the templates from the security category of the Logic Apps templates, you can modify them based on your needs, or you can create new playbooks by using an **Azure Logic Apps workflow**[31] and using Security Center as your trigger.

Exercise

The Security Center team has set up a GitHub repository with instructions on how to create a security playbook and how to run the newly created playbook by using Security Center.

The GitHub location is **here**[32]. The lab will take about half an hour to complete.

In Security Center, you can add an action or conditions to an existing playbook. on the **Playbooks** tab, select the name of the playbook you want to change. The Logic Apps Designer opens.

Note: For more information about how to create your own playbook by using Azure Logic Apps, refer to **Quickstart: Create your first automated workflow with Azure Logic Apps - Azure portal**[33].

30 https://docs.microsoft.com/azure/logic-apps/logic-apps-what-are-logic-apps
31 https://docs.microsoft.com/azure/logic-apps/logic-apps-create-a-logic-app
32 https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/security-center/security-center-playbooks.md
33 https://docs.microsoft.com/azure/logic-apps/logic-apps-create-a-logic-app

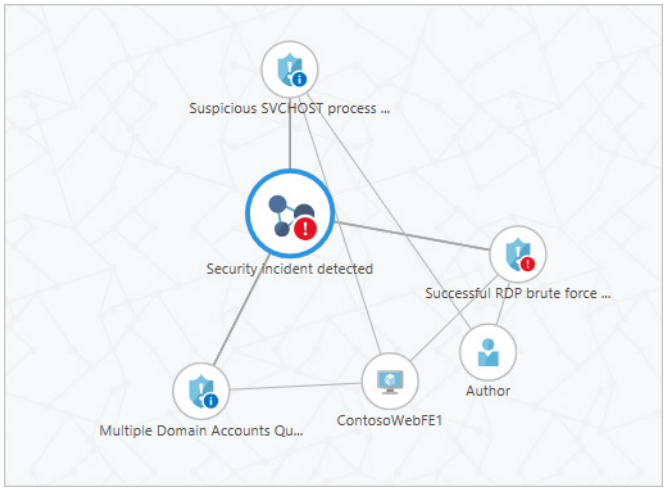
Respond to and remediate security issue

Investigate escalated security incidents

The investigation feature in Security Center allows you to triage a potential security incident, understand its scope, and track down its root cause.

The intent is to facilitate the investigation process by linking all entities—security alerts (custom alerts aren't supported), users, computers (currently just those running Windows Server), and incidents—that are involved with the incident you're investigating. Security Center does this by correlating relevant data with any involved entities and exposing this correlation in a live graph that helps you navigate through the objects and visualize or conceptualize the relevant information.

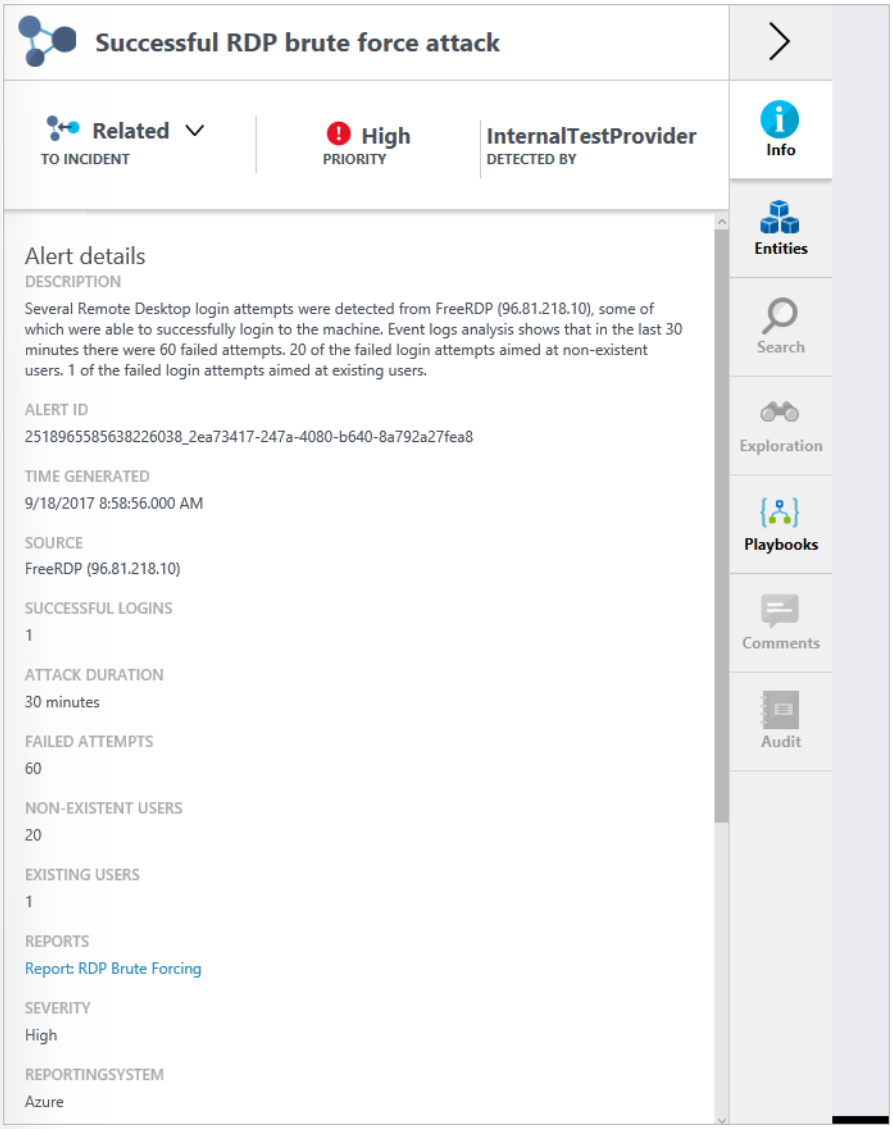
A graph occupies the central area of the investigation dashboard. The graph always focuses on a specific entity and presents the entities that are related to it. An entity can be a security alert, a user, a computer, or an incident.



You can navigate from one entity to another by selecting the entity you want on the graph. The graph automatically centralizes the selected entity and its related entities. Entities that are no longer relevant might be removed from the graph.

Info tab

When the graph presents an entity, the tabs show additional information about that entity. The Info tab presents general information about the entity from various available information sources.

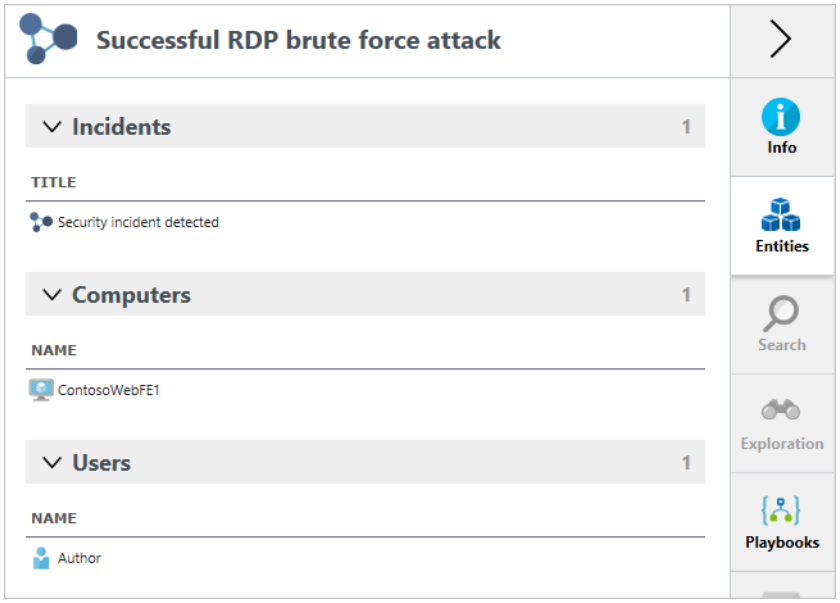


The **Info tab** displays information relevant to the incident selected in the map. An incident in this context is a container that includes the results of an investigation. Every investigation happens in the context of an incident.

An incident is created only when you select **Start investigation** for a specific alert. The basic capability available is to mark entities, such as users, computers, and alerts. When an entity is marked as related, a reason is provided. From that point onward, the entity appears directly under the incident on the graph and in the **incident entities** list.

Entities tab

The **Entities tab** displays all the related entities grouped by type. It is useful in two cases: when there too many entities exist to present on the graph and when the entities' names are too long, and it is easier to examine them in a tabular way.

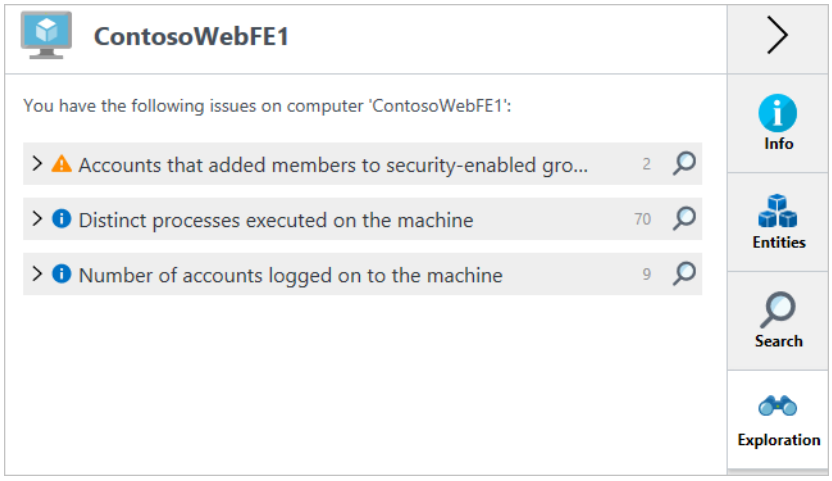


Search tab

The **Search tab** presents all the log types that are available for the entity. For each log type, the tab displays how many records are available. Selecting a log type takes you to the search screen. On the search screen, you can refine your search and use the various search features, such as setting alerts. In the current release, the **Search tab** is available only for user and computer entities.

Exploration tab

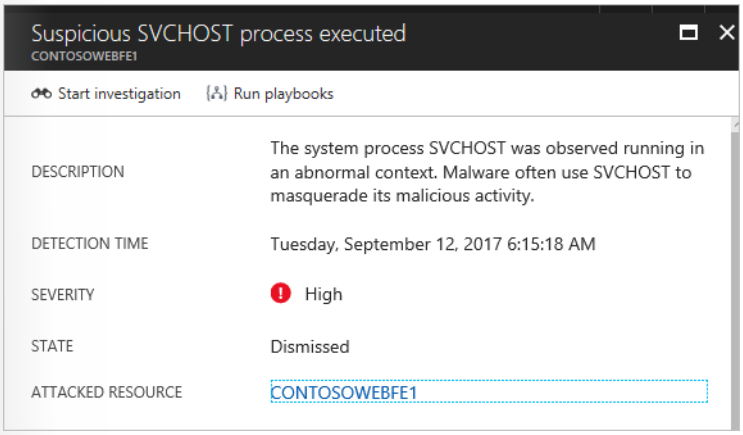
The **Exploration tab** allows you to examine data related to various issues about the entity. For example, when a machine is investigated, the Exploration tab presents a list of processes that executed on it. In some cases, the Exploration tab presents data that might indicate a suspicious issue. You can examine the data on the tab or open it on the search screen to examine large sets of data and to use advanced search options, such as filtering and exporting to Microsoft Excel.



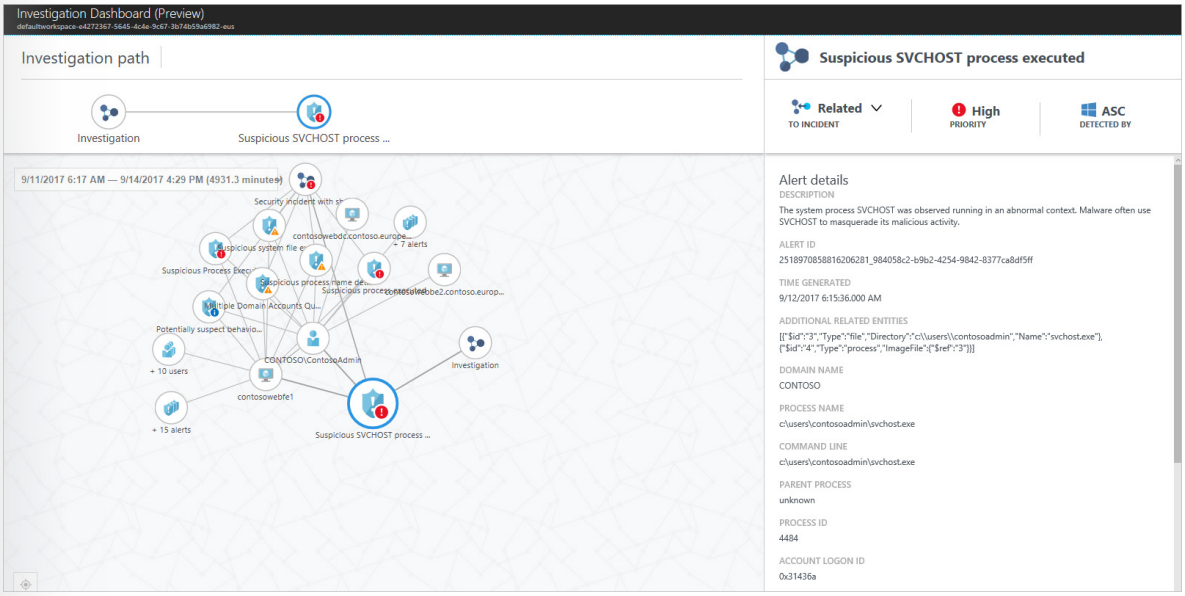
Start the investigation

You can start your investigation from a security incident or from an alert according to your needs. To start an investigation from an alert:

1. Open the Security Center dashboard.
2. Select Security Alerts, and then select the incident that you want to investigate.
3. On the incident's page, select Start Investigation. The Investigation dashboard opens.



1. On this dashboard, select the entity on the map. The relevant information about this entity appears on the right side of the screen.



At this point, you can explore the entities that were involved in this incident and more details about each one.

Analyze threat protection and manage actionable recommendations by using Azure Security Center

To analyze how well one is protected from various threats, a security score would be very helpful. With so many services offering security benefits, it's often hard to know which what steps to take first to help secure and harden your workload. Azure Secure Score reviews your security recommendations and prioritizes them for you, so you know which recommendations to perform first. This helps you find the most serious security vulnerabilities so you can prioritize your investigation. Azure Secure Score helps you assess your workload security posture.

Azure Security Score calculation

Security Center mimics the work of a security analyst, reviewing your security recommendations and applying advanced algorithms to determine how crucial each recommendation is. Security Center constantly reviews your active recommendations and calculates your secure score based on them. The score of a recommendation is derived from its severity and from security best practices that will affect your workload security the most.

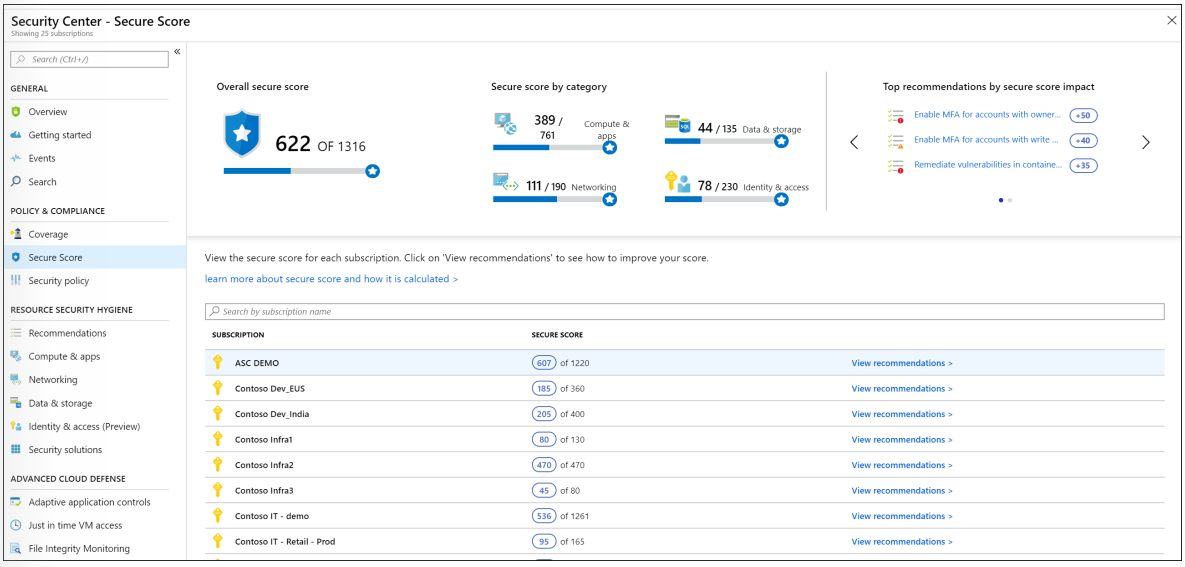
Security Center also provides you with an overall secure score.

The overall secure score is an accumulation of all your recommendation scores. You can view your overall secure score across your subscriptions or management groups, depending on what you select. The score varies based on the selected subscriptions and the active recommendations for those subscriptions.

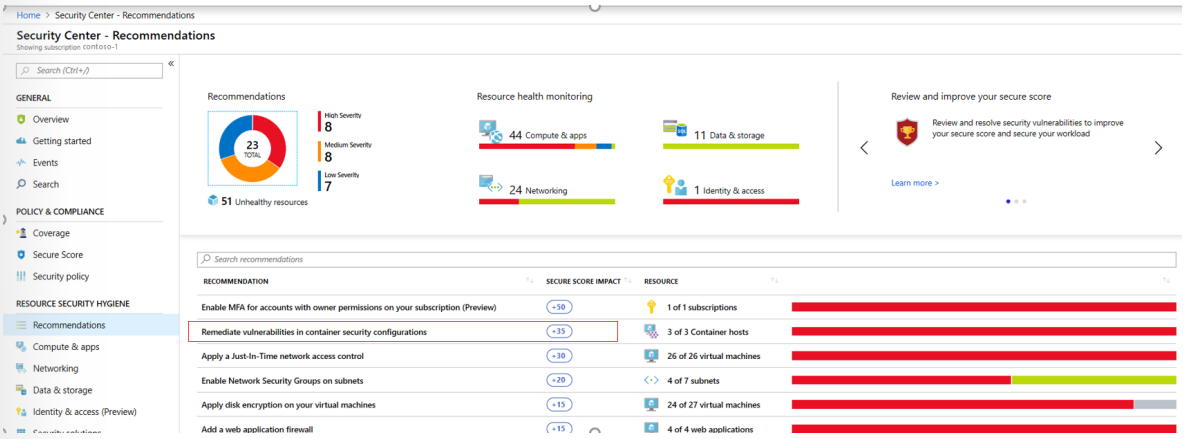
To check which recommendations, impact your secure score most, you can view the top three most impactful recommendations on the **Security Center dashboard**, or you can sort the recommendations on the **recommendations list** blade by using the **Secure score impact** column.

To view your overall secure score:

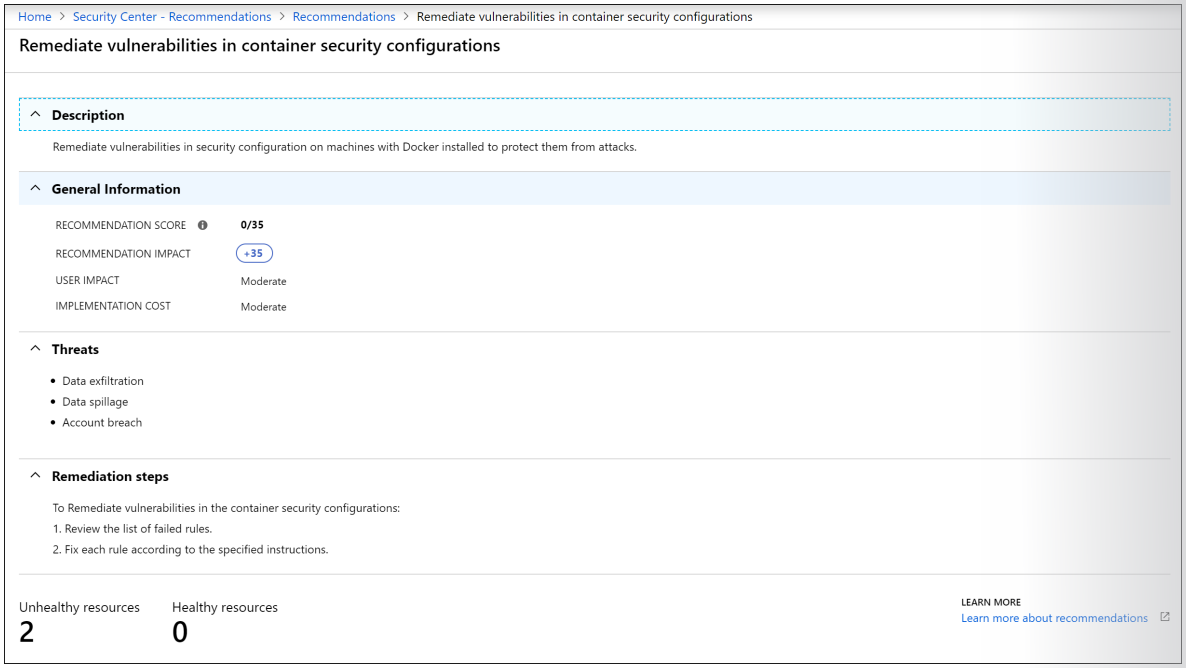
1. On the **Azure** dashboard, select **Security Center**, and then select **Secure score**.
2. Note that at the top of the screen, the following secure score highlights are displayed:
   - **Overall secure score** represents the score per policies, per selected subscription.
   - **Secure score by category** lists which resources need the most attention.
   - **Top recommendations by secure score impact** lists the recommendations that will improve your secure score the most if you implement them.



1. Note that the table at the bottom of the screen has each of your subscriptions and the overall secure score for each. The sum of the secure scores of the subscriptions doesn't equal the overall secure score. The overall secure score is a calculation based on the ratio of your healthy resources to your total resources per recommendation and not a sum of secure scores across your subscriptions.
2. Select **View recommendations** to view the recommendations for a particular subscription that you can remediate to improve your secure score.
3. Note that in the list of recommendations, each recommendation has a number in the **SECURE SCORE IMPACT** column. This number represents how much your overall secure score will improve if you follow the recommendations. For example, if you follow the **Remediate vulnerabilities in container security configurations** recommendation depicted in the following figure, your secure score will increase by 35 points.



The secure score is calculated based on the ratio of your healthy resources to your total number of resources. If the number of healthy resources equals the total number of resources, you get the highest secure score possible for a recommendation, which is 50. To get your secure score closer to the maximum score, you can fix the unhealthy resources by following the remediation steps in the recommendation.



When a recommendation is remediated, both the recommendation score and the overall secure score update.

The main goal of Azure Secure Score is to provide these capabilities to your organization:

- The visualization and conceptualization of the security posture
- Fast triage and suggestions for meaningful actions to increase your security posture
- The measurement of the workload security over time

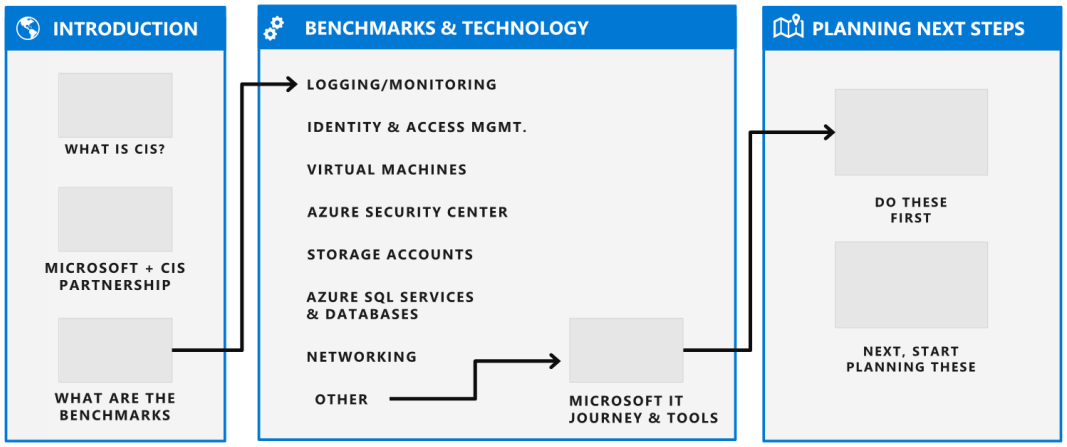
Create security baselines

Create a platform security baseline

Azure doesn't monitor for or respond to security incidents within the customer's area of responsibility, as discussed in Module 2, "Implement platform protection." Azure does provide many tools (such as Azure Security Center) that are used for this purpose. There is also an effort to help make every service as secure as possible by default. That is, every service comes with a baseline that is already designed to help provide security for most common-use cases. However, because no way exists to predict how a service will be used, you need to review these security controls to evaluate whether they adequately mitigate risks.

Microsoft's cybersecurity group in conjunction with the Center for Internet Security (**CIS**[34]) developed best practices to help establish security baselines.

Securing Azure Workloads with the CIS Benchmark



Microsoft initially partnered with CIS for the development of an off-the-shelf **hardened Azure VM**[35]. An initiative then began to use the CIS Benchmarks (their term for best practices) with Azure security services and tools to facilitate security and compliance for customer applications running on Azure services.

This document, **CIS Microsoft Azure Foundations Security Benchmark**[36], provides prescriptive guidance for establishing a secure baseline configuration for Azure. This guide was tested against the listed Azure services as of February 2018. The scope of this benchmark is to establish the foundational level of security for anyone adopting Azure.

The following section looks at this foundation.

A variety of security standards can help cloud service customers to achieve workload security when using cloud services. The following are recommended technology groupings to help create secure cloud-enabled workloads. These recommendations should not be considered an exhaustive list of all possible security configurations and architectures but just as a starting point.

34 https://www.cisecurity.org/
35 https://www.cisecurity.org/hardened-images/
36 https://azure.microsoft.com/en-us/resources/cis-microsoft-azure-foundations-security-benchmark/

CIS has implementation levels:

- **Level 1**. Recommended minimum security settings:
  - These should be configured on all systems.
  - These should cause little or no interruption of services nor reduced functionality.
- **Level 2**. Recommendations for highly secure environments:
  - These might result in reduced functionality.

For a detailed description of the information in the following table, refer to the **CIS Microsoft Azure Foundations Security Benchmark**[37] document.

Technology grouping

Technology group	Description	Number of recommendations
Identity & Access Management (IAM)	Recommendations related to IAM policies	23
Azure Security Center	Recommendations related to the configuration and use of Azure Security Center	19
Storage accounts	Recommendations for setting storage account policies	7
Azure SQL Database	Recommendations for helping secure Azure SQL databases	8
Logging and monitoring	Recommendations for setting logging and monitoring policies for your Azure subscriptions	13
Networking	Recommendations for helping to securely configure Azure networking settings and policies	5
VMs	Recommendations for setting security policies for Azure compute services—specifically, VMs	6
Other	Recommendations regarding general security and operational controls, including those related to Azure Key Vault and resource locks	3
	Total recommendations	84

37 https://azure.microsoft.com/en-us/resources/cis-microsoft-azure-foundations-security-benchmark/

Create an Identity & Access Management (IAM) baseline

As discussed in Module 1, "Manage identity and access," identity management is key to granting access and to the security enhancement of corporate assets.

Manage identity and access for your Azure administrators, application developers, and application users.

Here are the IAM recommendations:

- Restrict access to the Azure AD admin portal.
- About Azure Multi-Factor Authentication (MFA):
  - Enable it for privileged and nonprivileged users.
  - Block remembering MFA on trusted devices.
- About guests:
  - Ensure that no guest users exist.
  - Limit guest user permissions.
- About password security:
  - Notify users on password resets.
  - Notify all admins when other admins reset passwords.
  - Require two methods to reset passwords.
  - Establish an interval for reconfirming user authentication methods.
- Members and guests can invite:
  - Users to create and manage security groups.
- Self-service group management.
- Disable application options, and:
- Allow users to register apps.
  - Allow users to add apps to access the portal.
  - Allow users to consent to apps accessing company data.

Create an Azure Security Center baseline

The following are Security Center recommendations that, if followed, will set various security policies on an Azure subscription.

These policies define the set of controls that are recommended for your resources with an Azure subscription.

Here are the recommendations:

- Enable the Standard pricing tier.
- Enable the automatic provision of a monitoring agent.
- Enable System Updates.

- Enable Security Configurations.
- Enable Endpoint Protection.
- Enable Disk Encryption.
- Enable Network Security Groups.
- Enable Web Application Firewall.
- Enable Vulnerability Assessment.
- Enable Storage Encryption.
- Enable JIT Network Access.
- Enable Adaptive Application Controls.
- Enable SQL Auditing & Threat Detection.
- Enable SQL Encryption.
- Set Security Contact Email.
- Set Security Contact Phone Number.
- Enable Send me emails about alerts.
- Enable Send email also to subscription owners.

Create a storage accounts baseline

This section covers security recommendations for your Azure storage account policies.

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

Here are the recommendations:

- Require security-enhanced transfers.
- Enable binary large object (blob) encryption.
- Periodically regenerate access keys.
- Require SAS tokens to expire within an hour.
- Require SAS tokens to be shared only via HTTPS.
- Enable Azure Files encryption.
- Require only private access to blob containers.

Create an Azure SQL Database baseline

This section covers security recommendations that you should follow to set Microsoft SQL Server policies on your Azure subscription.

Here are the recommendations:

- Enable auditing.
- Enable a threat detection service.
- Enable all threat detection types.

- Enable the option to send security alerts.
- Enable the email service and co-administrators.
- Configure audit retention for more than 90 days.
- Configure threat detection retention for more than 90 days.
- Configure Azure AD administration.

Create a logging and monitoring baseline

This section covers security recommendations that you should follow to set logging and monitoring policies on your Azure subscription.
Here are the recommendations:

- Ensure that a log profile exists.
- Ensure that activity log retention is set to 365 days or more.
- Create an activity log alert for:
  - Creating a policy assignment.
  - Updating a security policy.
  - Creating, updating, or deleting a security solution.
- Enable Azure Key Vault logging.
- Create an activity log alerts for:
  - Creating, updating, or deleting an NSG.
  - Creating, updating, or deleting an NSG rule.
  - Creating or updating an SQL Server firewall rule.
  - Creating an activity log alert for deleting an SQL Server firewall rule.

Note: An Azure activity log is a subscription log that provides insight into subscription-level events that have occurred in Azure.

Using the activity log, you can determine the what, the who, and the when for any write operations taken on the resources in your subscription.

Activity logs provide data on a resource from the outside, or control plane.

A resource produces diagnostic logs, which provide information about the operation of that resource, or data plane.

Create a networking baseline

This section covers security recommendations that you should follow to set networking policies on your Azure subscription.

Here are the recommendations:

- Restrict RDP access from the internet.
- Restrict SSH access from the internet.
- Restrict SQL Server access from the internet.

- Configure the NSG flow log retention period for more than 90 days.
- Enable Network Watcher.

Create a VMs baseline

This section covers security recommendations that you should follow to set VM policies on your Azure subscription.

Here are the recommendations:

- A VM agent must be installed to enable data collection for Azure Security Center.

  Data is needed to assess the VM security state, provide security recommendations, and alert on host-based threats.
- **Note** that encryption helps protect the OS disk and its content from unwanted reads without the key.
- Help protect the data disk from unwanted reads.
- **Note** that extensions are small applications that provide post-deployment configuration and automation tasks on VMs.

  Extensions run with admin privileges and have access to all data and configurations for the VM. Carefully review extensions

  to help ensure that they don't compromise the security of the host or Azure subscription.
- **Note** that Azure Security Center retrieves a list of available security updates and critical updates from WSUS and checks

  whether they've been applied to the VM. VMs must be updated to help ensure their security.
- Ensure that VMs have an installed and running endpoint protection solution.

Other security considerations for a baseline

This section covers security recommendations that you should follow to set general security and operational controls on your Azure subscription.

Here are the recommendations:

- Set an expiration date on all keys.
- Set an expiration date on all secrets.
- Set resource locks for mission-critical Azure resources.

Summary of CIS recommendations

- Turn on Azure Security Center—it's free.
- Adopt CIS Benchmarks:
  - Apply them to existing tenants.
  - Use CIS VMs for new workloads (from Azure Marketplace).

Module 4 Secure Data and Applications

Configure Security Policies to Manage Data

Configure data classification

Classifying your data and identifying your data protection needs helps you select the right cloud solution for your organization. Data classification enables organizations to find storage optimizations that might not be possible when all data is assigned the same value. Classifying (or categorizing) stored data by sensitivity and business impact helps organizations determine the risks associated with the data. After your data has been classified, organizations can manage their data in ways that reflect their internal value instead of treating all data the same way.

Data classification can yield benefits such as compliance efficiencies, improved ways to manage the organization's resources, and facilitation of migration to the cloud. Some data protection solutions—such as encryption, rights management, and data loss prevention—have moved to the cloud and can help mitigate cloud risks. However, organization must be sure to address data classification rules for data retention when moving to the cloud.

Data exists in one of three basic states: at rest, in process, and in transit. All three states require unique technical solutions for data classification, but the applied principles of data classification should be the same for each. Data that is classified as confidential needs to stay confidential when at rest, in process, or in transit.

Data can also be either structured or unstructured. Typical classification processes for structured data found in databases and spreadsheets are less complex and time-consuming to manage than those for unstructured data such as documents, source code, and email. Generally, organizations will have more unstructured data than structured data.

Regardless of whether data is structured or unstructured, it's important for organizations to manage data sensitivity. When properly implemented, data classification helps ensure that sensitive or confidential data assets are managed with greater oversight than data assets that are considered public distribution.

Protect data at rest

Data encryption at rest is a mandatory step toward data privacy, compliance, and data sovereignty.

Best practice	Solution
Apply disk encryption to help safeguard your data.	Use **Microsoft Azure Disk Encryption** *(https://docs.microsoft.com/en-us/azure/security/azure-security-disk-encryption-overview)*, which enables IT administrators to encrypt both Windows infrastructure as a service (IaaS) and Linux IaaS virtual machine (VM) disks. Disk encryption combines the industry-standard BitLocker feature and the Linux DM-Crypt feature to provide volume encryption for the operating system (OS) and the data disks. Azure Storage and Azure SQL Database encrypt data at rest by default, and many services offer encryption as an option. You can use Azure Key Vault to maintain control of keys that access and encrypt your data. See **Azure resource providers encryption model support** *(https://docs.microsoft.com/en-us/azure/security/azure-security-encryption-atrest)* to learn more.
Use encryption to help mitigate risks related to unauthorized data access.	Encrypt your drives before you write sensitive data to them.

Organizations that don't enforce data encryption are risk greater exposure to data-integrity issues. For example, unauthorized users or malicious hackers might steal data in compromised accounts or gain unauthorized access to data coded in Clear Format. To comply with industry regulations, companies also must prove that they are diligent and using correct security controls to enhance their data security.

Protect data in transit

Protecting data in transit should be an essential part of your data protection strategy. Because data is moving back and forth from many locations, we generally recommend that you always use SSL/TLS protocols to exchange data across different locations. In some circumstances, you might want to isolate the entire communication channel between your on-premises and cloud infrastructures by using a VPN.

For data moving between your on-premises infrastructure and Azure, consider appropriate safeguards such as HTTPS or VPN. When sending encrypted traffic between an Azure virtual network and an on-premises location over the public internet, use Azure VPN Gateway.

The following table lists best practices specific to using Azure VPN Gateway, SSL/TLS, and HTTPS.

Best practice	Solution
Secure access from multiple workstations located on-premises to an Azure virtual network	Use site-to-site VPN.
Secure access from an individual workstation located on-premises to an Azure virtual network	Use point-to-site VPN.
Move larger data sets over a dedicated high-speed wide area network (WAN) link	Use Azure ExpressRoute. If you choose to use ExpressRoute, you can also encrypt the data at the application level by using SSL/TLS or other protocols for added protection.
Interact with Azure Storage through the Azure portal	All transactions occur via HTTPS. You can also use Storage REST API over HTTPS to interact with Azure Storage and Azure SQL Database.

Organizations that fail to protect data in transit are more susceptible to man-in-the-middle attacks, eavesdropping, and session hijacking. These attacks can be the first step in gaining access to confidential data.

Now that we've covered the physical aspects of data classification, let's look at the classification based on discovery and classification.

Data discovery and classification (currently in preview) provides advanced capabilities built into Azure SQL Database for discovering, classifying, labeling and protecting sensitive data (such as business, personal data (PII), and financial information) in your databases. Discovering and classifying this data can play a pivotal role in your organizational information protection stature. It can serve as infrastructure for:

- Helping meet data privacy standards and regulatory compliance requirements.
- Addressing various security scenarios such as monitoring, auditing, and alerting on anomalous access to sensitive data.
- Controlling access to and hardening the security of databases containing highly sensitive data.

Data discovery and classification is part of the **Advanced Data Security**[1] offering, which is a unified package for advanced Microsoft SQL Server security capabilities. You access and manage data discovery and classification via the central SQL Advanced Data Security portal.

Data discovery and classification introduces a set of advanced services and SQL capabilities, forming a SQL Information Protection paradigm aimed at protecting the data, not just the database:

- Discovery and recommendations. The classification engine scans your database and identifies columns containing potentially sensitive data. It then provides you with an easier way to review and apply the appropriate classification recommendations via the Azure portal.
- Labeling. Sensitivity classification labels can be persistently tagged on columns using new classification metadata attributes introduced into the SQL Server Engine. This metadata can then be utilized for advanced sensitivity-based auditing and protection scenarios.
- Query result set sensitivity. The sensitivity of the query result set is calculated in real time for auditing purposes.
- Visibility. You can view the database classification state in a detailed dashboard in the Azure portal. Additionally, you can download a report (in Microsoft Excel format) that you can use for compliance and auditing purposes, in addition to other needs.

Steps for discovery, classification, and labeling

Classifications have two metadata attributes:

- Labels. These are the main classification attributes used to define the sensitivity level of the data stored in the column.
- Information Types. These provide additional granularity into the type of data stored in the column.

SQL data discovery and classification comes with a built-in set of sensitivity labels and information types, and discovery logic. You can now customize this taxonomy and define a set and ranking of classification constructs specifically for your environment.

Definition and customization of your classification taxonomy takes place in one central location for your entire Azure Tenant. That location is in **Azure Security Center**[2], as part of your Security Policy. Only a user with administrative rights on the Tenant root management group can perform this task.

1 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-advanced-data-security
2 https://docs.microsoft.com/azure/security-center/security-center-intro

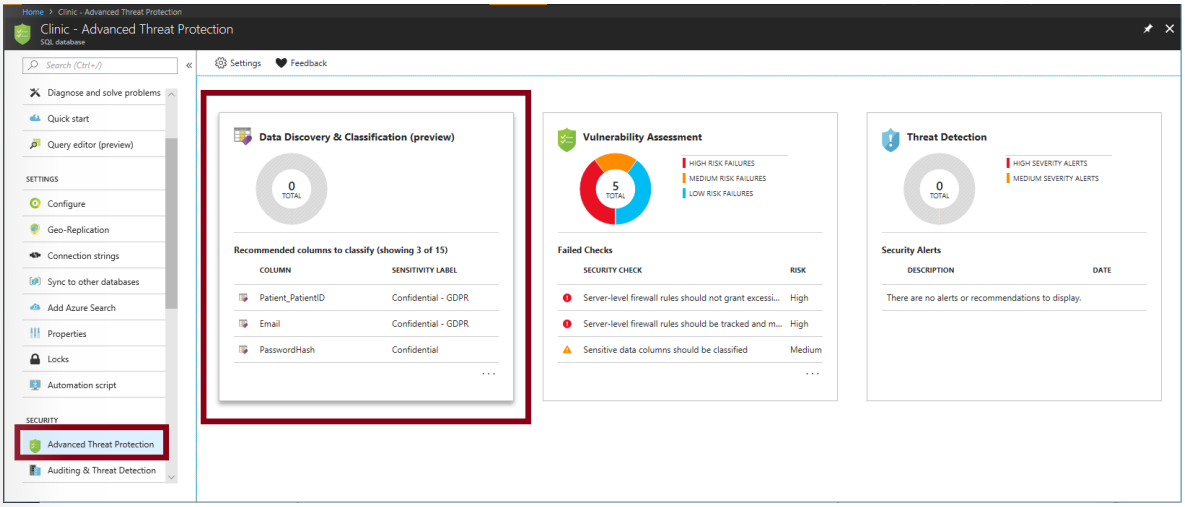
As part of Azure Information Protection policy management, you can define custom labels, rank them, and associate them with a selected set of information types. You can also add your own custom information types and configure them with string patterns, which are added to the discovery logic for identifying this type of data in your databases. Learn more about customizing and managing your policy in the **Information Protection policy how-to guide**[3].

After you've defined the tenant-wide policy, you can continue with classifying individual databases using your customized policy.

Exercise Classify your SQL Database

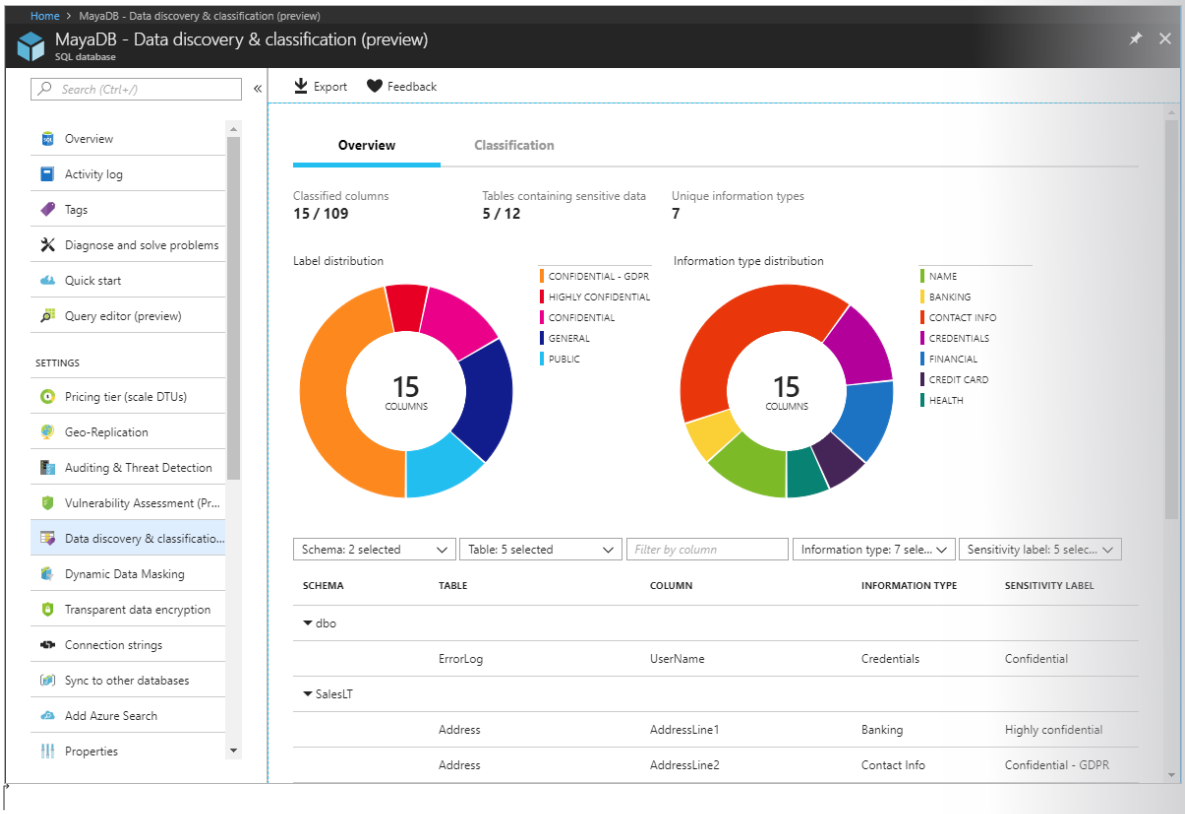
- Sign-in to the Azure portal.
- Under the Security heading in the Azure SQL Database pane, navigate to Advanced Data Security, and select to enable advanced data security.
- Select the Data discovery and classification (preview) card.

Note: depending on when you work on this exercise, the preview status might be removed.



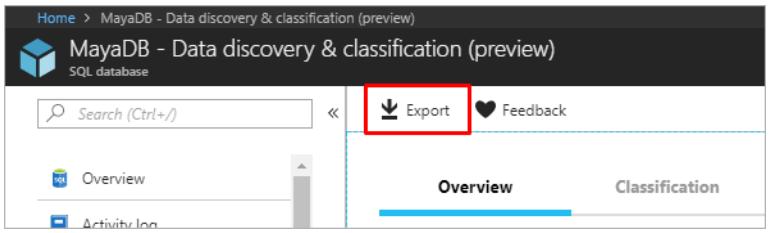
- Review the Overview tab. Notice that it includes a summary of the current classification state of the database, including a detailed list of all classified columns. You can also filter this view to only see specific schema parts, information types, and labels.

3 https://go.microsoft.com/fwlink/?linkid=2009845&clcid=0x409



Note: The next two steps assume you have classified data to export. If you don't, just review the steps.

- To download a report in Excel format, in the top menu of the window select Export.

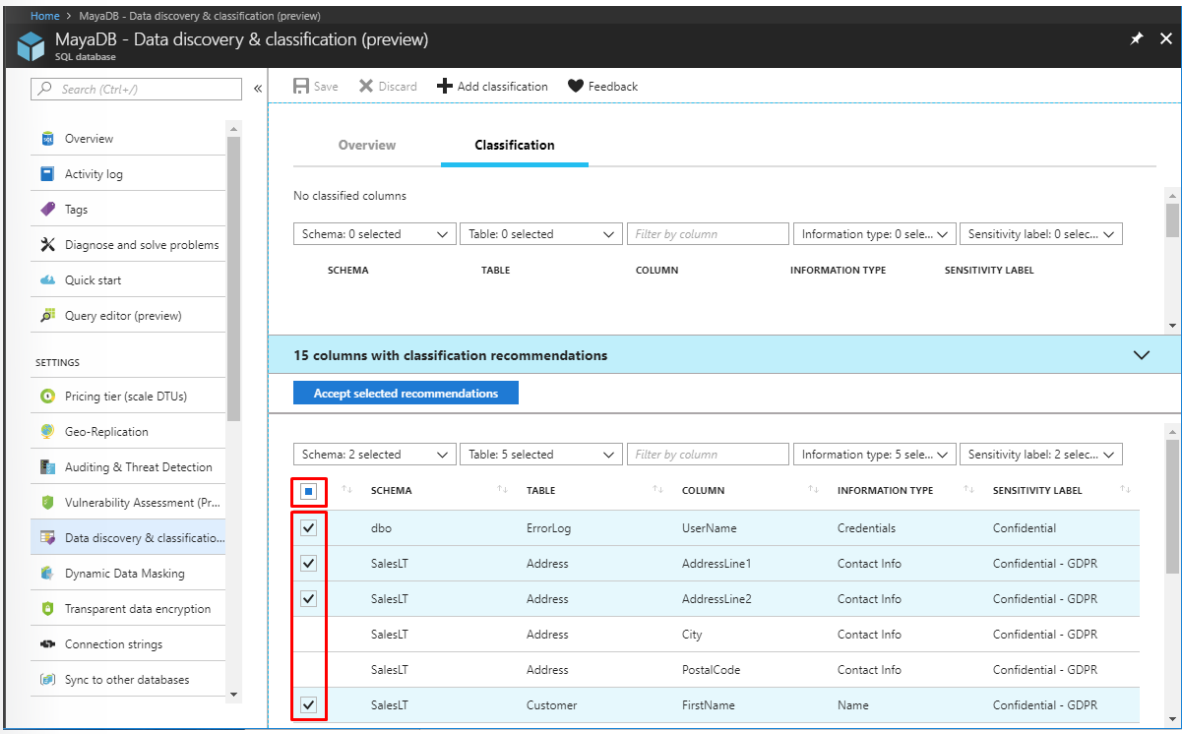


- To begin classifying your data, select the Classification tab at the top of the window.

The classification engine scans your database for columns containing potentially sensitive data and provides a list of recommended column classifications.

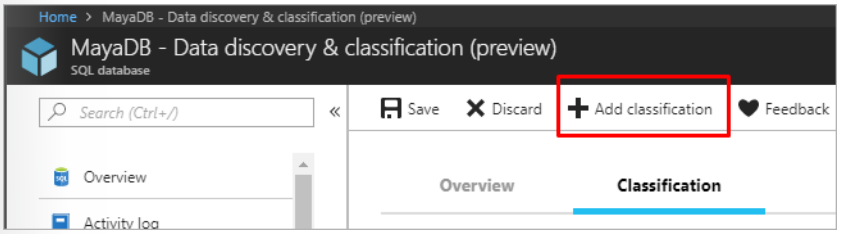
- To view and apply classification recommendations:
- View the list. To view the list of recommended column classifications, select the recommendations panel at the bottom of the window.

- Accept recommendations. To accept a recommendation for a specific column, select the check boxes in the left column of the relevant rows. You can also mark all recommendations as accepted by selecting the check box in the recommendations table header.



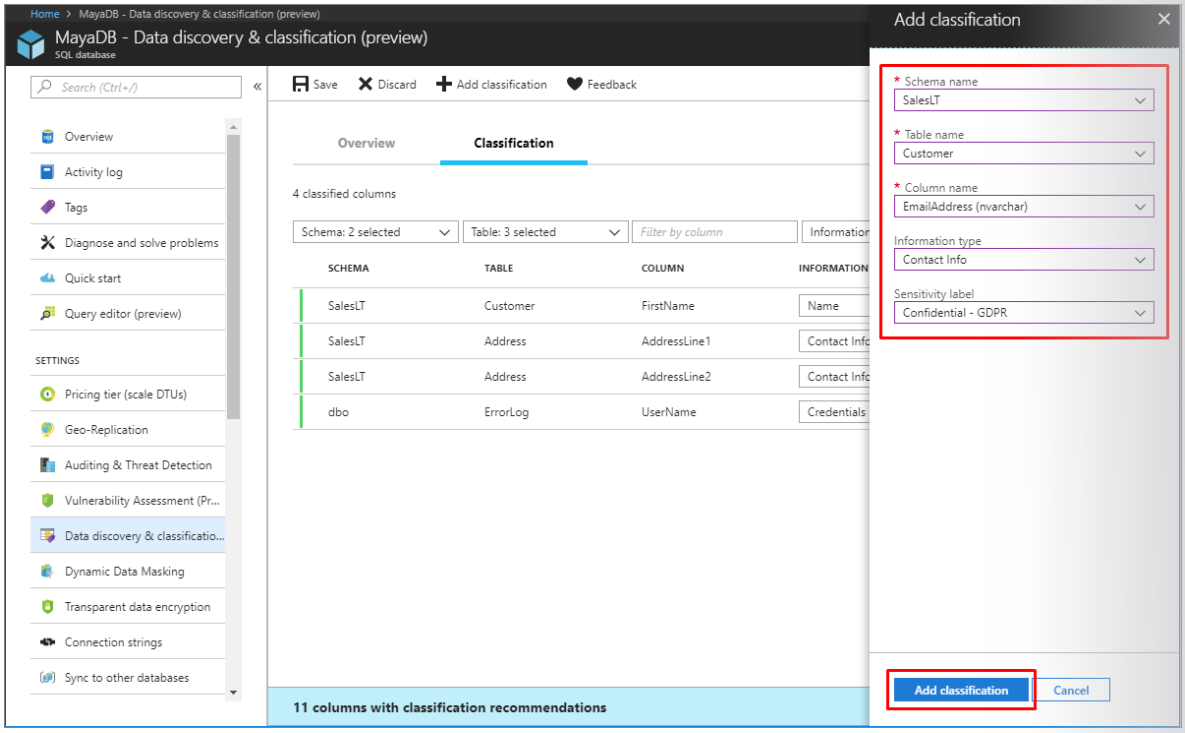
- To apply the selected recommendations, select the blue Accept selected recommendations button.

a) To manually classify columns as an alternative to or in addition to the recommendation-based classification, in the top menu of the window, select Add classification.



b) In the Add classification blade, configure the five fields that display, and then select Add classification:

- Schema name
- Table name
- Column name
- Information type
- Sensitivity label.



- To complete your classification and persistently label (tag) the database columns with the new classification metadata, in the top menu of the window, select Save.

Monitor access to sensitive data

An important aspect of the IP paradigm is the ability to monitor access to sensitive data. **Azure SQL Database Auditing**[4] has been enhanced to include a new field in the audit log. The data_sensitivity_information field logs the sensitivity classifications (labels) of the actual data that was returned by the query.

d	client_ip	application_name	duration_milliseconds	response_rows	affected_rows	connection_id	data_sensitivity_information
	[redacted]7.125	Microsoft SQL Server Management Studio - Query	1	847	847	C244A066-2271-...	Confidential - GDPR
	[redacted]7.125	Microsoft SQL Server Management Studio - Query	2	32	32	C244A066-2271-...	Confidential
	[redacted]7.125	Microsoft SQL Server Management Studio - Query	41	32	32	A7088FD4-759E-...	Confidential, Confidential - GDPR

Configure data retention

Data recovery and disposal, like data reclassification, is an essential aspect of managing data assets. The principles for data recovery and disposal are defined by a data retention policy and enforced in the same manner as data reclassification. These tasks are typically performed by the custodian and administrator roles as a collaborative task.

Failure to maintain a data retention policy could mean data loss, or failure to comply with regulatory and legal discovery requirements. Most organizations that do not have a clearly defined data retention policy tend to use a default, Keep everything retention policy. However, this poses additional risks in cloud services scenarios. For example, a data retention policy for cloud service providers can be considered as "for the duration of the subscription," meaning as long as the service is paid for, the data is retained. Such a pay-for-retention agreement might not address corporate or regulatory retention policies.

4 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-auditing

Defining a policy for confidential data can ensure that data is stored and removed based on best practices. In addition, you can create an archival policy to formalize an understanding about what data should be disposed of and when.

A data retention policy should address the required regulatory and compliance requirements, and corporate legal retention requirements. Classified data might provoke questions about retention duration and exceptions for data that has been stored with a provider, although such questions most likely pertain to data that has not been classified correctly. Data classification rules that pertain to data retention must be addressed when moving to the cloud, and that cloud solutions can help mitigate risk. Some data protection technologies such as encryption, rights management, and data loss prevention solutions have moved to the cloud and can help mitigate cloud risks.

Immutable storage and data retention

Immutable storage for Azure Blob Storage enables users to store business-critical data in a write once, read many (WORM) state. This state makes the data unerasable and unmodifiable for a user-specified interval. Blobs can be created and read, but not modified or deleted, for the duration of the retention interval.

Immutable storage enables:

- Time-based retention policy support. Users set policies to store data for a specified interval.
- Legal hold policy support. When the retention interval is not known, users can set legal holds to store data immutably until the legal hold is cleared. When a legal hold is set, blobs can be created and read, but not modified or deleted. Each legal hold is associated with a user-defined alphanumeric tag that is used as an identifier string (such as a case ID).
- Support for all blob tiers. WORM policies are independent of the Azure Blob Storage tier and apply to all tiers: hot, cool, and archive. Users can transition data to the most cost-optimized tier for their workloads while maintaining data immutability.
- Container-level configuration. Users can configure time-based retention policies and legal hold tags at the container level. By using simple container-level settings, users can create and lock time-based retention policies, extend retention intervals, set and clear legal holds, and more. These policies apply to all the blobs in the container, both existing and new.
- Audit logging support. Each container includes an audit log, which displays up to five time-based retention commands for locked time-based retention policies. However, the log has a maximum of three logs for retention interval extensions.
- For time-based retention, the log contains the user ID, command type, time stamps, and retention interval.
- For legal holds, the log contains the user ID, command type, time stamps, and legal hold tags.

The audit log is retained for the lifetime of the container, in accordance with the SEC 17a-4(f) regulatory guidelines. The Azure Activity Log shows a more comprehensive log of all the control plane activities. It is the user's responsibility to store those logs persistently, as might be required for regulatory or other purposes.

Immutable storage for Azure Blob storage supports two types of WORM or immutable policies: time-based retention and legal holds.

When a time-based retention policy or a legal hold is applied on a container, all existing blobs move to the immutable (write-protected and delete-protected) state. All new blobs that are uploaded to the container will also move to the immutable state.

When a time-based retention policy is applied on a container, all blobs in the container will stay in the immutable state for the duration of the effective retention period. The effective retention period for existing blobs is equal to the difference between the blob creation time and the user-specified retention interval.

For new blobs, the effective retention period is equal to the user-specified retention interval. Because users can extend the retention interval, immutable storage uses the most recent value of the user-specified retention interval to calculate the effective retention period.

For example: A user creates a time-based retention policy with a retention interval of five years. The existing blob in that container, testblob1, was created one year ago. The effective retention period for testblob1 is four years. A new blob, testblob2, is now uploaded to the container. The effective retention period for this new blob is five years.

Legal holds

When you set a legal hold, all new and existing blobs stay in the immutable state until the legal hold is cleared.

A container can have both a legal hold and a time-based retention policy simultaneously. All blobs in that container stay in the immutable state until all legal holds are cleared, even if their effective retention period has expired. Conversely, a blob stays in an immutable state until the effective retention period expires, even though all legal holds have been cleared.

Exercise

For this exercise, use the **Windows PowerShell sample script**[5]. This script creates a new storage account and container. It then shows you how to set and clear legal holds, create and lock a time-based retention policy (also known as an immutability policy), and extend the retention interval.

Configure data sovereignty

According to the TechTarget article, **What is data sovereignty**[6],

Data sovereignty is the concept that information which has been converted and stored in binary digital form is subject to the laws of the country or region in which it is located. Many of the current concerns that surround data sovereignty relate to enforcing privacy regulations and preventing data that is stored in a foreign country or region from being subpoenaed by the host country or region's government.

In Azure, customer data might be replicated within a selected geographic area for enhanced data durability in case of a major data center disaster, and in some cases will not be replicated outside it.

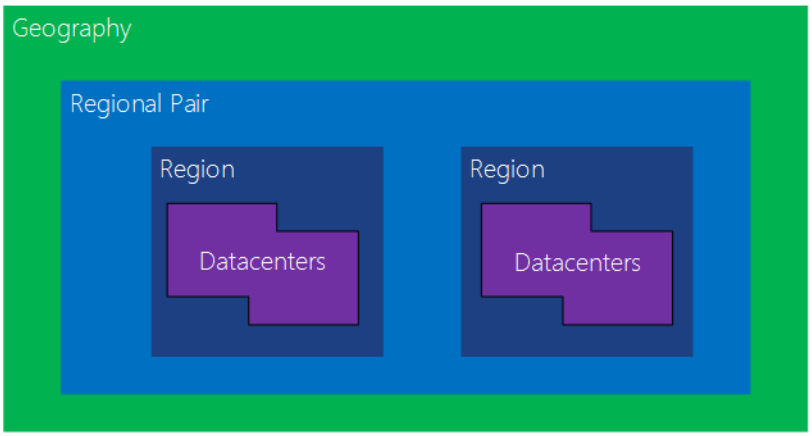
Paired regions

Azure operates in multiple geographies around the world. An Azure geography is a defined area of the world that contains at least one Azure Region. An Azure region is an area within a geography, containing one or more datacenters.

Each Azure region is paired with another region within the same geography, forming a regional pair. The exception is Brazil South, which is paired with a region outside its geography. Across the region pairs Azure serializes platform updates (or planned maintenance), so that only one paired region is updated at

5 https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-immutable-storage#sample-powershell-code
6 https://whatis.techtarget.com/definition/data-sovereignty

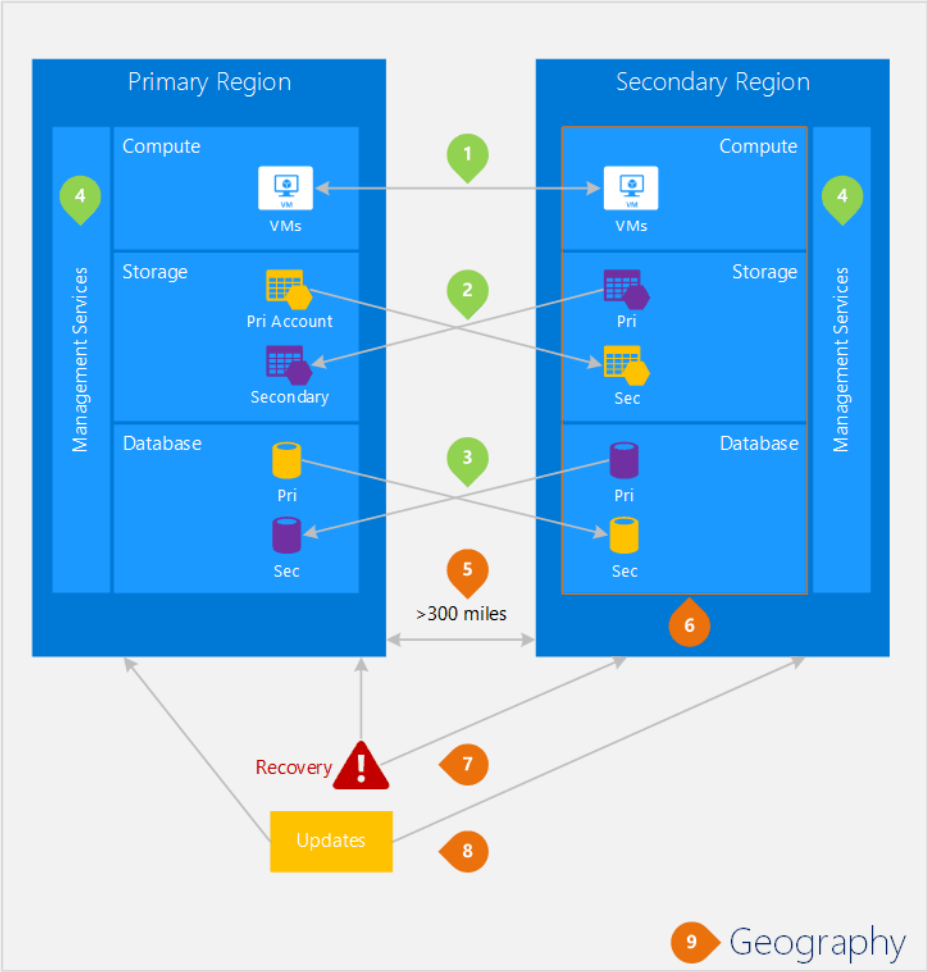
a time. In the event of an outage affecting multiple regions, one region in each pair will be prioritized for recovery.



We recommend that you configure business continuity and disaster recovery (BCDR) across regional pairs to benefit from Azure's isolation and VM policies. For applications that support multiple active regions, we recommend using both regions in a region pair where possible. This will ensure optimal availability for applications and minimized recovery time in the event of a disaster.

An example of paired regions

The following illustration is of a hypothetical application that uses a regional pair for disaster recovery. The green numbers highlight the cross-region activities of three Azure services (Azure Compute, Azure Storage, and Azure Database) and how they are configured to replicate across regions. The orange numbers highlight the unique benefits of deploying across paired regions.



Cross-region activities number key:

- Azure Compute (IaaS). You must provision additional compute resources in advance to ensure resources are available in another region during a disaster. For more information, see **Designing resilient applications for Azure**[7].
- Azure Storage. Geo-redundant storage (GRS) is configured by default when an Azure Storage account is created. With GRS, data is automatically replicated three times within the primary region, and three times in a paired region. For more information, see **Azure Storage redundancy**[8].
- Azure SQL Database. With Azure SQL Database geo-replication, you can configure asynchronous replication of transactions to any region in the world; however, we recommend you deploy these resources in a paired region for most disaster recovery scenarios. For more information, see **Configure active geo-replication for Azure SQL Database in the Azure portal**[9].

7 https://docs.microsoft.com/en-us/azure/resiliency/resiliency-technical-guidance
8 https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy
9 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-active-geo-replication-portal

- Azure Resource Manager. Resource Manager inherently provides logical isolation of components across regions. This means that logical failures in one region are less likely to impact another other regions.

Benefits of Azure paired regions number key:

- Physical isolation. When possible, Azure services prefers at least 300 miles of separation between datacenters in a regional pair (although this isn't practical or possible in all geographies). Physical datacenter separation reduces the likelihood of both regions being affected simultaneously as a result of natural disasters, civil unrest, power outages, or physical network outages. Isolation is subject to the constraints within the geography, such as geography size, power and network infrastructure availability, and regulations.
- Platform-provided replication. Some services such as geo-redundant storage provide automatic replication to the paired region.
- Region recovery order. In the event of a broad outage, recovery of one region is prioritized out of every pair. Applications that are deployed across paired regions are guaranteed to have one of the regions recovered with priority. If an application is deployed across regions that are not paired, recovery might be delayed. In the worst case the chosen regions might be the last two to be recovered.
- Sequential updates. Planned Azure system updates are rolled out to paired regions sequentially, not at the same time. This helps minimize downtime, the effect of bugs, and logical failures in the rare event of a bad update.
- Data residency. To meet data residency requirements for tax and law enforcement jurisdiction purposes, a region resides within the same geography as its pair (with the exception of Brazil South).

Microsoft also complies with international data protection laws regarding transfers of customer data across borders. For example, to accommodate the continuous flow of information required by international business (including the cross-border transfer of personal data), many Microsoft business cloud services offer customers **European Union Model Clauses**[10] that provide additional contractual guarantees around transfers of personal data for in-scope cloud services. European Union data protection authorities have validated the Microsoft implementation of the EU Model Clauses as being in line with the rigorous privacy standards that regulate international data transfers by companies operating in its member states.

In addition to our commitments under the Standard Contractual Clauses and other model contracts, Microsoft is certified to the EU-U.S. Privacy Shield framework as set forth by the U.S. Department of Commerce regarding the collection, use, and retention of personal information transferred from the European Union to the United States. Microsoft participation in the EU-U.S. Privacy Shield applies to all personal data that is subject to the Microsoft Privacy Statement, and is received from the EU, European Economic Area, and Switzerland. Microsoft also abides by Swiss data protection law regarding the processing of personal data from the European Economic Area and Switzerland.

Microsoft will not transfer to any third party (not even for storage purposes) data that you provide to Microsoft through the use of our business cloud services, and that are covered under the Microsoft Online Services Terms.

Note that no matter where customer data is stored, Microsoft does not control or limit the locations from which customers, or their end users might access their data.

10 https://www.microsoft.com/en-us/TrustCenter/Compliance/eu-model-clauses

Configure Security for Data Infrastructure

Configure a SQL Database firewall

Azure SQL Database and Azure SQL Data Warehouse (both referred to as SQL Database in this lesson) provide a relational database service for Azure and other internet-based applications. To help protect your data, firewalls prevent all access to your database server until you specify which computers have permission. The firewall grants access to databases based on the originating IP address of each request.

In addition to IP rules, the firewall also manages virtual network rules. Virtual network rules are based on virtual network service endpoints. Virtual network rules might be preferable to IP rules in some cases. To learn more, see **Use virtual network service endpoints and rules for database servers**[11].

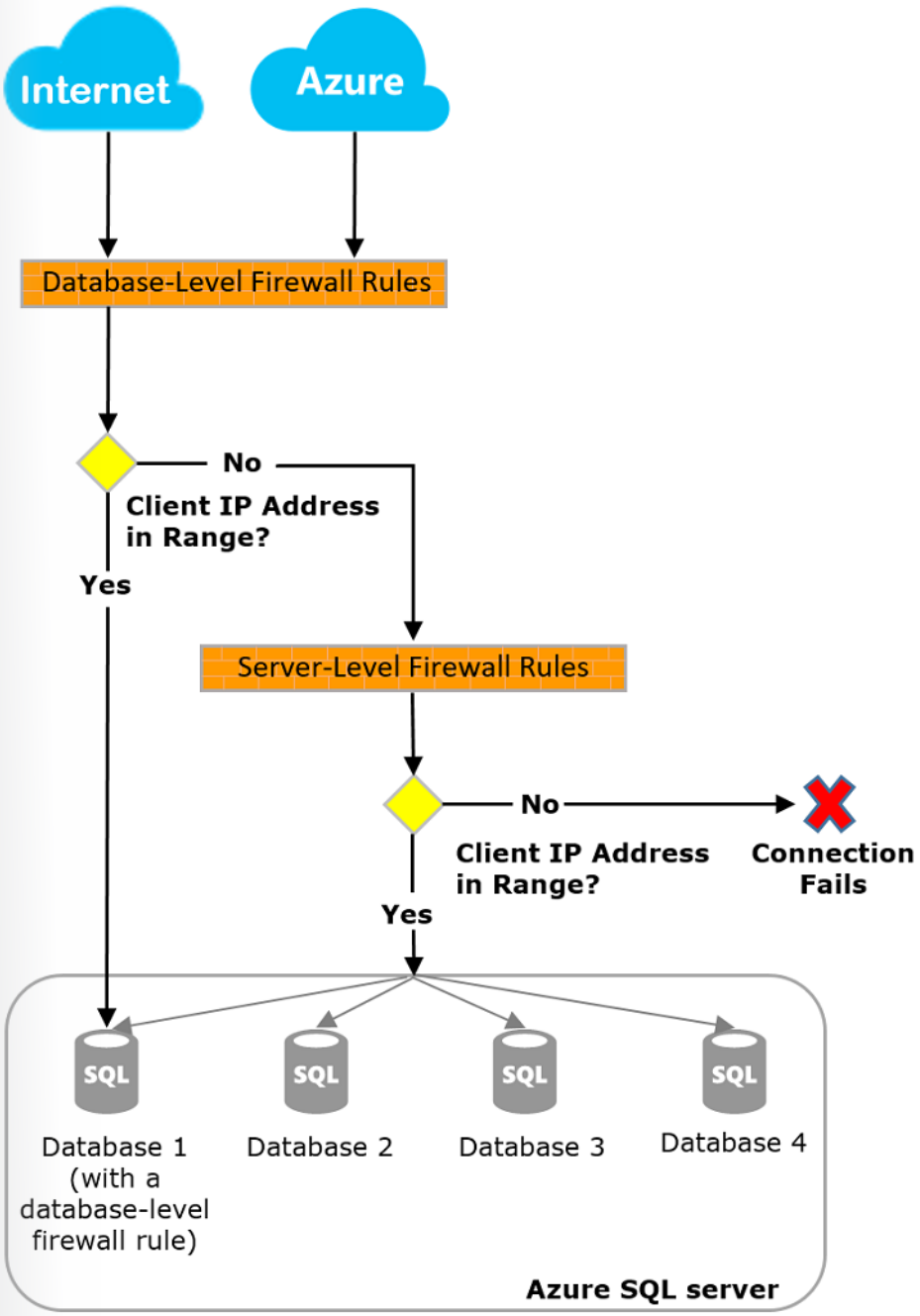
Overview

Initially, all access to your Azure SQL Database is blocked by the SQL Database firewall. To access a database server, you must specify one or more server-level IP firewall rules that enable access to your Azure SQL Database. Use the IP firewall rules to specify which IP address ranges from the internet are allowed, and whether Azure applications can attempt to connect to your Azure SQL Database.

To selectively grant access to just one of the databases in your Azure SQL Database, you must create a database-level rule for the required database. Specify an IP address range for the database IP firewall rule that is beyond the IP address range specified in the server-level IP firewall rule, and ensure that the IP address of the client falls in the range specified in the database-level rule.

Note: SQL Data Warehouse only supports server-level IP firewall rules, and not database-level IP firewall rules.

11 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-vnet-service-endpoint-rule-overview



Connecting from the internet

When a computer attempts to connect to your database server from the internet, the firewall first checks the originating IP address of the request against the database-level IP firewall rules for the database that the connection is requesting:

- If the IP address of the request is within one of the ranges specified in the database-level IP firewall rules, the connection is granted to the SQL Database containing the rule.
- If the IP address of the request is not within one of the ranges specified in the database-level IP firewall rules, the firewall checks the server-level IP firewall rules. If the IP address of the request is

within one of the ranges specified in the server-level IP firewall rules, the connection is granted. Server-level IP firewall rules apply to all SQL databases on the Azure SQL Database.

- If the IP address of the request is not within the ranges specified in any of the database-level or server-level IP firewall rules, the connection request fails.

Connecting from Azure

To allow applications from Azure to connect to your Azure SQL Database, Azure connections must be enabled. When an application from Azure attempts to connect to your database server, the firewall verifies that Azure connections are allowed. A firewall setting with starting and ending addresses equal to 0.0.0.0 indicates Azure connections are allowed. If the connection attempt is not allowed, the request does not reach the Azure SQL Database server.

This option configures the firewall to allow all connections from Azure including connections from the subscriptions of other customers. When selecting this option, make sure your sign-in and user permissions limit access to authorized users only.

Server-level IP firewall rules

Server-level IP firewall rules enable clients to access your entire Azure SQL Database—that is, all the databases within the same SQL Database server. These rules are stored in the master database.

You can configure server-level IP firewall rules using the Azure portal, or by using Transact-SQL statements. To create server-level IP firewall rules using the Azure portal or PowerShell, you must be the subscription owner or a subscription contributor. To create a server-level IP firewall rule using Transact-SQL, you must connect to the SQL Database instance as the server-level principal login or the Azure Active Directory (Azure AD) administrator (which means that a server-level IP firewall rule must have first been created by a user with Azure-level permissions).

Database-level IP firewall rules

Database-level IP firewall rules enable clients to access certain secure databases within the same SQL Database server. You can create these rules for each database (including the master database), and they are stored in the individual databases. You can only create and manage database-level IP firewall rules for master databases and user databases by using Transact-SQL statements, and only after you have configured the first server-level firewall. If you specify an IP address range in the database-level IP firewall rule that is outside the range specified in the server-level IP firewall rule, only those clients that have IP addresses in the database-level range can access the database. You can have a maximum of 128 database-level IP firewall rules for a database.

Recommendation

Whenever possible, as a best practice, use database-level IP firewall rules to enhance security and to make your database more portable. Use server-level IP firewall rules for administrators and when you have several databases with the same access requirements, and you don't want to spend time configuring each database individually.

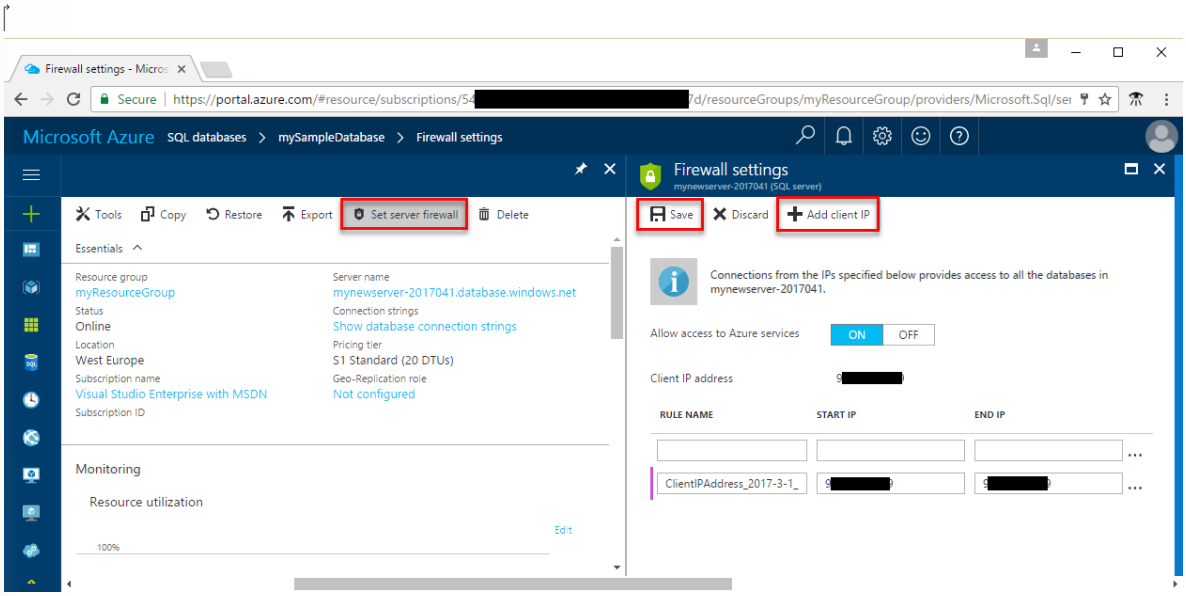
Creating and managing IP firewall rules

You create the first server-level firewall setting using the Azure portal, or programmatically using PowerShell, Azure Command-Line Interface (Azure CLI), or the REST API. You can create and manage subsequent server-level IP firewall rules using these methods, and through Transact-SQL.

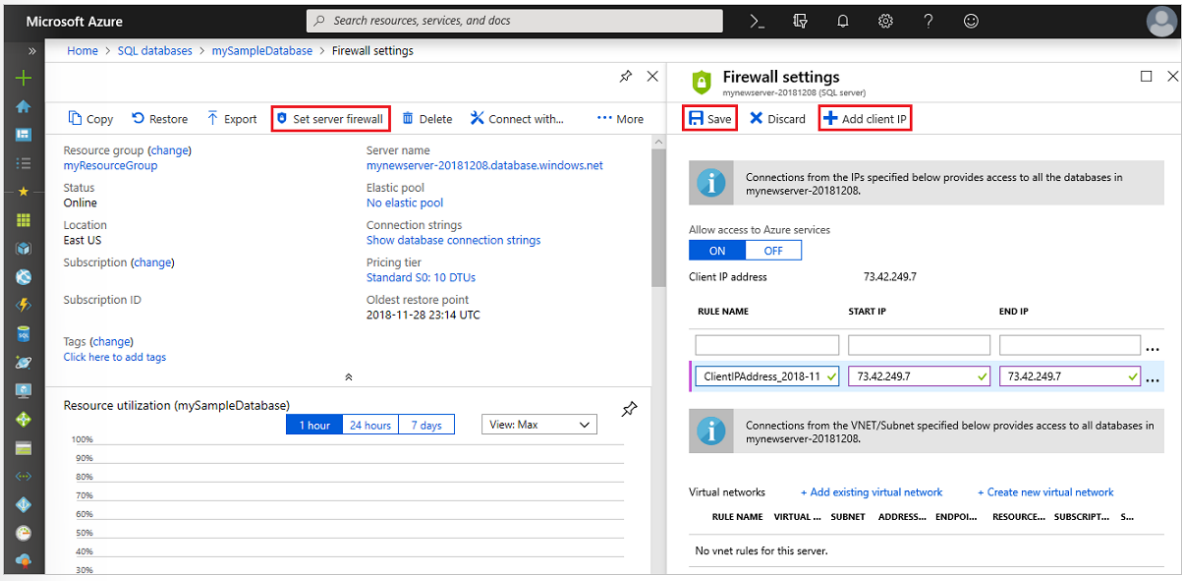
To create a DB using the Azure portal:

1. Sign into your Azure portal.
2. From the left pane, select Database.

3. From the database overview page, on the toolbar, select Set server firewall.



4. In the Firewall settings blade, select Add client IP to add the IP address of the computer you are currently using, and then select Save.
5. Verify that a server-level IP firewall rule is created for your current IP address.



6. When the overview page for your server opens, review the fully qualified server name (such as mynew-server20170403.database.windows.net) and the provided options for further configuration.
7. To set a server-level rule from server overview page, under Settings, select Firewall.
8. On the toolbar, select Add client IP to add the IP address of the computer you are currently using, and then select Save.
9. Verify that a server-level IP firewall rule is created for your current IP address.

For more information on other options for IP management for firewall rules, see:

- **Manage IP firewall rules using Transact-SQL**[12]
- **Manage server-level IP firewall rules using Azure PowerShell**[13]
- **Manage server-level IP firewall rules using Azure CLI**[14]
- **Manage server-level IP firewall rules using REST API**[15]

Enable database authentication

In this lesson, we examine the authentication process for Azure Cosmos DB, SQL Database, and Azure HDInsight.

SQL Database authentication

SQL Database supports two types of authentication:

- SQL authentication. This authentication method uses a username and password. When you created the SQL Database server for your database, you specified a server admin login with a username and password. Using these credentials, you can authenticate to any database on that server as the database owner.
- Azure AD authentication. This authentication method uses identities managed by Azure AD, and is supported for managed and integrated domains. Use Azure AD authentication (or integrated security) whenever possible. If you want to use Azure AD authentication, you must create another server admin called Azure AD admin. This admin is allowed to administer Azure AD users and groups, in addition to performing all operations that a regular server admin can. See **Use Azure Active Directory Authentication for authentication with SQL**[16] for a walkthrough of how to create an Azure AD admin to enable Azure Active Directory Authentication.

You can create user accounts in the master database, and grant permissions in all databases on the server, or you can create them in the database itself (called contained database users). By using contained databases, you obtain enhance portability and scalability.

As a best practice, your application should use a dedicated account to authenticate. This way, you can limit the permissions granted to the application and reduce the risks of malicious activity in case the application code is vulnerable to a SQL injection attack. The recommended approach is to create a contained database user, which allows your app to authenticate directly to the database. For more information, see **Contained Database Users - Making Your Database Portable**[17].

Cosmos DB authentication

Cosmos DB is Microsoft's globally distributed, horizontally partitioned, multi-model database service. The service is designed to allow customers to elastically (and independently) scale throughput and storage across any number of geographical regions. Cosmos DB offers guaranteed low latency at the 99th percentile, 99.99 percent high availability (which is a potential downtime of 1 minute and 0.5 seconds a week), predictable throughput, and multiple well-defined consistency models.

12 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-firewall-configure
13 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-firewall-configure
14 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-firewall-configure
15 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-firewall-configure
16 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-aad-authentication
17 https://docs.microsoft.com/en-us/sql/relational-databases/security/contained-database-users-making-your-database-portable?view=sql-server-2017

Cosmos DB is the first globally distributed database service in the industry to offer comprehensive service-level agreements (SLAs) encompassing all four dimensions of global distributions. It offers throughput, latency at the 99th percentile, availability, and consistency. As a cloud service, we have carefully designed and engineered Cosmos DB with multi-tenancy and global distribution in mind. More information on Cosmos DB is available at:

- **A technical overview of Azure Cosmos DB**[18].
- **Global data distribution with Azure Cosmos DB - overview**[19]

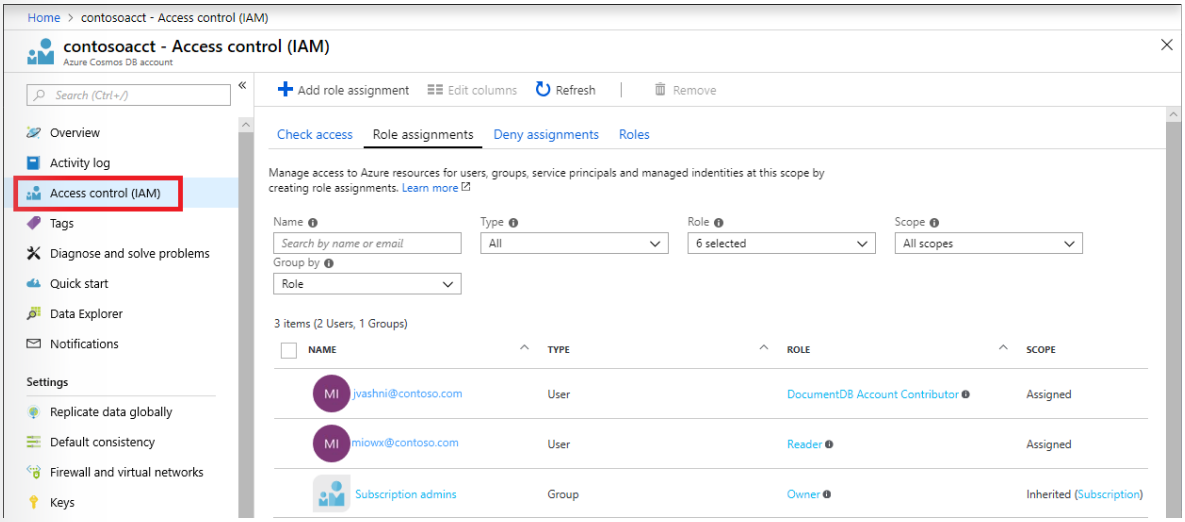
Cosmos DB authentication

Cosmos DB uses HMAC for authorization. Each request is hashed using the secret account key, and the subsequent base-64–encoded hash is sent with each call to Cosmos DB. To validate the request, the Cosmos DB service uses the correct secret key and properties to generate a hash. It then compares the value with the one in the request. If the two values match, the operation is authorized and the request is processed. If the values don't match, an authorization failure occurs, and the request is rejected.

You can use either a master key, or a resource token to allow fine-grained access to a resource such as a document. Using the master key, you can create user resources and permission resources per database. A resource token is associated with a permission in a database. It determines whether the user has access (read-write, read-only, or no access) to an application resource in the database such as container, documents, attachments, stored procedures, triggers, and UDFs. The resource token is then used during authentication to provide or deny access to the resource. For more information, see:

- **Master keys**[20]
- **Resource tokens**[21]

You can also provide access to the database account using Identity and Access Management (IAM) in the Azure portal. IAM provides role-based access control (RBAC), and integrates with Azure AD. You can use built-in roles or custom roles for individuals and groups, as in the following image.



Cosmos DB supports connections using connection strings generated using the MongoDB client. The following example creates a Cosmos DB and a connection string that MongoDB clients can use:

18 https://azure.microsoft.com/en-us/blog/a-technical-overview-of-azure-cosmos-db/
19 https://aka.ms/acdbglobaldist
20 https://docs.microsoft.com/en-us/azure/cosmos-db/secure-access-to-data
21 https://docs.microsoft.com/en-us/azure/cosmos-db/secure-access-to-data

```
az cosmosdb create --name <cosmosdb_name> --resource-group myResourceGroup
--kind MongoDB
```

The **–kind MongoDB** parameter enables MongoDB client connections.

Azure HDInsight database authentication

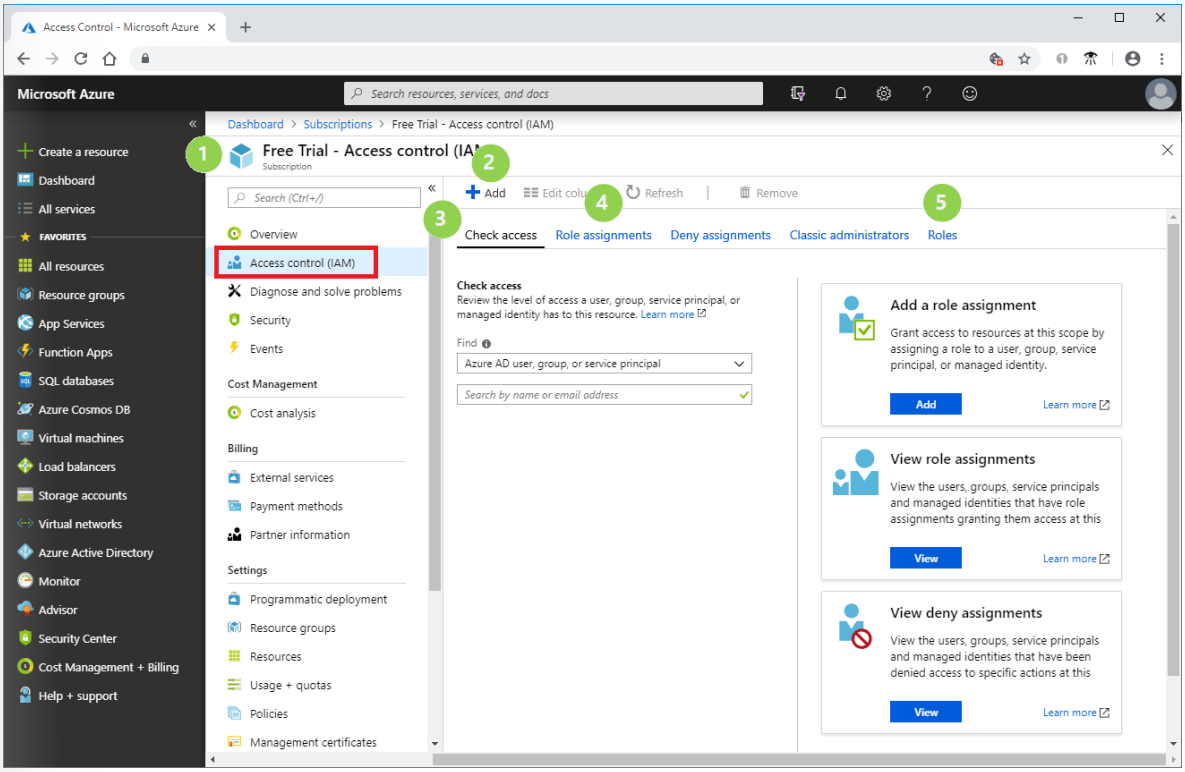
HDInsight is a fully managed, full-spectrum, open-source analytics service for enterprises. You can use several of the most popular open-source frameworks, including:

- Hadoop
- Apache Spark
- Apache Hive
- Hive LLAP
- Apache Kafka
- Apache Storm
- R

What is HDInsight and the Hadoop technology stack?

HDInsight is a cloud distribution of the Hadoop components from the Hortonworks Data Platform (HDP). HDInsight makes it easier, faster, and more cost-effective to process massive amounts of data. Utilizing open-source frameworks, you can enable a broad range of scenarios such as extract, transformation, and loading (ETL), data warehousing, machine learning, and the Internet of Things (IoT).

Because HDInsight is integrated into the Azure infrastructure, it uses the Azure authentication model of RBAC. Access control (IAM) appears in several locations in the Azure portal. The following screenshot is an example of the Access control (IAM) blade for a subscription.



The following table describes what some of the elements in the blade are used for.

Element	What you use it for
Resource where Access control (IAM) is opened	Identify scope (subscription in this example).
Add button	Add role assignments.
Check access tab	View the role assignments for a single user.
Role assignments tab	View the role assignments at the current scope.
Roles tab	View all roles and permissions.

Enable Azure AD authentication for SQL DB

This lesson applies to Azure SQL Database, and to both SQL Database and SQL Data Warehouse databases that are created on the Azure SQL Database.

Note: For simplicity, throughout this module the term SQL Database is used when referring to both SQL Database and SQL Data Warehouse.

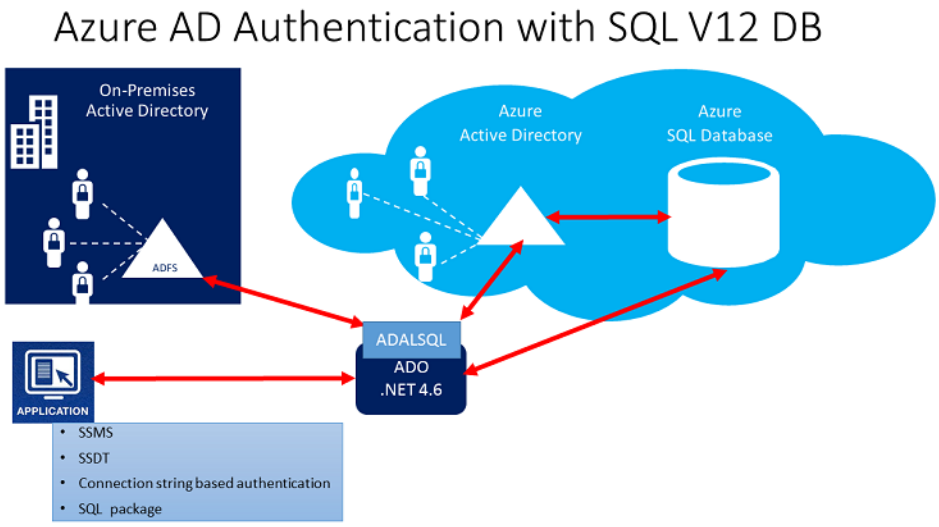
With Azure AD authentication, you can centrally manage the identities of database users and other Microsoft services in one central location. Central ID management provides a single place to manage database users and simplifies permissions management. There are many benefits to using this approach for authentication, such as stopping the proliferation of user identities across database servers.

The following high-level steps explain how to configure and use Azure Active Directory authentication:

1. Create and populate Azure AD.
2. Optional. Associate or change the active directory that is currently associated with your Azure Subscription.

3. Create an Azure AD administrator for the Azure SQL Database server, the Managed Instance, or the Azure SQL Data Warehouse.
4. Configure your client computers.
5. Create contained database users in your database mapped to Azure AD identities.
6. Connect to your database by using Azure AD identities.

The following high-level diagram summarizes the solution architecture for using Azure AD authentication with Azure SQL Database. The same concepts apply to SQL Data Warehouse. To support Azure AD native user passwords, only the cloud portion and Azure AD/Azure SQL Database is considered. To support Federated authentication (or user/password for Windows credentials), communication with Active Directory Federation Services (AD FS) block is required. The arrows indicate communication pathways.



Definitions for items in the diagram are as follows:

- SSMS. SQL Server Management Studio
- SSDT. SQL Server Data Tools
- ADALSQL. Active Directory Authentication Library for Microsoft SQL Server.
- ADO.net. ActiveX Data Object that is part of the .NET Framework

To learn how to create and populate Azure AD and then configure Azure AD with Azure SQL Database, Managed Instance, and SQL Data Warehouse, see **Configure and manage Azure Active Directory authentication with SQL**[22].

Enable database auditing

This lesson applies to Azure SQL Database, and the SQL Database and SQL Data Warehouse databases that are created on the Azure SQL Database.

You can use SQL database auditing to:

- Retain an audit trail of selected events. You can define categories of database actions to be audited.

22 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-aad-authentication-configure

- Report on database activity. You can use preconfigured reports and a dashboard to get started quickly with activity and event reporting.
- Analyze reports. You can find suspicious events, unusual activity, and trends.

Auditing policy

You can define an auditing policy for a specific database, or as a default server policy:

- A server policy applies to all existing and newly created databases on the server.
- If server blob auditing is enabled, it always applies to the database. The database will be audited regardless of the database auditing settings.
- Enabling blob auditing on the database or data warehouse—in addition to enabling it on the server—does not override or change any of the settings of the server blob auditing. Both audits will exist side by side. In other words, the database is audited twice in parallel; once by the server policy, and once by the database policy.

As a best practice, avoid enabling both server blob auditing and database blob auditing together, unless:

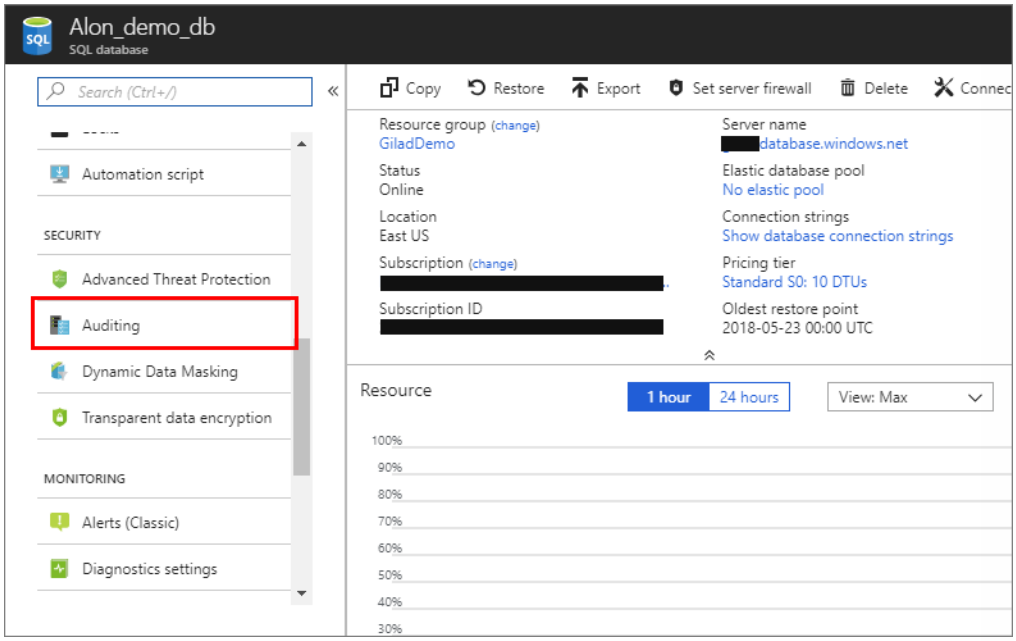
- You want to use a different storage account or retention period for a specific database.
- You want to audit event types or categories for a specific database that differs from the rest of the databases on the server. For example, you might have table inserts that need to be audited but only for a specific database.

Otherwise, we recommended that you enable only server-level blob auditing and leave the database-level auditing disabled for all databases.

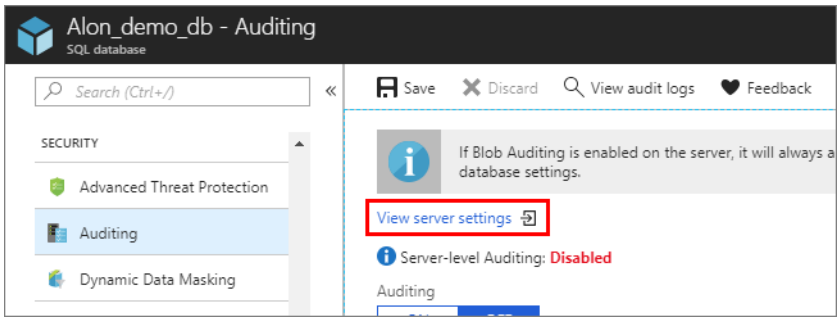
Exercise

The following high-level steps explain how to enable auditing on your database:

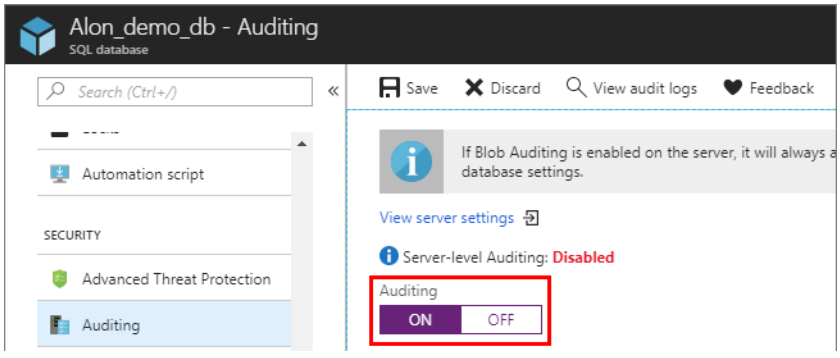
1. Sign in to the Azure portal.
2. Under Security, select Auditing.



1. If you prefer to set up a server auditing policy, you can select the View server settings link on the database auditing page. From here, you can view or modify the server auditing settings. Server auditing policies apply to all existing and newly created databases on this server.

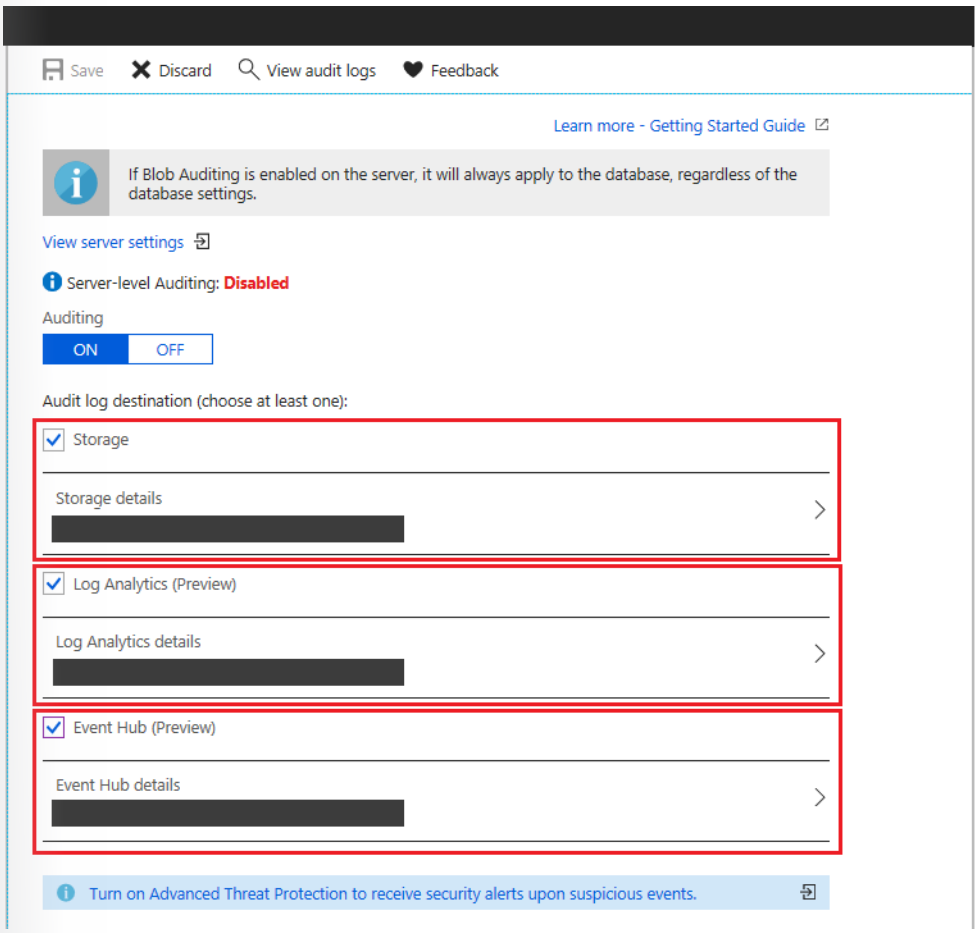


1. If you prefer to enable auditing on the database level, switch Auditing to ON.

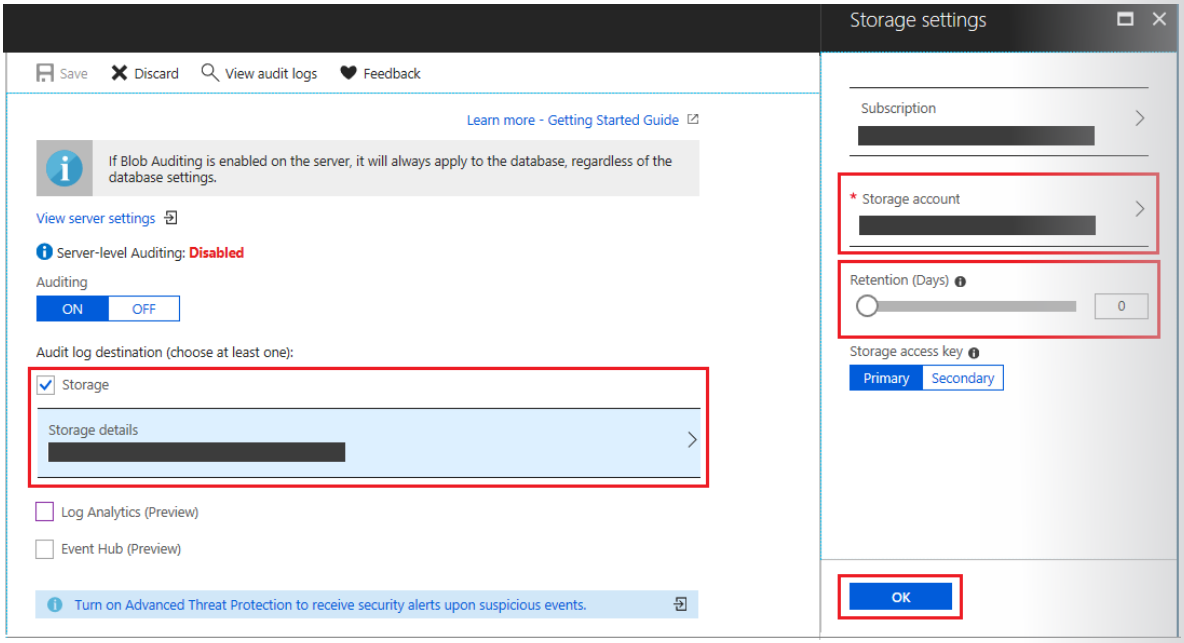


1. If server auditing is enabled, the database-configured audit will exist side-by-side with the server audit.

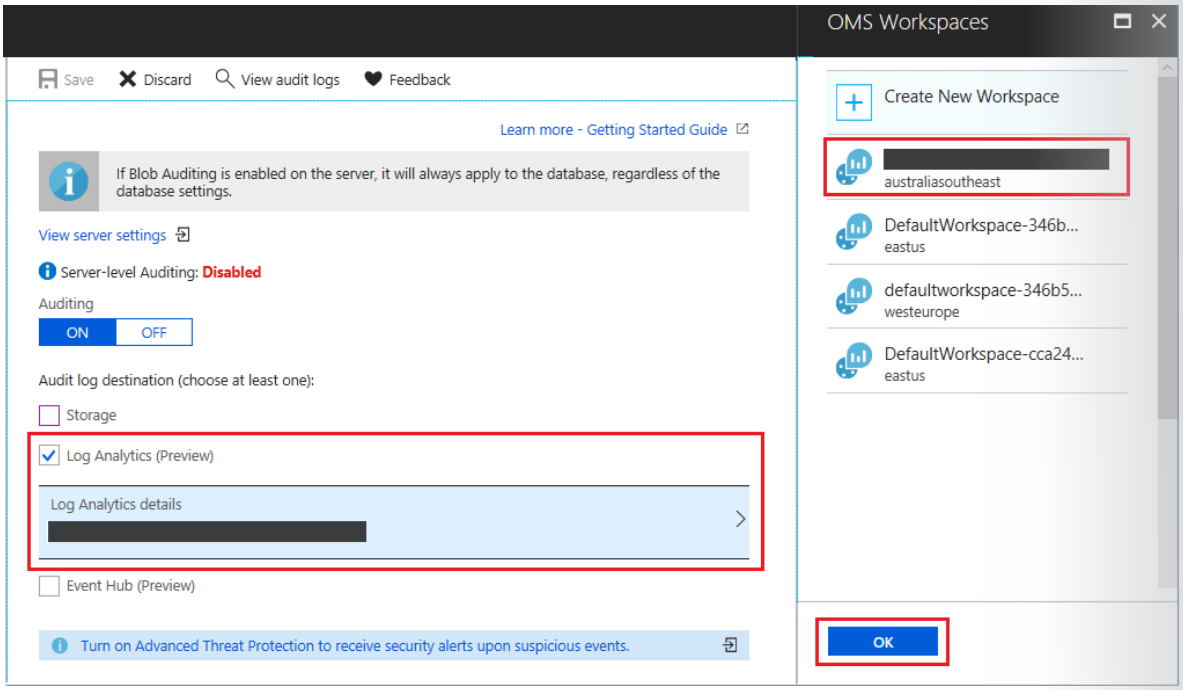
2. Notice that you can select for audit logs to be written to an Azure storage account, to a Log Analytics workspace for consumption by Azure Monitor logs, or to event hub for consumption using an event hub. You can configure any combination of these options, and audit logs will be written to each.



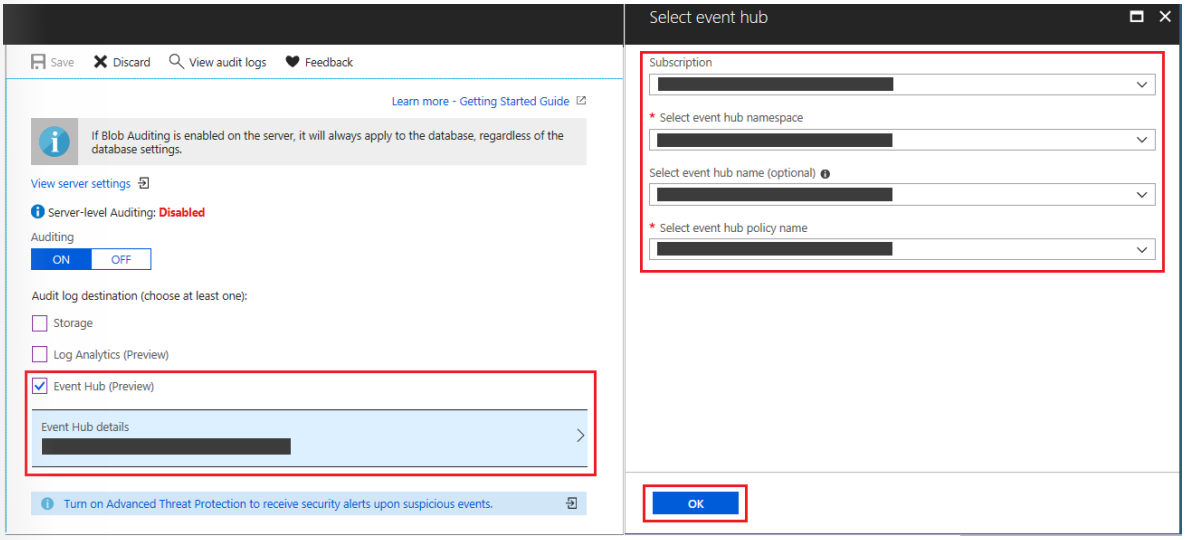
1. To configure writing audit logs to a storage account, select Storage and expand Storage details. Select the Azure storage account where logs will be saved, define the retention period, and then select OK. (Note that the old logs will be deleted.)



1. To configure writing audit logs to a Log Analytics workspace, select Log Analytics (Preview) and open Log Analytics details. Select or create the Log Analytics workspace where logs will be written to, and then select OK.



1. To configure writing audit logs to an event hub, select Event Hub (Preview) and open Event Hub details. Select the event hub where logs will be written to, and then select OK. (Be sure that the event hub is in the same region as your database and server.)



1. Select Save.

Note: If you want to customize the audited events, you can do this via PowerShell cmdlets or the REST API

- **Manage SQL database auditing using Azure PowerShell**[23]
- **Manage SQL database auditing using REST API**[24]

1. After you've configured your auditing settings, you can turn on the new threat detection feature and configure emails to receive security alerts. By using threat detection, you receive proactive alerts on anomalous database activities that can indicate potential security threats.

Exercise

For this exercise follow the instructions at **Analyze audit logs and reports**[25]. This exercise takes about 30 minutes.

Configure SQL DB threat detection

This lesson applies to Azure SQL Database, and to both SQL Database and SQL Data Warehouse databases that are created on the Azure SQL Database.

Threat detection provides a new layer of security that enables customers to detect and respond to potential threats as they occur, by providing security alerts on anomalous activities. Users receive an alert for suspicious database activities, potential vulnerabilities, SQL injection attacks, and anomalous database access and queries patterns. Threat detection integrates alerts with Azure Security Center, which includes details of suspicious activity and recommend action on how to investigate and mitigate the threat. Threat detection makes it simple to address potential threats to the database without the need to be a security expert or manage advanced security monitoring systems.

23 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-auditing
24 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-auditing
25 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-auditing

For a full investigation experience, enable SQL Database Auditing, which writes database events to an audit log in your Azure storage account. For more information about SQL Database Auditing, see **Get started with SQL database auditing**[26].

Threat detection for Azure SQL Database detects anomalous activities indicating unusual and potentially harmful attempts to access or exploit databases.

Threat detection is part of the **advanced data security**[27] (ADS) offering, which is a unified package for advanced SQL security capabilities. Threat detection can be accessed and managed via the central SQLbAdvanced Data Security portal.

Advanced data security provides a set of advanced SQL security capabilities, including data discovery and classification, vulnerability assessment, and threat detection:

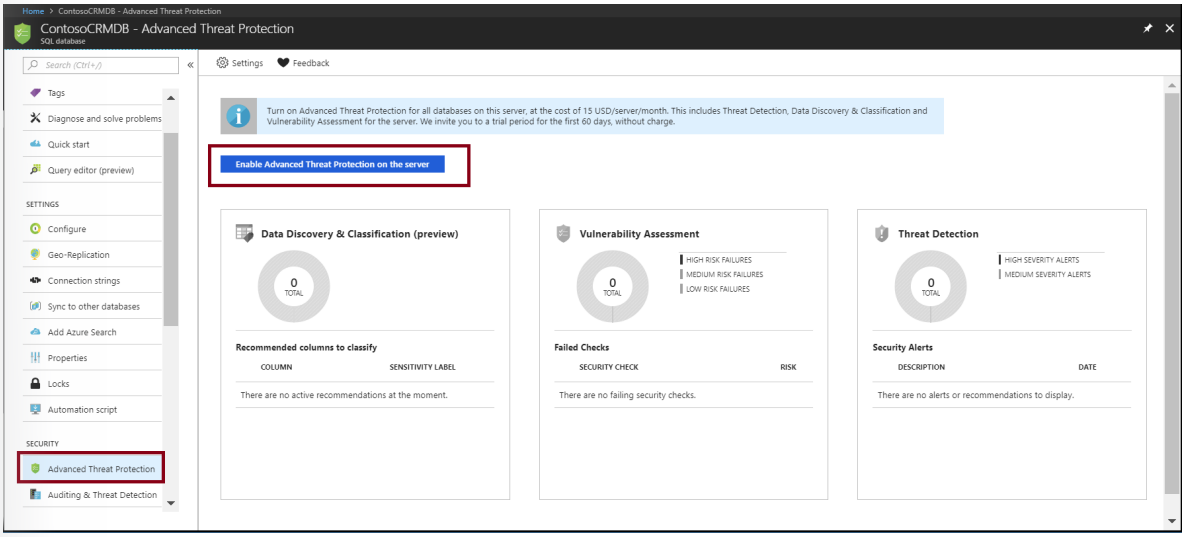
- Data discovery and classification (currently in preview) provides capabilities built into Azure SQL Database for discovering, classifying, labeling, and protecting the sensitive data in your databases. You can use it to provide visibility into your database classification state, and to track access to sensitive data within the database and beyond its borders.
- Vulnerability assessment is an easy-to-configure service that can discover, track, and help you remediate potential database vulnerabilities. It provides visibility into your security state, and includes actionable steps to resolve security issues and enhance your database fortifications.
- Threat detection detects unusual, anomalous activities indicating potentially harmful attempts to access or exploit your database. It continuously monitors your database for suspicious activities, and provides immediate security alerts on potential vulnerabilities, SQL injection attacks, and anomalous database access patterns. Threat detection alerts provide details of the suspicious activity and recommend action on how to investigate and mitigate the threat.

You only need to turn on SQL Advanced Data Security once to enable all of these included features for all databases on your SQL Database server or managed instance. Enabling or managing Advanced Data Security settings requires belonging to the SQL security manager role, SQL database admin role, or SQL server admin role.

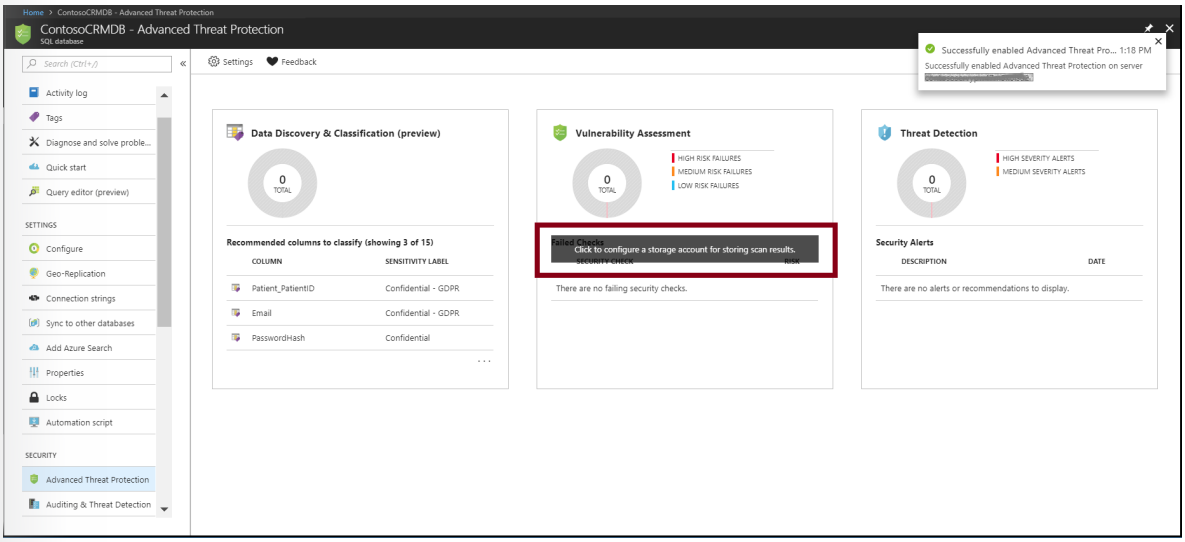
Enable SQL Advanced Data Security and threat detection

1. To enable Advanced Data Security, navigate to Advanced Data Security under the Security heading for your SQL Database server or managed instance. To enable Advanced Data Security for all databases on the database server or managed instance, select Enable Advanced Data Security on the server.

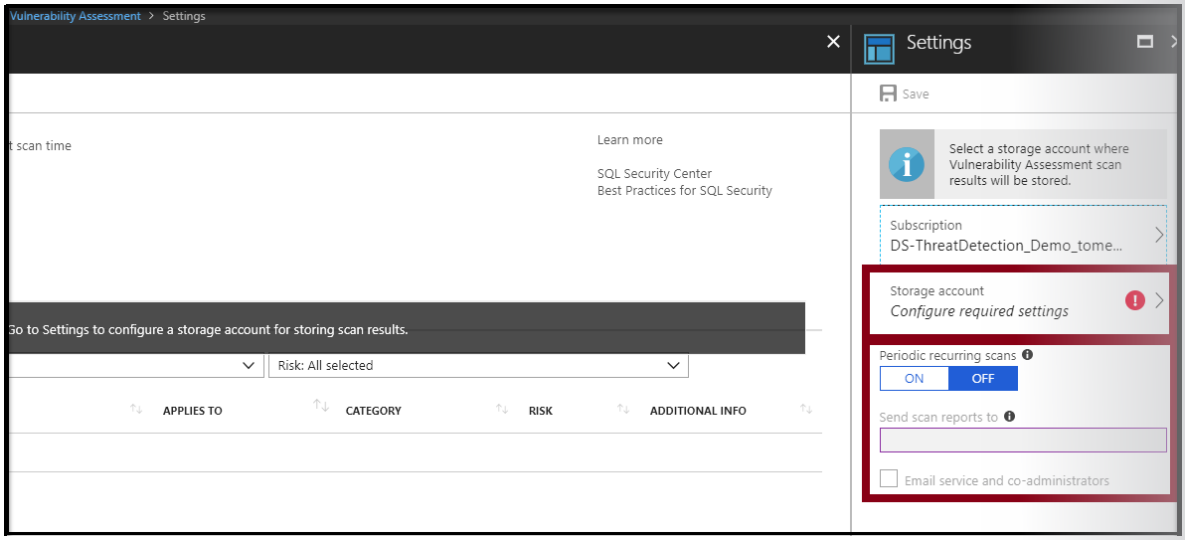
26 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-auditing
27 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-advanced-data-security



1. To start using vulnerability assessment, configure a storage account where scan results are saved. Start by selecting the vulnerability assessment card.



1. Next, select or create the storage account for saving scan results. From here you can also turn on periodic recurring scans to configure vulnerability assessment to run automatic scans once per week. A scan result summary is sent to the email address (or addresses) you provide.

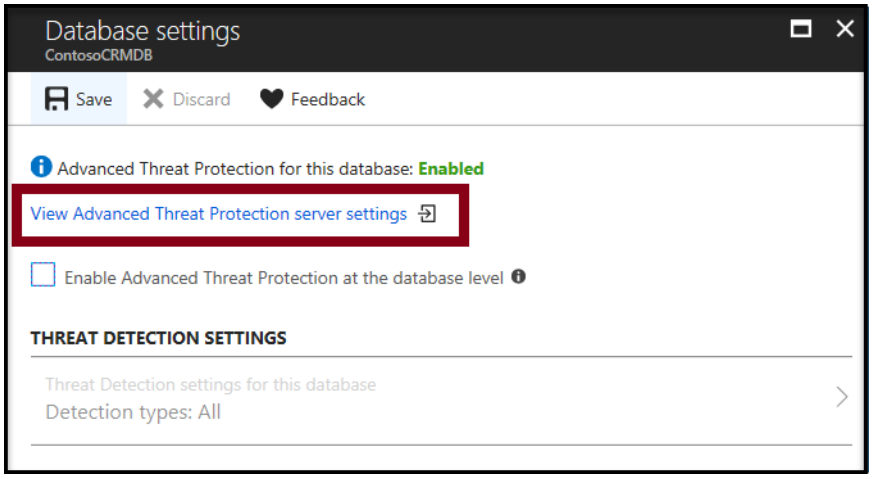


1. Start classifying data, tracking vulnerabilities, and investigating threat alerts by selecting the Data Discovery and Classification card. This will display recommended sensitive columns to classify, and to classify your data with persistent sensitivity labels.
2. Select the Vulnerability Assessment card to view and manage vulnerability scans and reports, and to track your security status. If security alerts have been received, select the Threat Detection card to view details of the alerts and to see a consolidated report on all alerts in your Azure subscription, via the Azure Security Center security alerts page.

For more information, see:

- **Manage Advanced Data Security settings on your SQL Database server or managed instance**[28].
- **Manage Advanced Data Security settings for a SQL database**[29]

1. You can also access Advanced data security settings for your database server or managed instance from the Advanced Data Security database blade. In the main Advanced Data Security blade, select Settings, and then select View Advanced Data Security server settings.



28 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-advanced-data-security
29 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-advanced-data-security

Summary

A threat as defined in the security world is an external force that could exploit a vulnerability. Not all threats have malevolent intentions though. For example, a storm that knocks out power to a datacenter can deny customers service, but there is no malicious intent. Instead, the storm is a vulnerability that the service provider should account for with a backup power system. The risk to this service is the intersection of threat and vulnerability. This risk–threat–vulnerability relationship is often defined as:

vulnerability x threat = risk

Knowing the threat for your database service will allow you to reduce the associated vulnerability, thereby reducing risk.

Configure access control for storage accounts

Every request made against a secured resource in the Blob, File, Queue, or Table service must be authorized. Authorization ensures that resources in your storage account are accessible only when you want them to be, and only to those users or applications to whom you grant access.

Options for authorizing requests to Azure Storage include:

- Azure AD (Preview). Azure AD is the Microsoft cloud-based identity and access management service. Azure AD integration is currently available in preview for the Blob and Queue services. With Azure AD, you can assign fine-grained access to users, groups, or applications via RBAC. For more information about Azure AD integration with Azure Storage, see **Authenticate with Azure Active Directory (Preview)**[30].
- Shared Key. Shared Key authorization relies on your account access keys and other parameters to produce an encrypted signature string that is passed on via the request in the Authorization header. For more information about Shared Key authentication, see **Authorize with Shared Key**[31].
- Shared Access Signatures. A Shared Access Signature (SAS) delegates access to a particular resource in your account with specified permissions and over a specified time interval. For more information about SAS, see **Delegating Access with a Shared Access Signature**[32].
- Anonymous access to containers and blobs. You can optionally make blob resources public at the container or blob level. A public container or blob are accessible to any user for anonymous read access. Read requests to public containers and blobs do not require authorization. For more information, see **Manage anonymous read access to containers and blobs**[33]

Authorizing applications that access Azure Storage using Azure AD provides better security and ease of use over other authorization options. While you can continue to use shared key authorization with your applications, using Azure AD circumvents the need to store your account access key with your code. Similarly, while you could continue to use SAS to grant fine-grained access to resources in your storage account, Azure AD offers similar capabilities without the need to manage SAS tokens or worry about revoking a compromised SAS. For more information about Azure AD integration in Azure Storage, see **Authenticate access to Azure blobs and queues using Azure Active Directory (Preview)**[34].

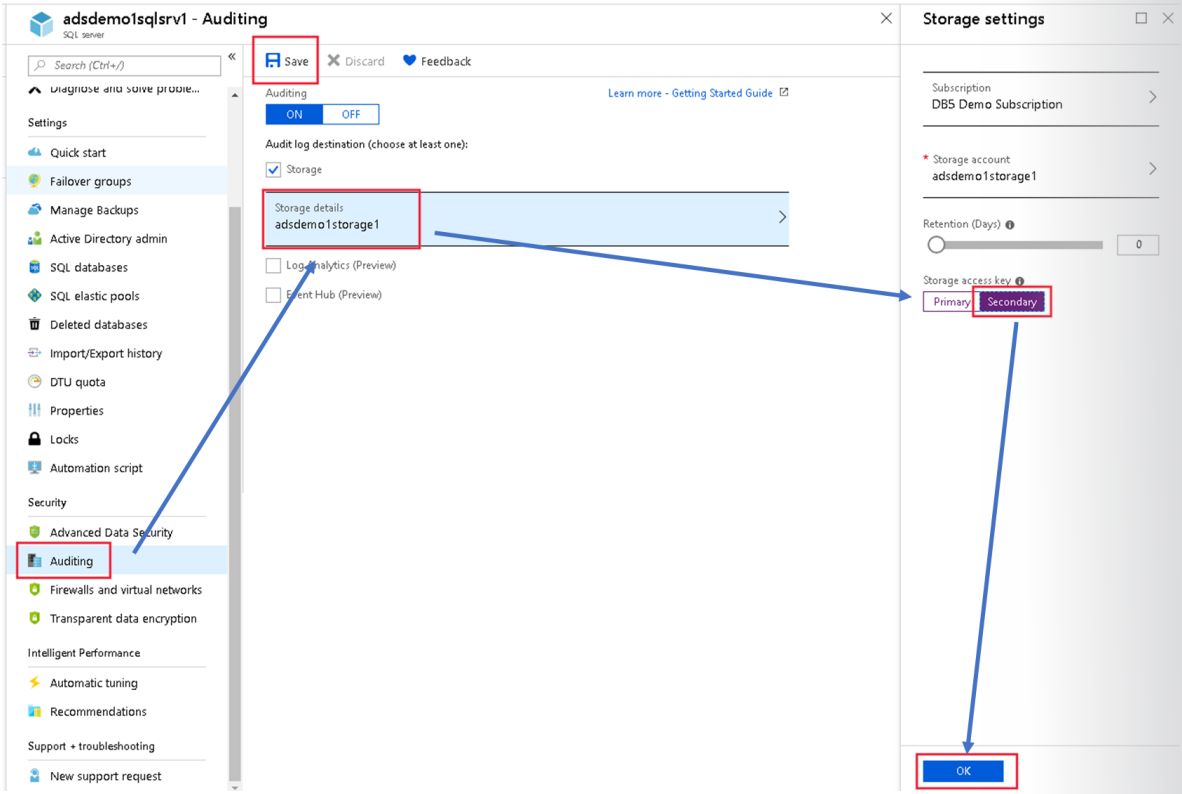
30 https://docs.microsoft.com/en-us/rest/api/storageservices/authenticate-with-azure-active-directory
31 https://docs.microsoft.com/en-us/rest/api/storageservices/authorize-with-shared-key
32 https://docs.microsoft.com/en-us/rest/api/storageservices/delegating-access-with-a-shared-access-signature
33 https://docs.microsoft.com/azure/storage/blobs/storage-manage-access-to-resources
34 https://docs.microsoft.com/azure/storage/common/storage-auth-aad

Configure key management for storage accounts

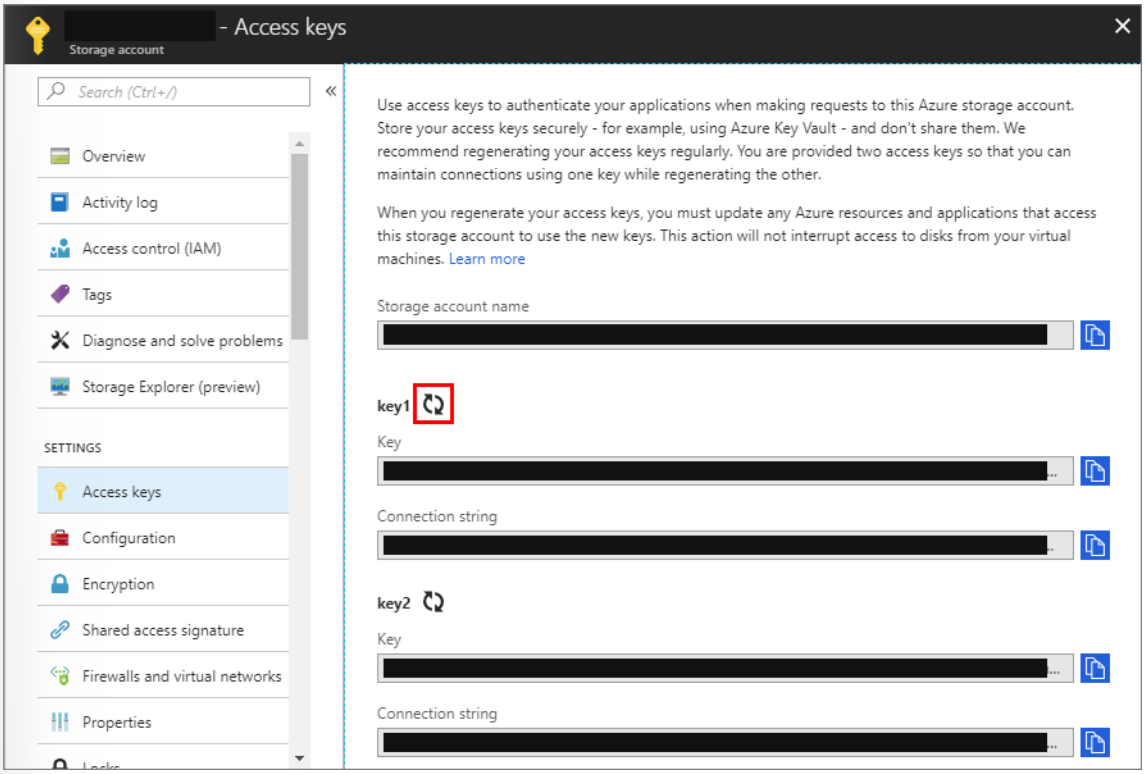
In production, you are likely to refresh your storage keys periodically. When writing audit logs to Azure storage, you need to resave your auditing policy when refreshing your keys.

The following steps detail how to refresh your storage keys and resave your auditing policy:

1. Open the Storage Details blade. In the Storage settings blade, under Storage Access Key, select Secondary, select OK, and then select Save.



1. On the Access keys blade, select the key icon to regenerate the primary key.



1. Return to the Auditing Configuration page, switch the storage access key from secondary to primary, select OK, and then select Save.
2. Return to the Storage Configuration page, and regenerate the secondary access key (in preparation for the next key's refresh cycle).

Create and manage Shared Access Signatures (SAS)

Every time you access data in your storage account, your client makes a request over HTTP/HTTPS to Azure Storage. Every request to a secure resource must be authorized so that the service ensures that the client has the permissions required to access the data.

Azure Storage offers the following options for authorizing access to secure resources:

- Azure AD integration (Preview) for blobs and queues. Azure AD provides RBAC for fine-grained control over a client's access to resources in a storage account.
- Shared Key authorization for blobs, files, queues, and tables. A client using Shared Key passes a header with every request that is signed using the storage account access key.
- SAS for blobs, files, queues, and tables. SAS provides limited delegated access to resources in a storage account. Adding constraints on the time interval for which the signature is valid or on permissions it grants provides flexibility in managing access.
- Anonymous public read access for containers and blobs. This option does not require authorization. For more information, see **Manage anonymous read access to containers and blobs**[35].

35 https://docs.microsoft.com/azure/storage/blobs/storage-manage-access-to-resources

By default, all resources in Azure Storage are secured, and are available only to the account owner. Although you can use any of the authorization strategies outlined above to grant clients access to resources in your storage account, we recommend using Azure AD whenever possible for maximum security and ease of use.

An SAS provides delegated access to resources in your storage account. With an SAS, you can grant clients access to resources in your storage account, without sharing your account keys, which is the key point of using SAS in your applications—an SAS is a secure way to share your storage resources without compromising your account keys.

Your storage account key is similar to the root password for your storage account. Always be careful to protect it. Avoid distributing your storage account key to other users, hard-coding it, or saving it anywhere in plaintext that is accessible to others. If you believe your account key might have been compromised, regenerate it using the Azure portal.

Just like your account access keys, you must protect your SAS tokens as well. While providing granularity, SAS grants clients access to the resources in your storage account and should not be shared publicly. When sharing is required for troubleshooting reasons, consider using a redacted version of any log files or deleting the SAS tokens (if present) from the log files. Also, make sure the screenshots don't contain any SAS information either.

An SAS gives you granular control over the type of access you grant to clients who have the SAS, including:

- The interval over which the SAS is valid, including the start time and the expiry time.
- The permissions granted by the SAS. For example, an SAS for a blob might grant read and write permissions to that blob, but not delete permissions.
- An optional IP address or range of IP addresses from which Azure Storage will accept the SAS. For example, you might specify a range of IP addresses belonging to your organization.
- The protocol over which Azure Storage will accept the SAS. You can use this optional parameter to restrict access to clients using HTTPS.

You can create two types of SAS's:

- Service SAS. The service SAS delegates access to a resource in just one of the storage services: Blob, Queue, Table, or File service. For in-depth information about constructing the service SAS token, see:
- **Constructing a Service SAS**[36]
- **Service SAS Examples**[37]
- Account SAS. The account SAS delegates access to resources in one or more of the storage services. All of the operations available via a service SAS are also available via an account SAS. Additionally, with the account SAS, you can delegate access to operations that apply to a given service, such as Get/Set Service Properties and Get Service Stats. You can also delegate access to read, write, and delete operations on blob containers, tables, queues, and file shares that are not permitted with a service SAS.

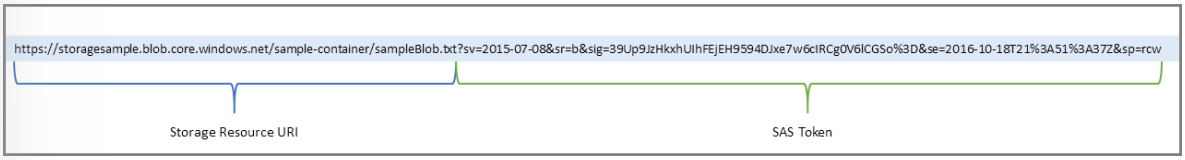
How SAS works

An SAS is a signed Uniform Resource Identifier (URI) that points to one or more storage resources and includes a token that contains a special set of query parameters. The token indicates how the client can access the resources. One of the query parameters—the signature—is constructed from the SAS parame-

36 https://docs.microsoft.com/en-us/rest/api/storageservices/constructing-a-service-sas
37 https://docs.microsoft.com/en-us/rest/api/storageservices/service-sas-examples

ters and signed with the account key. Azure Storage uses this signature to authorize access to the storage resource.

This example is of an SAS URI, breaking out the resource URI and the SAS token.



The SAS token is a string you generate on the client side. As an example, an SAS token that you generate with the storage client library is not tracked by Azure Storage in any way. You can create an unlimited number of SAS tokens on the client side.

When a client provides an SAS URI to Azure Storage as part of a request, the service checks the SAS parameters and signature to verify that it is valid for authenticating the request. If the service verifies that the signature is valid, then the request is authorized. Otherwise, the request is declined with error code 403 (Forbidden).

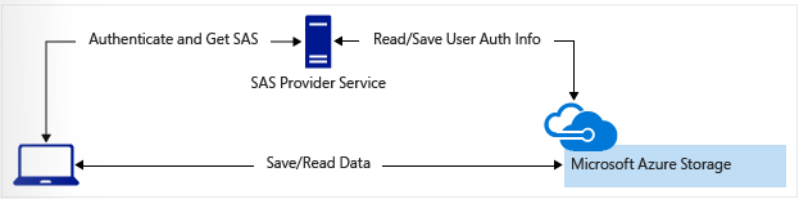
When to use an SAS

You can use an SAS when you want to provide access to resources in your storage account to any client not possessing your storage account's access keys. Your storage account includes both a primary and secondary access key, both of which grant administrative access to your account, and all resources within it. Exposing either of these keys opens your account to the possibility of malicious or negligent use. Conversely, SAS provides a safe alternative that allows clients to read, write, and delete data in your storage account according to the permissions you've explicitly granted, and without need for an account key.

A common scenario where an SAS is useful is a service where users read and write their own data to your storage account. For this scenario, there are two typical design patterns:

- Clients upload and download data via a front-end proxy service that performs authentication. This front-end proxy service has the advantage of allowing validation of business rules, but for large amounts of data or high-volume transactions, creating a service that can scale to match demand could be expensive or difficult.

- Alternatively, a lightweight service authenticates the client as needed and then generates an SAS. After the client receives the SAS, they can access storage account resources directly using the permissions defined by the SAS and the interval allowed by the SAS. The SAS mitigates the need for routing all data through the front-end proxy service.



Many real-world services use a hybrid of these two approaches. For example, some data might be processed and validated via the front-end proxy, while other data is saved and read (or just read) directly using SAS.

Summary

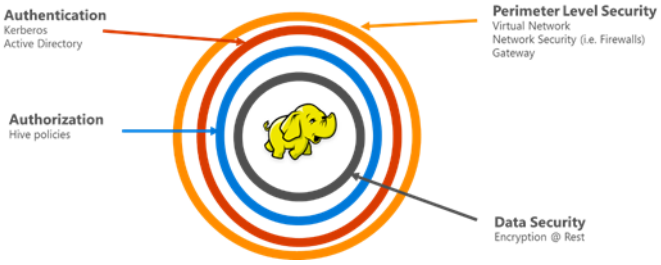
SAS is useful for providing limited permissions to your storage account to clients that should not have the account key. As such, they are a vital part of the security model for any application using Azure Storage. You can use SAS to provide greater flexibility of access to resources in your storage account without compromising the security of your application.

Configure security for HDInsight

An enterprise admin can create an add-on (optional) Enterprise Security Package cluster that's joined to an Active Directory domain, and then configure a list of employees from the enterprise who can authenticate through Azure AD to sign in to the HDInsight cluster. Using this method, no one from outside the enterprise can sign in or access the HDInsight cluster.

The enterprise admin can configure RBAC for Apache Hive security by using Apache Ranger by Hortonworks. Configuring RBAC restricts data access to only what's needed. Finally, the admin can audit the data access by employees and make any changes to the access control policies. The admin can then achieve a higher degree of governance over their corporate resources.

Enterprise security contains four major pillars: perimeter security, authentication, authorization, and encryption.



Perimeter security

You can achieve stronger perimeter security in HDInsight through virtual networks and the Azure VPN Gateway service. An enterprise admin can create an Enterprise Security Package cluster inside a virtual network and use network security groups (firewall rules) to restrict access to the virtual network. Only the IP addresses defined in the inbound firewall rules will be able to communicate with the HDInsight cluster. This configuration provides perimeter security.

Another layer of perimeter security is achieved through the VPN Gateway service. The gateway acts as first line of defense for any incoming request to the HDInsight cluster. It accepts the request, validates it, and only then allows the request to pass to the other nodes in cluster. In this way, the VPN Gateway service provides perimeter security to other name and data nodes in the cluster.

Authentication

An enterprise admin can create an HDInsight cluster with Enterprise Security Package in a virtual network. All the nodes of the HDInsight cluster are joined to the domain that the enterprise manages. This is

achieved through the use of Azure Active Directory Domain Services (Azure AD Domain Services). You can read more about this at **Azure Active Directory Domain Services**[38].

With this setup, enterprise employees can sign in to the cluster nodes by using their domain credentials. They can also use their domain credentials to authenticate and interact with the cluster via other approved endpoints such as:

- Apache Ambari View
- Microsoft Open Database Connectivity (ODBC)
- Java Database Connectivity (JDBC)
- PowerShell,
- REST APIs

The admin has full control over limiting the number of users who interact with the cluster via these endpoints.

Authorization

A best practice that most enterprises is ensuring that not every employee has access to all enterprise resources. Likewise, the admin can define RBAC policies for the cluster resources.

For example, the admin can configure Apache Ranger to set access control policies for Hive. This functionality ensures that employees can access only as much data as they need, to be successful in their jobs. SSH access to the cluster is also restricted to only the administrator.

Encryption

Protecting data is important for meeting organizational security and compliance requirements. Along with restricting access to data from unauthorized employees, as a best practice you should also encrypt it.

Both data stores for HDInsight clusters—Azure Blob storage and Azure Data Lake Storage Gen1/Gen2—support transparent server-side encryption of data at rest. Secure HDInsight clusters will seamlessly work with this server-side encryption capability of data at rest.

For more information, see:

- **Configure a HDInsight cluster with Enterprise Security Package by using Azure Active Directory Domain Services**[39]
- **Configure Apache Hive policies in HDInsight with Enterprise Security Package**[40]

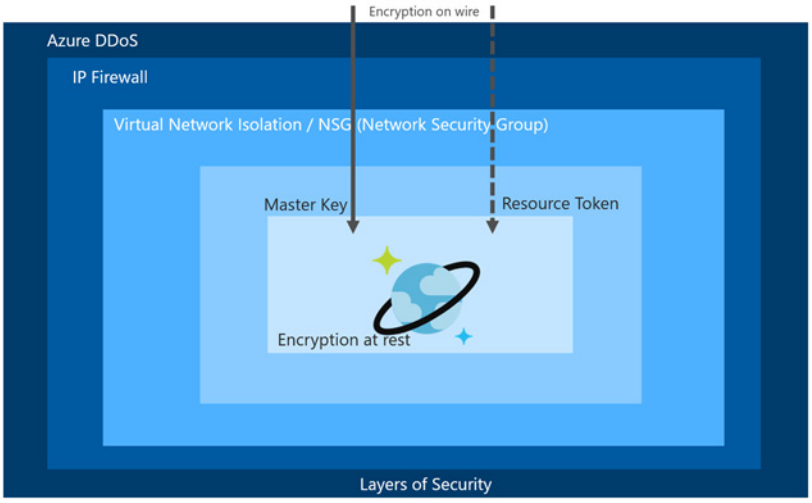
Configure security for Cosmos DB

The following descriptions of Cosmos DB security are from Rafat Sarosh, Principle PM for Cosmos DB.

Azure Cosmos DB is a ring zero Azure service. This means it will be available in any new Azure data center as soon as it goes online and must keep all its compliance certificates current. Azure Cosmos DB has a plethora of certifications that you can read more about in the blog post "**Azure #CosmosDB: Secure, private, compliant**[41]."

38 https://docs.microsoft.com/en-us/azure/active-directory-domain-services/active-directory-ds-overview
39 https://docs.microsoft.com/en-us/azure/hdinsight/domain-joined/apache-domain-joined-configure
40 https://docs.microsoft.com/en-us/azure/hdinsight/domain-joined/apache-domain-joined-run-hive
41 https://azure.microsoft.com/en-us/blog/azure-cosmosdb-secure-private-compliant/

Layers of Security



The first layer of Azure provides physical safety of data centers and continuous protections from distributed denial of service (DDoS) attacks. Azure has dedicated teams that continuously monitor security issues. All Azure services run a common security agent to collect anomalous activity. Production resources are patched regularly, and all secrets, certificates, and passwords have a defined lifetime. These secrets and certificates should be rotated after they expire.

All production ports in Cosmos DB are scanned and penetration-tested regularly. The source code is scanned for security issues, and require two approvers before integrating into the product.

Access to Azure is restricted. Operations staff or developers working for Cosmos DB cannot access any production from their machines. Instead, all production is accessed via dedicated secure access workstations. These workstations are watertight; there is no outside access to or from these machines unless it is through Azure.

Engineers get just-in-time (JIT) approval for all production access, which is monitored, and every activity of an engineer is monitored by an escort. All production deployments require multiple approvals including test sign out and approvers.

Azure has a very strict access policy by Microsoft personnel. It's almost impossible for any Microsoft employee to access the production system if they are not authorized

Security offered by an IP firewall

Using an IP firewall is the first layer of protection to secure your database. Cosmos DB supports policy-driven IP-based access controls for inbound firewall support. This model is similar to the firewall rules of a traditional database system, and provides an additional level of security to the Cosmos DB account. With this model, you can configure a Cosmos DB account to be accessible only from an approved set of machines and cloud services. Access to Cosmos DB resources from these approved sets of machines and services still require the caller to present a valid authorization token.

You can set the IP access control policy in the Azure portal, or programmatically through Azure CLI, PowerShell, or the REST API by updating the ipRangeFilter property. For additional information on how to set up an IP firewall for Cosmos DB, visit **IP firewall in Azure Cosmos DB**[42].

42 https://docs.microsoft.com/en-us/azure/cosmos-db/firewall-support

Security offered by a virtual network

Virtual network is the next layer that secures Cosmos DB accounts. You can configure your Cosmos DB account to allow access only from a specific subnet of Azure Virtual Networks. By enabling a service endpoint for Cosmos DB from a virtual network and its subnet, traffic is provided with an optimal and secure route to Cosmos DB.

After a Cosmos DB account is configured with a virtual network service endpoint, it can be accessed only from the specified subnet and the public or internet access is removed. To learn in detail about service endpoints, refer to the Azure **Virtual Network Service Endpoints**[43] overview article. You can also learn more about virtual network and Cosmos DB at **Access Azure Cosmos DB resources from virtual networks (VNet**[44]).

You can filter network traffic to and from Azure resources in an Azure Virtual Network with a network security group. A network security group contains security rules that allow or deny both inbound and outbound network traffic from several types of Azure resources.

Access control with keys

The security layers discussed so far come as part of Cosmos DB and don't require much more than configuring the firewall and virtual network. Now, let's discuss what more you can do as an application developer.

All access to Cosmos DB is controlled by two keys: a master key and a read-only key. Master key, as its name implies is a master key and can do all operations on Cosmos DB. A read-only key enables you to read the data, but no other actions are possible with this key.

Keys should always be kept in a key vault. Your application or users can have managed identities registered with the key vault, and they can get the keys at the run time.

Cosmos DB keeps all your data encrypted at rest and on wire. With virtual network, IP filtering, and key vault you can built a more secure application with Cosmos DB.

Configure security for Azure Data Lake

Data Lake Storage Gen2 is the result of converging the capabilities of our two existing storage services: Azure Blob storage and Azure Data Lake Storage Gen1. Features from Azure Data Lake Storage Gen1 (such as file system semantics, directory, and file-level security and scale) are combined with low-cost, tiered storage, and high availability disaster recovery capabilities from Azure Blob storage.

A fundamental part of Data Lake Storage Gen2 is the addition of a hierarchical namespace to Blob storage. The hierarchical namespace organizes objects/files into a hierarchy of directories for efficient data access.

In this lesson we focus on the improved security of Data Lake Storage Gen 2, a superset of POSIX permissions. The security model for Data Lake Gen2 supports ACL and Portable Operating System Interface for UNIX (POSIX) permissions along with some extra granularity specific to Data Lake Storage Gen2. You configure settings through Storage Explorer, or through frameworks such as Hive and Spark.

Security features of Data Lake Storage Gen 2 are:

- All data written to Azure Storage is automatically encrypted using Storage Service Encryption (SSE).

43 https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview
44 https://docs.microsoft.com/en-us/azure/cosmos-db/vnet-service-endpoint

- Azure AD and RBAC are supported for Azure Storage for both resource management operations and data operations, as follows:
- You can assign RBAC roles scoped to the storage account to security principals, and use Azure AD to authorize resource management operations such as key management.
- Azure AD integration is supported for data operations on Azure Storage. You can assign RBAC roles scoped to a subscription, resource group, storage account, or an individual filesystem to a security principal or a managed identity for Azure resources. .
- You can grant delegated access to the data objects in Azure Storage using SAS.

As discussed in a previous lesson, the management plane's resources are used to manage and secure your storage account. While the data plane secures access to your data.

Management plane security

The management plane consists of operations that affect the storage account itself. For example, you can create or delete a storage account, get a list of storage accounts in a subscription, retrieve the storage account keys, or regenerate the storage account keys.

Use the Resource Manager model of deployment, which is the means for creating storage accounts with Data Lake Storage Gen2 capabilities. With the Resource Manager storage accounts, rather than giving access to the entire subscription, you can control access on a more finite level to the management plane using RBAC.

Data Plane Security

Data plane security refers to the methods used to secure the data objects stored in Azure Storage. We've seen methods to encrypt the data, and security during transit of the data, but how do you go about controlling access to the objects?

There are three options for authorizing access to data objects in Azure Storage:

- Use Azure AD to authorize access to filesystems and queues. Azure AD provides advantages over other approaches to authorization, including removing the need to store secrets in your code.
- Use your storage account keys to authorize access via Shared Key. Authorizing via Shared Key requires storing your storage account keys in your application, so we recommend using Azure AD instead where possible. For production applications or for authorizing access to Azure tables and files, continue using Shared Key while Azure AD integration (Preview).
- Use SAS to grant controlled permissions to specific data objects for a specific amount of time.

In addition to limiting access through authorization, you can also use firewalls and virtual networks to limit access to the storage account based on network rules. This approach enables you deny access to public internet traffic, and to grant access only to specific Azure Virtual Networks or public internet IP address ranges.

For more information, see:

- **How to secure your storage account with Role-Based Access Control (RBAC)**[45]
- **Manage access to Azure resources using RBAC and the Azure portal**[46]

45 https://docs.microsoft.com/en-us/azure/storage/common/storage-data-lake-storage-security-guide?toc=%2fazure%2fstorage%2fblobs%2ftoc.json

46 https://docs.microsoft.com/en-us/azure/role-based-access-control/role-assignments-portal

- **Built-in roles for Azure resources**[47]
- **Storage Account Keys**[48]

47 https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles
48 https://docs.microsoft.com/en-us/azure/storage/common/storage-data-lake-storage-security-guide?toc=%2fazure%2fstorage%2fblobs%2ftoc.json

Configure Encryption for Data at Rest

Implement SQL Database Always Encrypted

In this lesson, we address how to secure sensitive data in a SQL database with data encryption using the Always Encrypted Wizard in SQL Server Management (SSMS).

Always Encrypted is a new data encryption technology in Azure SQL Database and SQL Server that helps protect sensitive data at rest on the server, during movement between client and server, and while the data is in use. Always Encrypted ensures that sensitive data never appears as plaintext inside the database system. After you configure data encryption, only client applications or app servers that have access to the keys can access plaintext data.

Exercise

In this exercise, you will follow the steps to learn how to set up Always Encrypted for an Azure SQL database.

More specifically, you will learn how to perform the following tasks:

- Use the Always Encrypted wizard in SSMS to create Always Encrypted keys.
  - Create a column master key
  - Create a column encryption key
- Create a database table and encrypt columns
- Create an application that inserts, selects, and displays data from the encrypted columns

For this exercise, you'll need:

- An Azure account and subscription (If you don't have one, sign up for a free trial.)
- SQL Server Management Studio version 13.0.700.242 or later
- .NET Framework 4.6 or later (on the client computer)
- Visual Studio
- Azure PowerShell

Follow the steps at **Enable your client application to access the SQL Database service**[49]. The exercise will take about 1.5 hours.

For more information, review the following pages:

- **Always Encrypted Wizard**[50]
- **SQL Server Management Studio (SSMS)**[51]
- **Always Encrypted (Database Engine)**[52]
- **CREATE COLUMN MASTER KEY (Transact-SQL)**[53]
- **CREATE COLUMN ENCRYPTION KEY (Transact-SQL)**[54]

49 https://docs.microsoft.com/en-us/azure/sql-database/sql-database-always-encrypted-azure-key-vault
50 https://msdn.microsoft.com/library/mt459280.aspx
51 https://msdn.microsoft.com/library/hh213248.aspx
52 https://msdn.microsoft.com/library/mt163865.aspx
53 https://msdn.microsoft.com/library/mt146393.aspx
54 https://msdn.microsoft.com/library/mt146372.aspx

- **Download SQL Server Management Studio (SSMS)**[55]
- **.NET Framework Guide**[56]
- **Downloads: Visual Studio**[57]
- **Overview of Azure PowerShell**[58]

Implement database encryption

Always Encrypted is a feature designed to protect sensitive data such as credit card numbers or national identification numbers (for example, U.S. social security numbers), stored in Azure SQL Database or SQL Server databases. Always Encrypted allows clients to encrypt sensitive data inside client applications and never reveal the encryption keys to the Database Engine (SQL Database or SQL Server). As a result, Always Encrypted provides a separation between those who own the data (and can view it) and those who manage the data (but should have no access). By ensuring that on-premises database administrators, cloud database operators, or other high-privileged, but unauthorized users cannot access the encrypted data, Always Encrypted enables customers to confidently store sensitive data outside of their direct control. This allows organizations to encrypt data at rest and in use for storage in Azure, to enable delegation of on-premises database administration to third parties, or to reduce security clearance requirements for their own database administrator (DBA) staff.

Always Encrypted makes encryption transparent to applications. An Always Encrypted–enabled driver installed on the client computer achieves this by automatically encrypting and decrypting sensitive data in the client application. The driver encrypts the data in sensitive columns before passing the data to the Database Engine, and automatically rewrites queries so that the semantics to the application are preserved. Similarly, the driver transparently decrypts data stored in encrypted database columns contained in query results.

Configuring Always Encrypted

The initial setup of Always Encrypted in a database involves generating Always Encrypted keys, creating key metadata, configuring encryption properties of selected database columns, and/or encrypting data that might already exist in columns that need to be encrypted. Some of these tasks are not supported in Transact-SQL and require the use of client-side tools.

As Always Encrypted keys and protected sensitive data are never revealed in plaintext to the server, the Database Engine cannot be involved in key provisioning and perform data encryption or decryption operations. You can use SSMS or PowerShell to accomplish such tasks.

Task	SSMS	PowerShell	T-SQL
Provision column master keys, column encryption keys and encrypted column encryption keys with their corresponding column master keys	Yes	Yes	No
Create key metadata in the database	Yes	Yes	Yes

55 https://msdn.microsoft.com/library/mt238290.aspx
56 https://msdn.microsoft.com/library/w0x726c2.aspx
57 https://www.visualstudio.com/downloads/download-visual-studio-vs.aspx
58 https://docs.microsoft.com/en-us/powershell/azure/overview

Task	SSMS	PowerShell	T-SQL
Create new tables with encrypted columns	Yes	Yes	Yes
Encrypt existing data in selected database columns	Yes	Yes	No

For details on configuring Always Encrypted, see:

- **Configure Always Encrypted using SQL Server Management Studio**[59]
- **Configure Always Encrypted using PowerShell**[60]

Exercise

In this exercise, you will use the Always Encrypted Wizard to configure Always Encrypted. The wizard will provision the required
keys and configure encryption for selected columns. If the columns you are setting encryption for already contain data, the wizard will encrypt the data.

Follow the steps that demonstrates the process for encrypting a column, at **Getting Started with Always Encrypted**[61].

Implement SSE

Azure SSE for data at rest helps you protect your data to meet your organizational security and compliance commitments. With this feature, the Azure storage platform automatically encrypts your data before persisting it to Azure Managed Disks, Azure Blob, Queue, or Table storage, or Azure Files, and decrypts the data before retrieval. Managing encryption, encryption at rest, decryption, and key management in SSE is transparent to users. All data written to the Azure storage platform is encrypted through 256-bit Advanced Encryption Standard (AES) encryption, one of the strongest block ciphers available.

SSE is enabled for all new and existing storage accounts and cannot be disabled. Because your data is secured by default, you don't need to modify your code or applications to utilize SSE.

SSE automatically encrypts data in:

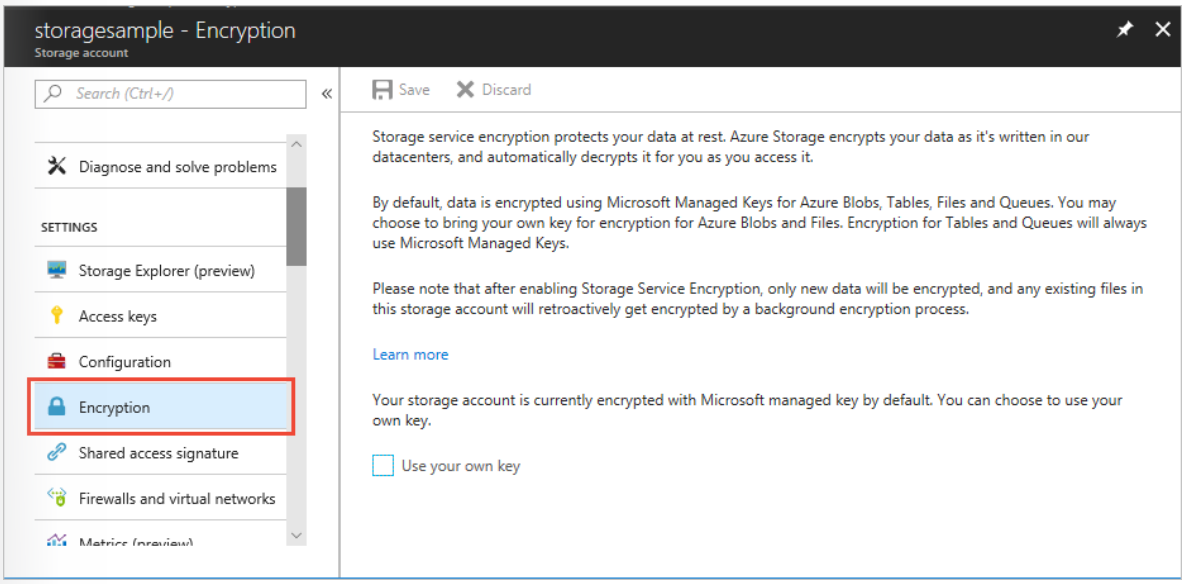
- Azure storage services:
  - Azure Managed Disks
  - Azure Blob storage
  - Azure Files
  - Azure Queue storage
  - Azure Table storage.
- Both performance tiers (Standard and Premium).
- Both deployment models (Resource Manager and classic).

SSE does not affect the performance of Azure storage services.

59 https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/configure-always-encrypted-using-sql-server-management-studio?view=sql-server-2017
60 https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/configure-always-encrypted-using-powershell?view=sql-server-2017
61 https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine?view=sql-server-2017

You can use Microsoft-managed encryption keys with SSE, or you can use your own encryption keys. For more information about using your own keys,
see **Storage Service Encryption using customer-managed keys in Azure Key Vault**[62].

To view settings for SSE, sign in to the **Azure portal**, and select a storage account. Under **SETTINGS**, select **Encryption**.



Azure Storage provides a comprehensive set of security capabilities that together help developers build secure applications. For more information, see **Azure Storage security guide**[63].

Implement disk encryption

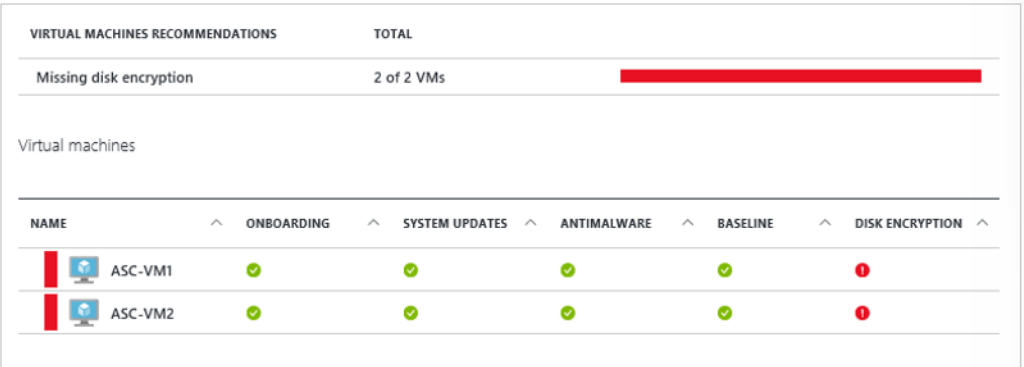
Azure Disk Encryption helps you encrypt your Windows IaaS and Linux IaaS VM disks. Disk Encryption leverages the industry standard **BitLocker**[64] feature of Windows, and the **DM-Crypt**[65] feature of Linux to provide volume encryption for the OS and data disks. The solution is integrated with Azure Key Vault to help you control and manage the disk-encryption keys and secrets. The solution also ensures that all data on the VM disks are encrypted at rest in your Azure storage.

Disk Encryption for Windows IaaS and Linux VMs is in General Availability in all Azure public regions and Azure Government regions for Standard VMs and VMs with Azure Premium Storage. When you apply the Disk Encryption management solution, you can satisfy the following business needs:

- IaaS VMs are secured at rest by using industry-standard encryption technology to address organizational security and compliance requirements.
- IaaS VMs boot under customer-controlled keys and policies. You can audit their usage in your key vault.

If you use Azure Security Center, you're alerted if you have VMs that aren't encrypted. The alerts display as High Severity, and the recommendation is to encrypt these VMs.

62 https://docs.microsoft.com/en-us/azure/storage/common/storage-service-encryption-customer-managed-keys
63 https://docs.microsoft.com/en-us/azure/storage/storage-security-guide
64 https://docs.microsoft.com/windows/security/information-protection/bitlocker/bitlocker-overview
65 https://en.wikipedia.org/wiki/Dm-crypt



Encryption features

When you enable and deploy Disk Encryption for Azure IaaS VMs, depending on the provided configuration the following capabilities are enabled:

- Encryption of the OS volume to protect the boot volume at rest in your storage.
- Encryption of data volumes to protect the data volumes at rest in your storage.
- Disable encryption on the OS and data drives for Windows IaaS VMs.
- Disable encryption on the data drives for Linux IaaS VMs (only when the OS drive isn't already encrypted).
- Safeguard the encryption keys and secrets in your Azure Key Vault subscription.
- Report the encryption status of the encrypted IaaS VM.
- Remove the disk encryption configuration settings from the IaaS VM.
- Back up and restore the encrypted VMs by using Azure Backup.

Azure Disk Encryption for IaaS VMS for Windows and Linux solution includes:

- Disk encryption extension for Windows
- Disk encryption extension for Linux
- PowerShell disk encryption cmdlets
- Azure CLI disk encryption cmdlets
- Resource Manager disk encryption templates

The Azure Disk Encryption solution is supported on IaaS VMs that run the Windows or Linux OS. Further information about encryption scenarios is available at **Encryption scenarios**[66].

Encryption workflow

To enable disk encryption for Windows and Linux VMs, complete the following high-level steps:

1.Choose an encryption scenario from the scenarios listed in the Encryption scenarios section.

2.Enable disk encryption and specify the encryption configuration via one of the following:

- Azure Disk Encryption Resource Manager template
- PowerShell cmdlets

66 https://docs.microsoft.com/en-us/azure/security/azure-security-disk-encryption-overview

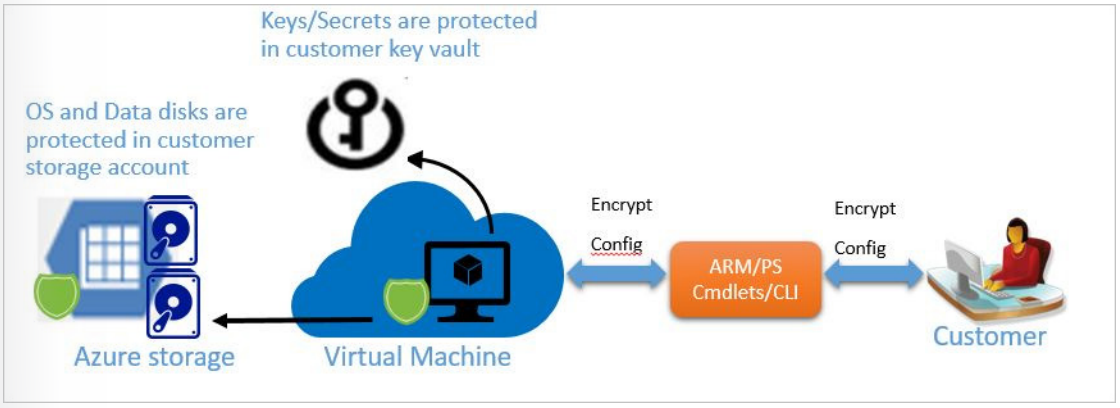
- Azure CLI

3.For the customer-encrypted VHD scenario, upload the encrypted VHD to your storage account and the encryption key material to your key vault. Then, provide the encryption configuration to enable encryption on a new IaaS VM.

4.For new VMs that you get from the Marketplace, and for existing VMs that already run in Azure, provide the encryption configuration to enable encryption on the IaaS VM.

5.Grant access to the Azure platform to read the encryption key material (BitLocker encryption keys for Windows systems and Passphrase for Linux) from your key vault to enable encryption on the IaaS VM.

Azure updates the VM service model with encryption and the key vault configuration, and sets up your encrypted VM.



Decryption workflow

To disable disk encryption for IaaS VMs:

- Choose to disable encryption (decryption) on a running IaaS VM in Azure, and specify the decryption configuration. You can disable via the Azure Disk Encryption Resource Manager template, PowerShell cmdlets, or the Azure CLI.

This step disables encryption of the OS, the data volume, or both on the running Windows IaaS VM. As mentioned in the previous section, disabling OS disk encryption for Linux isn't supported. The decryption step is allowed only for data drives on Linux VMs provided the OS disk isn't encrypted.

Azure updates the VM service model and the IaaS VM is marked as decrypted. The contents of the VM are no longer encrypted at rest.

Implement backup encryption

Azure Backup backs up data from on-premises machines and Azure VMs. You can back up and recover data at a granular level, including backups of files,
folders, machine system state, and app-aware data. Azure Backup manages the data at a granular level.

Azure Backup already provides encryption at rest by using the passphrase you provide while backing up data from on-premises to Azure. For virtual machines
in Azure, all new VMs that will be backed up and all new backup data from already backed-up VMs will be encrypted at rest through SSE. Existing data from your VM backups will be encrypted through SSE in a background activity.

Understand Application Security

Understanding Azure application endpoints

Enterprise developers and software as a service (SaaS) providers can develop commercial cloud services or line-of-business applications that can be integrated with Azure Active Directory (Azure AD) to provide secure sign-in and authorization for their services. To integrate an application or service with Azure AD, a developer must first register the application with Azure AD.

Any application that wants to use Azure AD capabilities must first be registered in an Azure AD tenant. This process involves giving Azure AD details about the application such as the URL where it's located, the URL to send replies after a user is authenticated, and the URI that identifies the app.

Exercise

In this exercise, you will add and register an application in Azure AD using the existing App registrations experience in the Azure portal.

Prerequisites

To get started, make sure have an Azure AD tenant that you can use to register your apps to. If you don't already have a tenant, see **Quickstart: Set up a dev environment**[67].

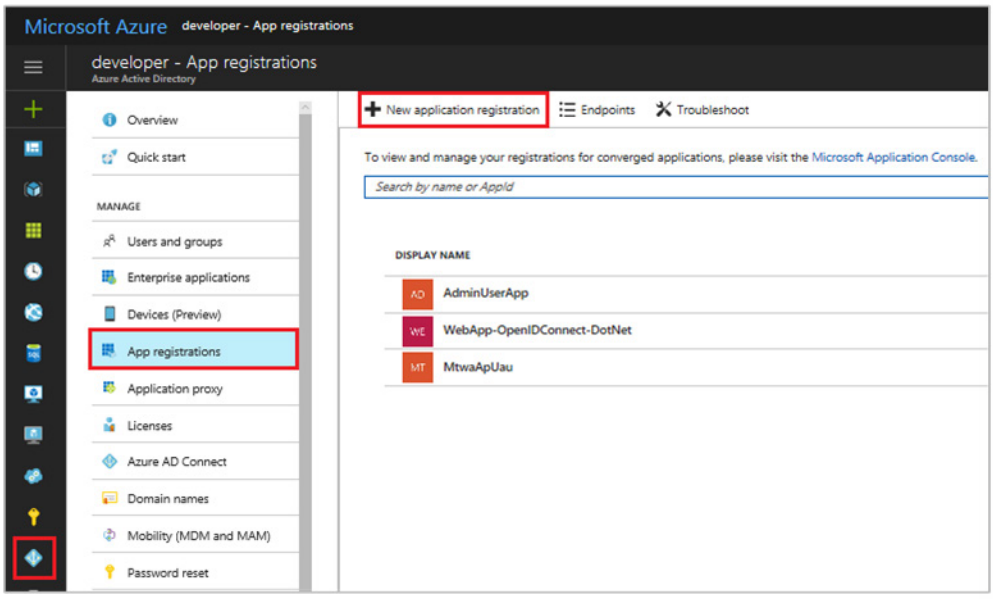
Register a new application using the Azure portal

1.Sign in to the Azure portal.

2.If your account gives you access to more than one portal, select your account in the top right corner, and set your portal session to the desired Azure AD tenant.

3.In the left navigation pane, select the **Azure Active Directory** service.

4.Select **App registrations**, and then select **New application registration**.



5.When the **Create** page appears, enter your application's registration information:

1.**Name**. Enter a meaningful application name.

67 https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-create-new-tenant

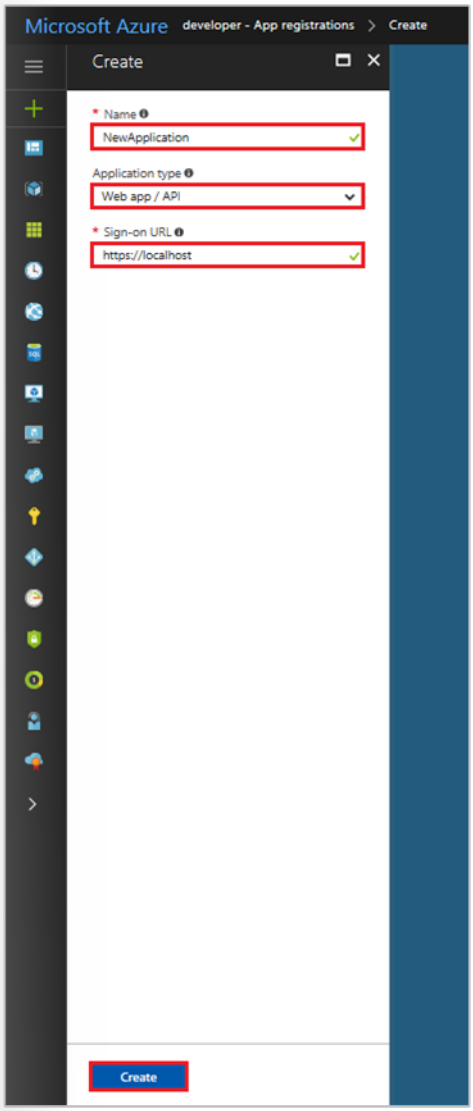
2.**Application type**. Select an application type from the drop-down:
for client applications that are installed locally on a device. This setting is used for OAuth public native clients.

*Select **Native** for client applications that are installed locally on a device. This setting is used for OAuth public native clients

*Select **Web app/API** for client applications and resource/API applications that are installed on a secure server. This setting is
used for OAuth confidential web clients and public user-agent-based clients. The same application can also expose both a client and resource/API.

3.**Sign-On URL**. For Web app/API applications, provide the base URL of your app. For example, http://localhost:31544 might be the URL for a web app running on your local machine. Users will use this URL to sign in to a web client application.

4.**Redirect URI**. For Native applications, provide the URI used by Azure AD to return token responses. Enter a value specific to your application, for example, **http://MyFirstAADApp**[68].



68 http://myfirstaadapp/

6.When finished, select **Create**.

Azure AD now assigns a unique Application ID to your application, and you're taken to your application's main registration page. Depending on whether your application is a web or native application, different options are provided to add additional capabilities to your application. By default, a newly registered web application is configured to allow only users from the same tenant to sign in to your application.

Summary

Any application that outsources authentication to Azure AD must be registered in a directory. This step involves telling Azure AD about your application, including the URL where it's located, the URL to send replies after authentication, and the URI to identify your application.

Understanding Azure Web App for Containers

Azure Container Service allows you to quickly deploy a production-ready Kubernetes, DC/OS, or Docker Swarm cluster. Web App for Containers provides a flexible way to use Docker images.

Azure Container Instances is a serverless way to run both Linux and Windows containers. It offers you an on-demand compute service delivering rapid deployment of containers with no VM management, and automatic, elastic scale.

Azure Container Instances is a deep security model, protecting each individual container at a hyper-visor level, which provides a stronger security boundary for multiple tenant scenarios. It can sometimes be a challenge to secure multiple tenant workloads running inside containers on the same virtual machine. Enabling this isolation without requiring you to create a hosting cluster is a true cloud native model.

Exercise

In this exercise you will use the Azure portal to create a Windows container in Azure and make its application available with a fully qualified domain name (FQDN). You will also use Azure Cloud Shell, and the new Azure PowerShell Az module.

Follow the steps that demonstrates the process for encrypting a column, at **Launch Azure Cloud Shell**[69]. This exercise is about 30 minutes.

For installation instructions, see **Install the Azure PowerShell module**[70].

To learn more about the new Az module and Azure Resource Manager compatibility, see **Introducing the new Azure PowerShell Az module.**[71]

Understanding of Application Insights

Application Insights is an extensible application performance monitoring (APM) service for web developers on multiple platforms. You use it to monitor your live web application, and it will automatically detect performance anomalies.

Application Insights includes powerful analytics tools to help you diagnose issues and understand what users actually do with your app. It's designed to help you continuously improve performance and usability. Application Insights works for apps on a wide variety of platforms including:

- Microsoft .NET
- Node.js
- Java Platform, Enterprise Edition (Java EE)

69 https://docs.microsoft.com/en-us/azure/container-instances/container-instances-quickstart-powershell
70 https://docs.microsoft.com/en-us/powershell/azure/install-az-ps
71 https://docs.microsoft.com/en-us/powershell/azure/new-azureps-module-az

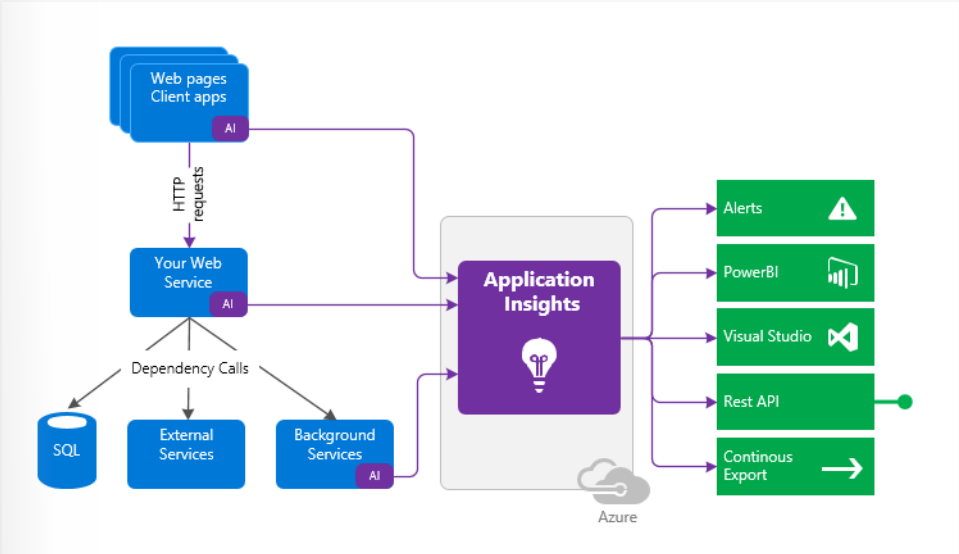
- Hosted on-premises
- Hybrid
- Any public cloud.

Application Insights integrates with your DevOps process, and has connection points to a variety of development tools. It can monitor and analyze telemetry from mobile apps by integrating with Microsoft Visual Studio App Center.

How APM works

After you install a small instrumentation package in your application and set up an Application Insights resource in the Azure portal, the instrumentation monitors your app and sends telemetry data to the portal. (The application can run anywhere—it doesn't have to be hosted in Azure.)

You can instrument the web service application, any background components, and the JavaScript in the web pages themselves.



In addition, you can pull in telemetry from the host environments such as performance counters, Azure diagnostics, or Docker logs. You can also set up web tests that periodically send synthetic requests to your web service. All these telemetry streams are integrated in the Azure portal, where you can apply powerful analytic and search tools to the raw data. The impact on your app's performance is very small. Tracking calls are non-blocking, and are batched and sent in a separate thread.

For insight on how users have used the telemetry for Application Insight, see the list of articles at **Where do I see my telemetry?**[72]

Understanding API management

API Management (APIM) helps organizations unlock the potential of their data and services by publishing APIs to external, partner, and internal developers. Businesses everywhere are looking to extend their operations as a digital platform by creating new channels, finding new customers, and driving deeper engagement with existing ones. APIM provides the core competencies to ensure a successful API program through developer engagement, business insights, analytics, security, and protection. You can use APIM to take any backend and launch a full-fledged API program based on it.

72 https://docs.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview

To use APIM, administrators create APIs. Each API consists of one or more operations, and can be added to one or more products. To use an API, developers subscribe to a product that contains that API, and then call the API's operation, subject to any usage policies that might be in effect. Common scenarios include:

- Securing mobile infrastructure by gating access with API keys, preventing denial of service attacks (DoS) by using throttling, or using advanced security policies like JSON Web Token (JWT) validation.
- Enabling independent software vendor (ISV) partner ecosystems by offering fast partner onboarding through the developer portal, and building an API facade to decouple from internal implementations that are not ready for partner consumption.
- Running an internal API program by offering a centralized location for the organization to communicate on a secured channel between the API gateway and the backend about the availability and latest changes to APIs, and gating access based on organizational accounts.

APIM is made up of the following components:

- API gateway. The API gateway is the endpoint that:
  - Accepts API calls and routes them to the backends.
  - Verifies API keys, JWT tokens, certificates, and other credentials.
  - Enforces usage quotas and rate limits.
  - Transforms your API on the fly without code modifications.
  - Caches backend responses where set up.
  - Logs call metadata for analytics purposes.
- Azure portal. The Azure portal is the administrative interface where you set up your API program. You can also use it to:
  - Define or import API schema.
  - Package APIs into products.
  - Set up policies such as quotas or transformations on the APIs.
  - Get insights from analytics.
  - Manage users.
- Developer portal. The Developer portal serves as the main web presence for developers. From here they can:
  - Read API documentation.
  - Try out an API via the interactive console.
  - Create an account and subscribe to get API keys.
  - Access analytics on their own usage.

For more information, see the **Cloud-based API Management: Harnessing the Power of APIs**[73] whitepaper.

The following video from Microsoft Ignite has additional details: **Azure API Management: Why, what, how, and what's next - BRK2186**[74]

73 https://j.mp/ms-apim-whitepaper
74 https://www.youtube.com/watch?v=HxhoAkhor-w

Understanding of Certificates

Transport Layer Security (TLS) is the basis for encryption of website data in transit. TLS uses certificates to encrypt and decrypt data. However, these certificates have a lifecycle that requires administrator management. Azure Key Vault is a tool that can assist in certificate management.

Certificates are used in Azure for cloud services (service certificates), and for authenticating with the management API (management certificates).

Certificates used in Azure are x.509 v3 can be signed by another trusted certificate, or they can be self-signed. A self-signed certificate is signed by its own creator; therefore, it is not trusted by default. Most browsers can ignore this problem. However, you should only use self-signed certificates when developing and testing your cloud services.

Certificates used by Azure can contain a private or a public key. Certificates have a thumbprint that provides a means to identify them in an unambiguous way. This thumbprint is used in the Azure configuration file to identify which certificate a cloud service should use.

For more information on Azure configuration files, see **Configuring SSL for an application in Azure**[75].

Service certificates

Service certificates are attached to cloud services and enable secure communication to and from the service. For example, if you deploy a web role, you would want to supply a certificate that can authenticate an exposed HTTPS endpoint. Service certificates, which are defined in your service definition, are automatically deployed to the VM that is running an instance of your role.

You can upload service certificates to Azure either using the Azure portal or by using the classic deployment model. Service certificates are associated with a specific cloud service. They are assigned to a deployment in the service definition file.

You can manage service certificates separately from your services, and you can have different people managing them. For example, a developer could upload a service package that refers to a certificate that an IT manager has previously uploaded to Azure. An IT manager can manage and renew that certificate (changing the configuration of the service) without needing to upload a new service package. Updating without a new service package is possible because the logical name, store name, and location of the certificate is in the service definition file, while the certificate thumbprint is specified in the service configuration file. To update the certificate, it's only necessary to upload a new certificate and change the thumbprint value in the service configuration file.

For more information on service certificates, see **What are service certificates?**[76]

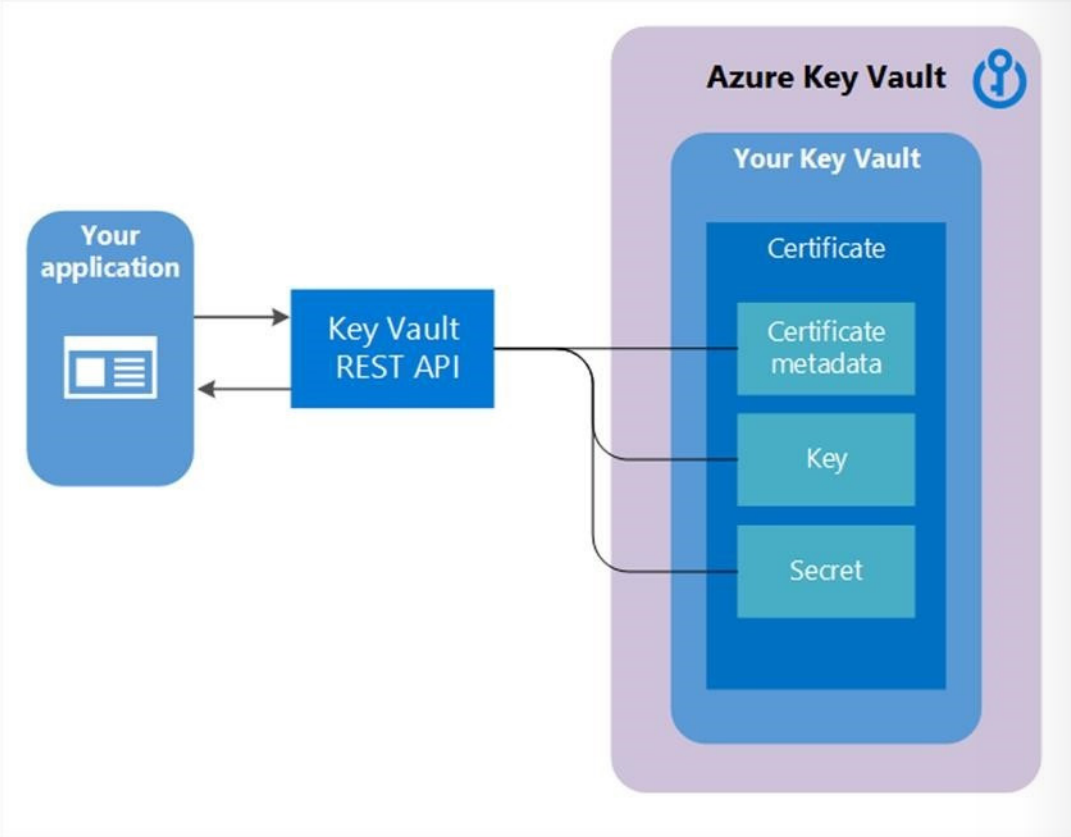
Management certificates

Management certificates allow you to authenticate with the classic deployment model. Many programs and tools (such as Visual Studio or the Azure SDK) use these certificates to automate configuration and deployment of various Azure services. However, these are not really related to cloud services.

For more information on management certificates, see **What are management certificates?**[77]

Certificates are composed of three interrelated resources linked together as a Key Vault certificate: certificate metadata, a key, and a secret.

75 https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-configure-ssl-certificate-portal
76 https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-certs-create
77 https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-certs-create



Before you can create a certificate in a Key Vault, prerequisite you must successfully complete steps 1 and 2, and a key vault must exist for this user or organization.

1.On-boarding as the IT Admin, public key infrastructure (PKI) Admin, or anyone managing accounts with certificate authority (CA) providers, for a given company is a prerequisite to using Key Vault certificates.

2.An account admin for a CA provider creates credentials for use by Key Vault to enroll, renew, and use SSL certificates via Key Vault.

3.Depending on the CA, an admin, along with an employee (Key Vault user) who owns certificates can get a certificate from the admin or directly from the account with the CA.

For more information on creating accounts with CA Providers, see **Manage certificates via Azure Key Vault**[78].

4.Set up certificate contacts for notifications. These are the contacts for the Key Vault user. While Key Vault does not enforce this step, this is a one-time operation.

Create a certificate with a CA-partnered Key Vault

The following steps correspond to the green numbered steps in the following diagram:

1.Your application is creating a certificate which internally begins by creating a key in your key vault.

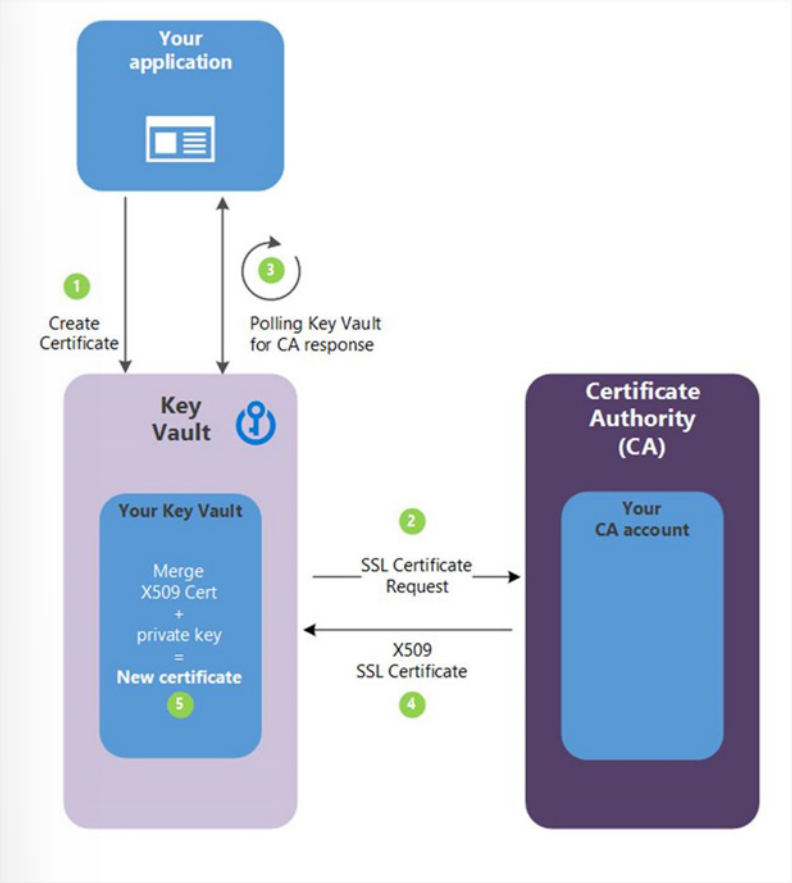
2.Key Vault sends an SSL Certificate Request to the CA.

3.Your application polls—in a loop and wait process—for your Key Vault for certificate completion. The certificate creation is complete when Key Vault receives the CA's response with x509 certificate.

78 https://aka.ms/kvcertsblog

4.The CA responds to Key Vault's SSL Certificate Request with an X509 SSL Certificate.

5.Your new certificate creation completes with the merger of the X509 Certificate for the CA.



Automating certificate management helps to reduce or eliminate the error prone task of manual certificate management.

Understand security considerations for application lifecycle management solutions

The Microsoft Security Development Lifecycle (SDL) introduces security and privacy considerations throughout all phases of the development process. It helps developers build highly secure software, address security compliance requirements, and reduce development costs. The guidance, best practices, tools, and processes in the SDL are practices used internally at Microsoft to build more secure products and services.

Since first sharing SDL in 2008, we've updated the practices as a result of our growing experience with new scenarios such as cloud services, IoT, and AI.

Provide training

Security is everyone's job. Developers, service engineers, and program and project managers must understand security basics. They all must know how to build security into software and services to make products more secure, while still addressing business needs and delivering user value. Effective training will complement and re-enforce security policies, SDL practices, standards, and requirements of software security, and be guided by insights derived through data or newly available technical capabilities.

Although security is everyone's job, it's important to remember that not everyone needs to be a security expert nor strive to become a proficient penetration tester. However, ensuring everyone understands the attacker's perspective, their goals, and the art of the possible will help capture the attention of everyone and raise the collective knowledge bar.

Define security requirements

Considering security and privacy is a fundamental aspect of developing highly secure applications and systems. Regardless of development methodology being used, security requirements must be continually updated to reflect changes in required functionality and changes to the threat landscape. Obviously, the optimal time to define the security requirements is during the initial design and planning stages as this allows development teams to integrate security in ways that minimize disruption.

Factors that influence security requirements include (but are not limited to) the legal and industry requirements, internal standards and coding practices, review of previous incidents, and known threats. These requirements should be tracked through either a work-tracking system or through telemetry derived from the engineering pipeline.

Define metrics and compliance reporting

It's essential for an organization to define the minimum acceptable levels of security quality, and to hold engineering teams accountable to meeting that criteria. Defining these expectations early helps a team understand risks associated with security issues, identify and fix security defects during development, and apply the standards throughout the entire project. Setting a meaningful security bar involves clearly defining the severity thresholds of security vulnerabilities (for example, all known vulnerabilities discovered with a "critical" or "important" severity rating must be fixed with a specified time frame), and never relaxing it after it's been set.

To track key performance indicators (KPIs) and ensure security tasks are completed, bug tracking and/or work tracking mechanisms used by an organization (such as Azure DevOps) should allow for security defects and security work items to be clearly labeled as security, and marked with their appropriate security severity. This allows for accurate tracking and reporting of security work.

You can read more about defining metrics and compliance reporting at:

- **SDL Privacy Bug Bar Sample**[79]
- **Add or modify a field to track work**[80]
- **SDL Security Bug Bar Sample**[81]

Perform threat modeling

Threat modeling should be used in environments where there is a meaningful security risk. As a practice, it allows development teams to consider, document, and discuss the security implications of designs in the context of their planned operational environment, and in a structured fashion. Applying a structured approach to threat scenarios helps a team more effectively and less expensively identify security vulnerabilities, determine risks from those threats, and then make security feature selections and establish appropriate mitigations. You can apply threat modeling at the component, application, or system level.

More information is available at **Threat Modeling**[82].

Establish design requirements

79 https://msdn.microsoft.com/en-us/library/cc307403.aspx
80 https://docs.microsoft.com/en-us/azure/devops/reference/add-modify-field?view=tfs-2018&viewFallbackFrom=vsts
81 https://msdn.microsoft.com/library/cc307404.aspx
82 https://www.microsoft.com/en-us/securityengineering/sdl/threatmodeling

The SDL is typically thought of as assurance activities that help engineers implement more secure features, meaning the features are well engineered with respect to security. To achieve this, engineers typically rely on security features such as cryptography, authentication, and logging. In many cases, selecting or implementing security features has proven to be so complicated that design or implementation choices are likely to result in vulnerabilities. Therefore, it's crucially that they are applied consistently and with a consistent understanding of the protection they provide.

Define and use cryptography standards

With the rise of mobile and cloud computing, it's important to ensure all data—including security-sensitive information and management and control data—is protected from unintended disclosure or alteration when it's being transmitted or stored. Encryption is typically used to achieve this. However, making an incorrect choice when using any aspect of cryptography can be catastrophic. Therefore, it's best to develop clear encryption standards that provide specifics on every element of the encryption implementation.

Encryption should be left to experts. A good general rule is to only use industry-vetted encryption libraries and ensure they're implemented in a way that allows them to be easily replaced if needed.

For more information on encryption, see the **Microsoft SDL Cryptographic Recommendations**[83] whitepaper.

Manage security risks from using third-party components

The vast majority of software projects today are built using third-party components (both commercial and open source). When selecting which third-party components to use, it's important to understand the impact that a security vulnerability in them could have to the security of the larger system into which they are integrated. Having an accurate inventory of these components and a plan to respond when new vulnerabilities are discovered, will go a long way toward mitigating this risk. However, you should also consider additional validation, depending on your organization's risk appetite, the type of component being used, and potential impact of a security vulnerability.

Learn more about managing the security risks of using third-party components at:

- **Open source**[84]
- **Managing Security Risks Inherent in the Use of Third-Party Components**[85]
- **Managing Security Risks Inherent in the Use of Open-Source Software**[86]

Use approved tools

Define and publish a list of approved tools and their associated security checks, such as compiler/linker options and warnings. Engineers should strive to use the latest version of approved tools (such as compiler versions), and to utilize new security analysis functionality and protections.

For more information, see:

- **Recommended Tools, Compilers and Options for x86, x64 and ARM**[87] (whitepaper)
- **SDL Resources**[88]

Perform Static Analysis Security Testing

83 http://download.microsoft.com/download/6/3/A/63AFA3DF-BB84-4B38-8704-B27605B99DA7/Microsoft SDL Cryptographic Recommendations.pdf
84 https://www.microsoft.com/en-us/securityengineering/opensource/
85 https://safecode.org/wp-content/uploads/2017/05/SAFECode_TPC_Whitepaper.pdf
86 https://www.microsoft.com/en-us/securityengineering/opensource/
87 http://download.microsoft.com/download/6/3/A/63AFA3DF-BB84-4B38-8704-B27605B99DA7/Recommended Tools, Compilers and Options for x86, x64 and ARM.pdf
88 https://www.microsoft.com/en-us/securityengineering/sdl/resources

Analyzing source code prior to compilation provides a highly scalable method of security code review, and helps ensure that secure coding policies are being followed. Static Analysis Security Testing (SAST) is typically integrated into the commit pipeline to identify vulnerabilities each time the software is built or packaged. However, some offerings integrate into the developer environment to spot certain flaws such as the existence of unsafe or other banned functions, and then replace those with safer alternatives while the developer is actively coding. There is no one size fits all solution; development teams should decide the optimal frequency for performing SAST, and consider deploying multiple tactics to balance productivity with adequate security coverage.

More information is available at:

- **Microsoft DevSkim on GitHub** [89]
- **Roslyn Security Guard Rules**[90]
- **Visual Studio Marketplace**[91]
- **Analyzing C/C++ Code Quality by Using Code Analysis**[92]
- **Microsoft binskim on GitHub**[93]

Perform Dynamic Analysis Security Testing

Performing run-time verification of your fully compiled or packaged software checks functionality that is only apparent when all components are integrated and running. This is typically achieved using a tool, a suite of prebuilt attacks, or tools that specifically monitor application behavior for memory corruption, user privilege issues, and other critical security problems. Similar to SAST, there is no one-size-fits-all solution and while some tools (such as web app scanning tools) can be more readily integrated into the CI/CD pipeline, other Dynamic Application Security Testing (DAST) such as fuzzing requires a different approach.

More information is available at:

- **Visual Studio Marketplace**[94]
- **Automated Penetration Testing with White-Box Fuzzing**[95]

Perform penetration testing

Penetration testing is a security analysis of a software system performed by skilled security professionals simulating the actions of a hacker. The objective of a penetration test is to uncover potential vulnerabilities resulting from coding errors, system configuration faults, or other operational deployment weaknesses.. By doing this, the test typically finds the broadest variety of vulnerabilities. Penetration tests are often performed in conjunction with automated and manual code reviews to provide a greater level of analysis than would ordinarily be possible.

More information is available at:

- **Attack Surface Analyzer**[96]
- **SDL Security Bug Bar Sample**[97]

Establish a standard incident response process

89 https://github.com/Microsoft/DevSkim
90 https://dotnet-security-guard.github.io/rules.htm
91 https://marketplace.visualstudio.com/search?term=security&target=AzureDevOps&category=All categories&sortBy=Relevance
92 https://msdn.microsoft.com/en-us/library/ms182025.aspx
93 https://github.com/Microsoft/binskim
94 https://marketplace.visualstudio.com/search?term=security&target=AzureDevOps&category=All categories&sortBy=Relevance
95 https://msdn.microsoft.com/library/cc162782.aspx
96 https://go.microsoft.com/?linkid=9758398
97 https://msdn.microsoft.com/library/cc307404.aspx

Preparing an incident response plan is crucial for helping to address new threats that can emerge over time. It should be created in coordination with your organization's dedicated Product Security Incident Response Team (PSIRT). The plan should include who to contact in case of a security emergency, and establish the protocol for security servicing, including plans for code inherited from other groups within the organization and for third-party code. The incident response plan should be tested before it is needed! (whitepaper)

For more information about incident responses, see:

- **Microsoft Incident Response Reference Guide**[98]
- **Using Azure Security Center for an incident response**[99]
- **Security Incident Management in Microsoft Office 365**[100] (whitepaper)
- **Microsoft Incident Response and shared responsibility for cloud computing**[101]
- **Microsoft Security Response Center**[102]

Summary

By introducing standardized security and compliance considerations throughout all phases of the development process, developers can help reduce the likelihood of vulnerabilities in products and services and avoid repeating the same security mistakes. Similarly, security integration throughout the operations lifecycle will assist in maintaining the integrity of those products and services. These Operational Security Assurance practices should align with the development processes. This will result in less time—and therefore cost—being spent on triage and response after the fact, and provide your customers with assurance that your products are highly secure.

98 https://info.microsoft.com/INCIDENT-RESPONSE-REFERENCE-GUIDE.html
99 https://docs.microsoft.com/en-us/azure/security-center/security-center-incident-response
100 http://download.microsoft.com/download/2/F/1/2F16A9CA-8D4F-4BB5-8F85-3A362131A95B/Office 365 Security Incident Management.pdf
101 https://azure.microsoft.com/en-us/blog/microsoft-incident-response-and-shared-responsibility-for-cloud-computing/
102 https://www.microsoft.com/en-us/msrc

Implement Security Validations for Application Development

Implement security validations for application development

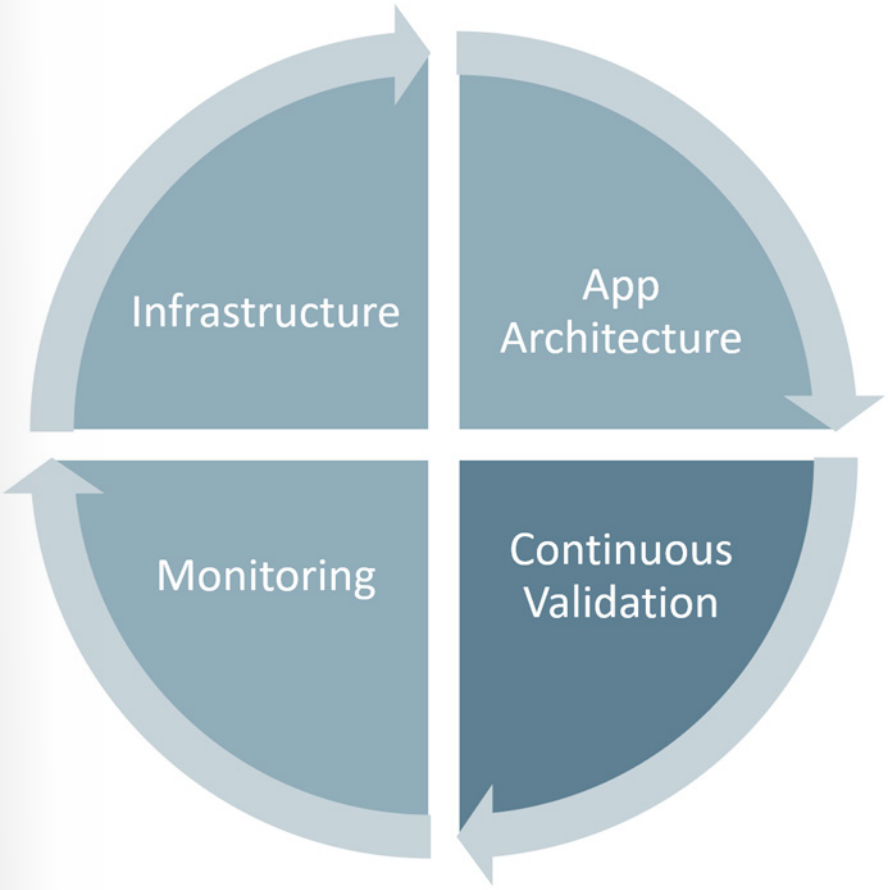
If you're planning Azure DevOps Continuous Integration/Continuous Deployment (CI/CD) pipelines, you probably have a few questions, such as:

- How do I ensure my application is safe?
- How do I add continuous security validation to my CI/CD pipeline?

DevOps practices are enabling businesses to stay ahead of the competition by delivering new features at a faster pace. Even though you might have an increase in production deployments, you must ensure that business agility doesn't come at the expense of security.

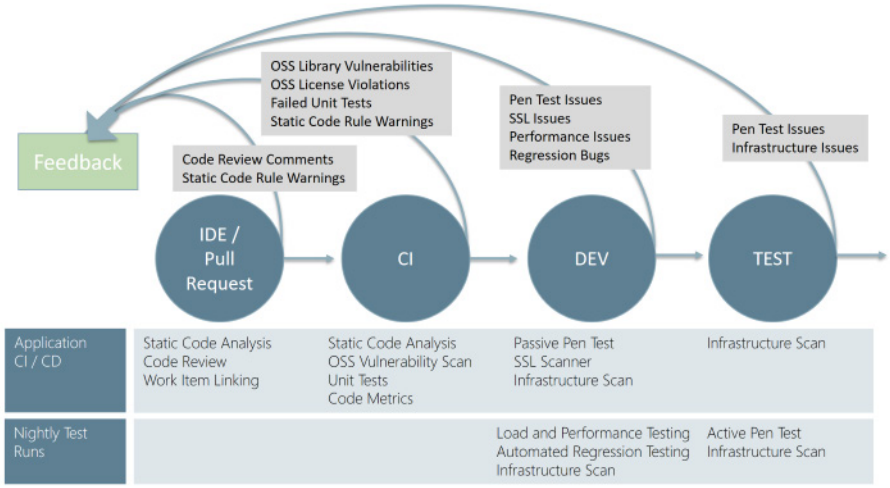
With continuous delivery, how do you ensure your applications remain secure? How can you find and fix security issues early
in the process? **DevSecOps** incorporates the security team and their capabilities into your DevOps practices, making security
a responsibility of everyone on the team. One key change is to add continuous security validation to your CI/CD pipeline. This section will walk through that process.

Security needs to shift from an afterthought to being evaluated at every step of the process. Securing applications is a continuous process that encompasses secure infrastructure, designing an architecture with layered security, continuous security validation, and monitoring for attacks.



Organizations should have continuous security validation at each step (from development through production) to help ensure the application is always secure. This switches the conversation with the security team from approving each release to approving the CI/CD process and having the ability to monitor and audit the process at any time.

The following diagram highlights the key validation points in the CI/CD pipeline for a greenfield application. Depending on your platform and where your application is at in its lifecycle, you might need to consider implementing tools gradually. This is true as well for mature products that haven't previously had any security validation run against them.



IDE / pull request

Validation in the CI/CD begins before the developer commits his or her code. Static code analysis tools in the integrated development environment (IDE) provide the first line of defense to help ensure that security vulnerabilities are not introduced into the CI/CD process. The process for committing code into a central repository should have controls to help prevent security vulnerabilities from being introduced.

Using Git source control in Azure DevOps with branch policies provides a gated commit experience that can provide this validation. By enabling branch policies on the shared branch, a pull request is required to initiate the merge process and ensure that all defined controls are being executed. The pull request should require a code review, which is the one manual but important check for identifying new issues being introduced into the code. Along with this manual check, commits should be linked to work items for auditing why the code change was made and require a continuous integration (CI) build process to succeed before the push can be completed.

CI

After the merge completes, you execute the CI build as part of the pull request (PR) process. Typically, the primary difference between the two runs is that the PR/CI process doesn't need any of the packaging or staging that's done in the CI build. These CI builds should run static code analysis tests to ensure that the code is following all rules for both maintenance and security. You can use several tools for this, including:

- Visual Studio Code Analysis and the Roslyn Security Analyzers
- Checkmarx. A Static Application Security Testing (SAST) tool
- BinSkim. A binary static analysis tool that provides security and correctness results for Windows portable executables

More information about these tools is available at:

- **Use Roslyn analyzers**[103]
- **Checkmarx**[104]
- **BinSkim**[105]

103 https://docs.microsoft.com/en-us/visualstudio/code-quality/use-roslyn-analyzers?view=vs-2017
104 https://www.checkmarx.com/
105 https://github.com/Microsoft/binskim

The following blog post looks at three extensions that add support for OSS security and license validation, as well as code scanning,
to assist you in spending less time to build more secure software.
Team Services October Extensions Roundup – Rugged DevOps[106]

Many of the tools seamlessly integrate into the Azure pipelines build process. Visit the VSTS Marketplace for more information on the integration capabilities of these tools.

In addition to code quality being verified with the CI build, two other tedious and often ignored validations are scanning third-party packages for vulnerabilities and open source license usage. Often, the process for managing third-party packages vulnerabilities and open source licenses is manual and tedious.

Fortunately, WhiteSource Software has a tool that helps make this identification process almost instantaneous. WhiteSource Bolt runs through each build and reports all of the vulnerabilities and the licenses of the third-party packages, and a 6-month license is included with your Visual Studio subscription. WhiteSource Bolt provides a report of these items, but doesn't include the advanced management and alerting capabilities that the full product offers. With new vulnerabilities being regularly discovered, your build reports could change even though your code doesn't. Checkmarx includes a similar WhiteSource Bolt, integration so there could be some overlap between the two tools.

For more information about WhiteSource Bolt and the Azure Pipelines integration, see **Manage your open source usage and security as reported by your CI/CD pipeline**[107].

Application deployment to DEV and TEST

After you have verified your code quality and deployed the application to a lower environment (such as development or quality assurance),
the process should verify that there are no security vulnerabilities in the running application. To test for this, you can execute
automated penetration tests against the running application to scan it for vulnerabilities.

There are different levels of tests, which are categorized as passive tests and active tests:

- Passive tests scan the target site as is, but don't try to manipulate the requests to expose additional vulnerabilities. These types of tests run fast and are usually a good candidate for a CI process that you want to complete in a few minutes.
- Active tests simulate many techniques that hackers commonly use to attack websites. Often referred to as dynamic tests or fuzz tests, active tests try a large number of different combinations to see if the site reveals any information. These types of tests can run for much longer, and thus are better executed nightly as part of a separate Azure DevOps release.

One tool to consider for penetration testing is **Open Web Application Security Project (OWASP) Zed Attack Proxy (ZAP)**. OWASP is a
worldwide not-for-profit organization dedicated to helping improve the quality of software. ZAP is a free penetration testing tool
for beginners to professionals. ZAP includes an API and a weekly docker container image that can be integrated into your deployment
process. The detailed how-to steps are outside the scope of this article. Refer to the OWASP ZAP VSTS extension repo for details
on how to set up the integration.

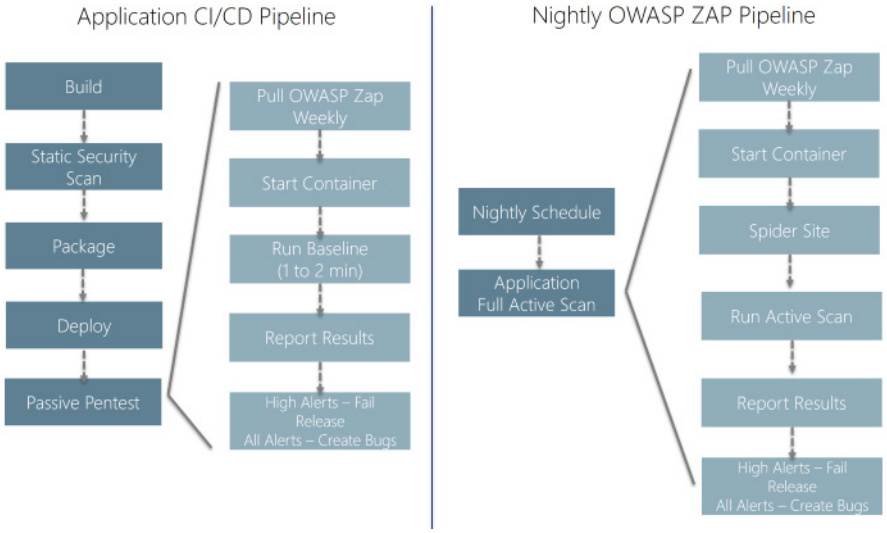
106 https://blogs.msdn.microsoft.com/devops/2016/10/11/team-services-october-extensions-roundup-rugged-devops/
107 https://blogs.msdn.microsoft.com/visualstudioalmrangers/2017/06/08/manage-your-open-source-usage-and-security-as-reported-by-your-cicd-pipeline/

More information about OWASP and ZAP is available at the following sites:

- **OWASP**[108]
- **Tools to run OWASP ZAP container in VSTS build and release**[109]

The application CI/CD pipeline should be able to complete within a few minutes, so you don't want to include any long-running processes.
The baseline scan is designed to identify vulnerabilities within a couple of minutes, making it a good option for the application
CI/CD pipeline. The nightly OWASP ZAP can search the website and run the full active scan to evaluate the most combinations of possible vulnerabilities.

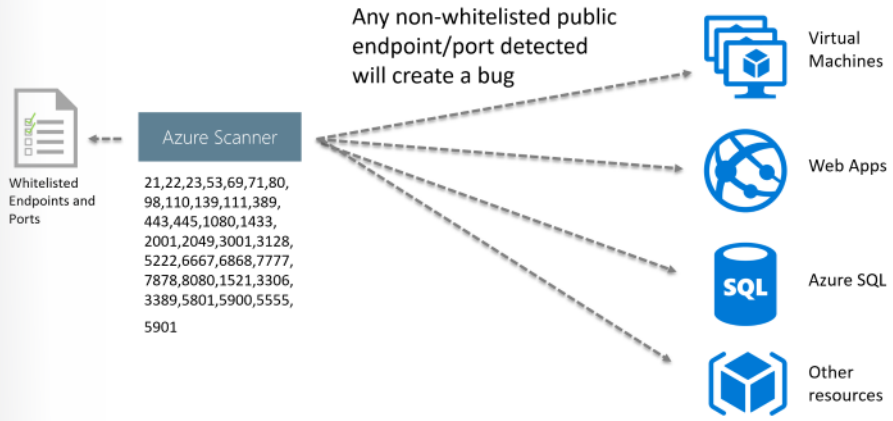
You can install OWASP ZAP on any machine in your network, but we recommend using the OWASP Zap/Weekly docker container within Azure
Container Service. This method allows for the latest updates to the image, and spinning up multiple instances of the image so several
applications within an enterprise can be scanned simultaneously. The following figure outlines the steps for both the Application CI/CD
pipeline and the longer running nightly OWASP ZAP pipeline.



In addition to validating the application, you should also validate the infrastructure to check for any vulnerabilities.

When using a public cloud such as Azure, deploying an application and shared infrastructure is easier. Therefore, it's important to validate that everything has been done securely. Azure includes many tools to help report and prevent these vulnerabilities, including Security Center and Azure Policy. In addition, Azure Information Protection Scanner (Azure Scanner) an ensure any public endpoints and ports have been whitelisted or else it will raise an infrastructure issue. This is run as part of the network pipeline to provide immediate verification, but it also needs to be executed each night to ensure that there are no publicly exposed resources that should not be.

108 https://www.owasp.org/
109 https://github.com/deliveron/owasp-zap-vsts-extension



After the scans complete, the Azure Pipeline's release is updated with a report that includes the results ,and bugs are created in the team's backlog. If the vulnerability has been fixed, resolved bugs will close. If the vulnerability still exists, the bugs will move back into in-progress.

The benefit of using this is that the vulnerabilities are created as bugs, which provide actionable work that can be tracked and measured. You can suppress false positives using the OWASP ZAP context file, so only true vulnerabilities are highlighted.

Even with continuous security validation running against every change, malicious hackers are continuously changing their approaches, and new vulnerabilities are being discovered. Good monitoring tools help you to detect, prevent, and remediate issues discovered while your application is running in production. Azure provides a number of tools that provide detection, prevention, and alerting using rules such as OWASP Top 10 / modSecurity. Now it's even using machine learning to detect anomalies and unusual behavior to help identify attackers.

Summary

Minimize your security vulnerabilities by taking a holistic and layered approach to security, including secure infrastructure, application architecture, continuous validation, and monitoring. DevSecOps practices enable your entire team to incorporate these security capabilities throughout the entire lifecycle of your application. Establishing continuous security validation into your CI/CD pipeline can allow your application to stay secure while you are improving the deployment frequency to meet needs of your business.

Configure synthetic security transactions

Distributed applications and services running in the cloud are by their nature complex pieces of software that comprise many moving parts. In a production environment, it's important to be able to track the way in which users utilize your system, trace resource utilization, and monitor the health and performance of your system. You can use this information as a diagnostic aid to detect and correct issues, and to help spot potential problems and prevent them from occurring.

Synthetic transactions are the capability to check the availability of an application across a network. These transactions are automated, self-contained, simulated user transactions. When run, they do not cause regression.

Health monitoring

A system is healthy if it's running and capable of processing requests. The purpose of health monitoring is to generate a snapshot of the system's current health so that you can verify that all of its components are functioning as expected.

The raw data that's required to support health monitoring can be generated as a result of:

- Tracing execution of user requests. This information can be used to determine which requests have succeeded, which have failed, and how long each request takes.
- Synthetic user monitoring. This process simulates the steps performed by a user and follows a predefined series of steps. The results of each step should be captured.
- Logging exceptions, faults, and warnings. This information can be captured as a result of trace statements embedded into the application code, in addition to retrieving information from the event logs of any services that the system references.
- Monitoring the health of any third-party services that the system uses. This might require retrieving and parsing health data that these services supply. This information can take a variety of formats.
- Endpoint monitoring. This mechanism is described in more detail in the "Configure Traffic Manager" in module 2.
- Collecting ambient performance information. This might include background CPU utilization or I/O (including network) activity.

Synthetic user monitoring

With synthetic user monitoring, you write your own test client that simulates a user and performs a configurable but typical series of operations. You can track the test client's performance to help determine the state of the system. You can also use multiple instances of the test client as part of a load-testing operation to establish how the system responds under stress, and what monitoring output is generated under these conditions. Each test case will be dependent on what type of service you are testing.

A more in-depth discussion of how to instrument an application is available at **Monitoring and diagnostics**[110].

110 https://docs.microsoft.com/en-us/azure/architecture/best-practices/monitoring

Secure Applications

Configure SSL TLS certificates

Configure SSL/TLS certificates

Azure uses certificates for cloud services (service certificates) and for authenticating with the management API (management certificates). Certificates used in Azure are x.509 v3 certificates, and can either be signed by another trusted certificate or be self-signed.

A self-signed certificate is signed by its own creator; therefore, it's not trusted by default. Most browsers can ignore this problem. However, you should only use self-signed certificates when developing and testing your cloud services.

Certificates used by Azure can contain a private or a public key. Certificates have a thumbprint that provides a means to identify them in an unambiguous way. This thumbprint is used in the Azure configuration file to identify which certificate a cloud service should use.

Service certificates and **Management certificates** are covered earlier in this module. It is important to understand the differences.

Create a new self-signed certificate

To create a self-signed certificate, you can use any tool that adheres to the following requirements:

- Have an X.509 certificate
- Contains a private key
- Created for key exchange (.pfx file)
- Subject name must match the domain used to access the cloud service:
  - You cannot acquire an SSL certificate for the cloudapp.net (or for any Azure-related) domain; the certificate's subject name must match the custom domain name used to access your application. For example, contoso.net is acceptable, while contoso.cloudapp.net is not.
- Minimum of 2048-bit encryption
- Service certificate only, client-side certificate must reside in the Personal certificate store

You can create a certificate on Windows Server with the **makecert.exe** tool, or **IIS**.

The following PowerShell script can create a self-signed certificate.

PowerShell

```
$cert = New-SelfSignedCertificate -DnsName yourdomain.cloudapp.net -CertStore-
Location "cert:\LocalMachine\My" -KeyLength 2048 -KeySpec "KeyExchange"

$password = ConvertTo-SecureString -String <your-password> -Force -AsPlain-
Text

Export-PfxCertificate -Cert $cert -FilePath ".\my-cert-file.pfx" -Password
$password
```

If you want to use the certificate with an IP address instead of a domain, use the IP address in the -DnsName parameter.

Once you have a management certificate created, (.cer file with only the public key) you can upload it into the portal.

When the certificate is available in the portal, anyone with a matching certificate (private key) can connect through
the Management API and access the resources for the associated subscription.

If you want to use this certificate with the management portal, export it to a .cer file:

```
Export-Certificate -Type CERT -Cert $cert -FilePath .\my-cert-file.cer
```

Instructions on how to upload the management certificate to can be found **here**[111].

Internet Information Services (IIS)

More information about how to create a certificate with IIS can be found at **How to create an IIS website that requires client certificate using self-signed certificates**[112].

Linux

You can read more about how to create certificates with SSH at **Quick steps: Create and use an SSH public-private key pair for Linux VMs in Azure**[113].

Next steps

For more information about uploading your service certificate to the Azure portal, see **Configuring SSL for an application in Azure**[114].

To learn more about uploading a management API certificate to the Azure portal, see **Upload an Azure Service Management Certificate**[115].

Configure managed service identity for app services

This lesson shows you how to create a managed identity for App Service and Azure Functions applications, and how to use that identity to access other resources. A managed identity from Azure Active Directory allows your app to easily access other Azure AD–protected resources such as Azure Key Vault. The identity is managed by the Azure platform and does not require you to provision or rotate any secrets.

Your application can be granted two types of identities:

- A *system-assigned identity* is tied to your application and is deleted if your app is deleted. An app can only have one system-assigned identity. System-assigned identity support is generally available for Windows apps.
- A *user-assigned identity* is a standalone Azure resource that can be assigned to your app. An app can have multiple user-assigned identities. User-assigned identity support is in preview for all app types.

Here is a quick review of managed identifies for Azure resources.

A common challenge when building cloud applications is how to manage the credentials in your code for authenticating to cloud services. Keeping the credentials secure is an important task. Ideally, the credentials never appear on developer workstations and aren't checked into source control. Azure Key Vault provides a way to securely store credentials, secrets, and other keys, but your code has to authenticate to Key Vault to retrieve them.

111 https://docs.microsoft.com/en-us/azure/azure-api-management-certs
112 https://blogs.msdn.microsoft.com/asiatech/2016/08/22/how-to-create-an-iis-website-that-requires-client-certificate-using-self-signed-certificates/
113 https://docs.microsoft.com/en-us/azure/virtual-machines/linux/mac-create-ssh-keys?toc=%2fazure%2fvirtual-machines%2flinux%2ftoc.json
114 https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-configure-ssl-certificate-portal
115 https://docs.microsoft.com/en-us/azure/azure-api-management-certs

The managed identities for Azure resources feature in Azure AD solves this problem. This feature provides Azure services with an automatically managed identity in Azure AD. You can use the identity to authenticate to any service that supports Azure AD authentication, including Key Vault, without any credentials in your code.

The managed identities for Azure resources feature is free with Azure AD for Azure subscriptions. Managed identities for Azure resources is the new name for the service formerly known as *Managed Service Identity (MSI)*.

For an in-depth look at how managed identities for Azure resources work, see **How does the managed identities for Azure resources work?**[116]

Adding a system-assigned identity

This lesson shows you how to create a managed identity for App Service, an Azure Functions applications, and how to use it to access other resources. A managed identity from Azure Active Directory allows your app to easily access other AAD-protected resources such as Azure Key Vault. The identity is managed by the Azure platform and does not require you to provision or rotate any secrets.

Using the Azure portal

To set up a managed identity in the Azure portal, you will first create an application and then enable the feature. The following high-level steps describe this process:

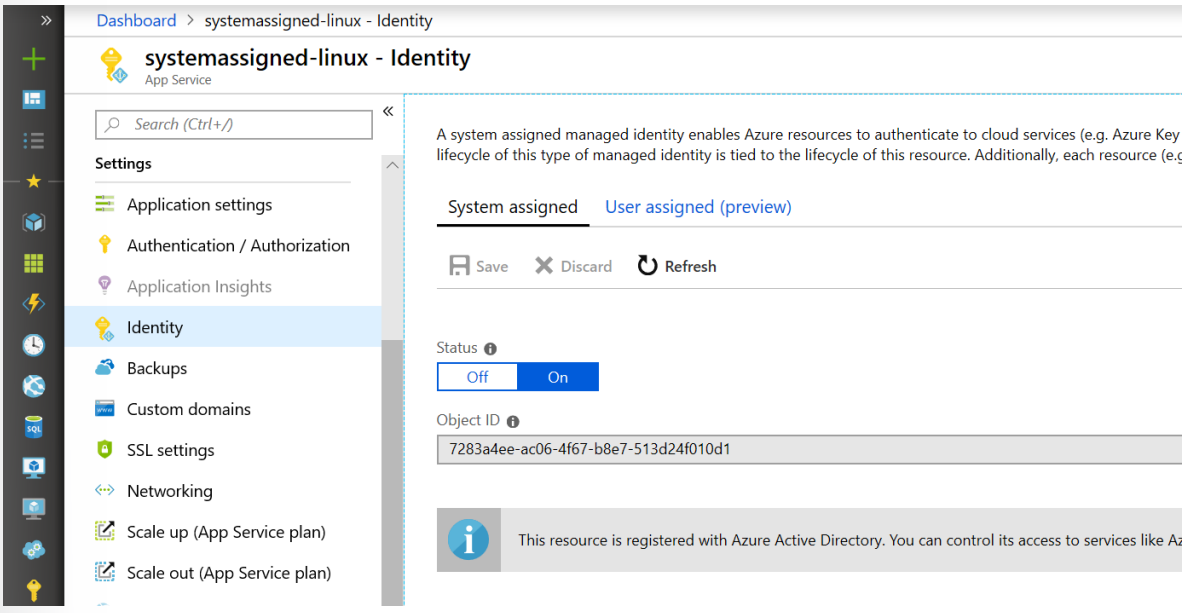
1.Create an app in the Azure portal.

2.Navigate to it in the portal.

3.If you are using a function app, navigate to **Platform features**. For other app types, scroll down to the **Settings** group in the left navigation.

4.Select **Managed identity**.

5.Within the **System assigned** tab, switch **Status** to **On**, and then select **Save**.



116 https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/overview

Other options include using Azure Command Line Interface (CLI), Azure PowerShell, or Azure Resource Manager template. You can read more about these options at:

- **Using the Azure CLI**[117]
- **Using Azure PowerShell**[118]
- **Using an Azure Resource Manager template**[119]

Adding a user-assigned identity (preview)

Creating an app with a user-assigned identity requires that you create the identity and then add its resource identifier to your app configuration.

Using the Azure portal

First, you'll need to create a user-assigned identity resource.

1.Create a user-assigned managed identity resource according to the steps in **Create a user-assigned managed identity**[120].

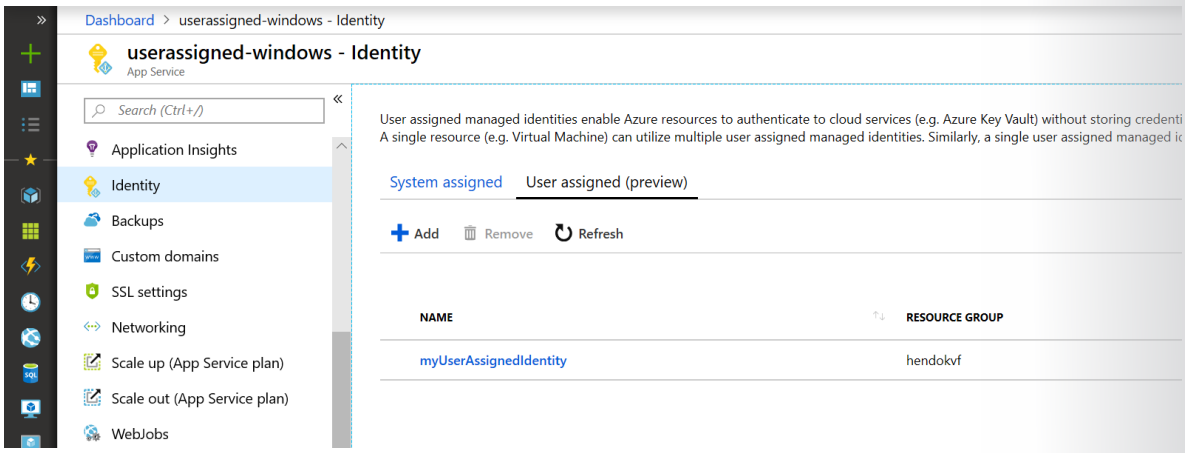
2.Create an app in the Azure portal, and then navigate to it.

3.If you are using a function app, navigate to **Platform features**. For other app types, scroll down to the Settings group in the left navigation.

4.Select **Managed identity**.

5.Within the **User assigned** (preview) tab, select **Add**.

6.Search for the identity you created earlier, select it, and then select **Add**.

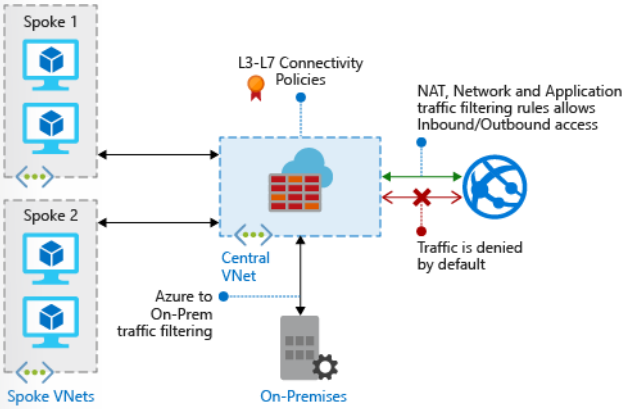


Another option you could use is the Azure Resource Manager template. You can read more about it at **Using an Azure Resource Manager template**[121].

Implement PaaS firewall rules

Azure Firewall is a managed, cloud-based network security service that protects your Azure Virtual Network resources. It's a fully stateful firewall-as-a-service with built-in high availability and unrestricted cloud scalability. By default, Azure Firewall blocks traffic.

117 https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity
118 https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity
119 https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity
120 https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/how-to-manage-ua-identity-portal
121 https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity



The Azure Firewall features include:

- Built-in high availability. Because high availability is built in, no additional load balancers are required and there's nothing you need to configure.
- Unrestricted cloud scalability. Azure Firewall can scale up as much as you need, to accommodate changing network traffic flows so you don't need to budget for your peak traffic.
- Application FQDN filtering rules. You can limit outbound HTTP/S traffic to a specified list of FQDNs, including wild cards. This feature does not require SSL termination.
- Network traffic filtering rules. You can centrally create allow or deny network filtering rules by source and destination IP address, port, and protocol. Azure Firewall is fully stateful, so it can distinguish legitimate packets for different types of connections. Rules are enforced and logged across multiple subscriptions and virtual networks.
- FQDN tags. FQDN tags make it easier for you to allow well known Azure service network traffic through your firewall. For example, say you want to allow Windows Update network traffic through your firewall. You create an application rule and include the Windows Update tag. Now network traffic from Windows Update can flow through your firewall.
- Outbound Source Network Address Translation (OSNAT) support. All outbound virtual network traffic IP addresses are translated to the Azure Firewall public IP. You can identify and allow traffic originating from your virtual network to remote internet destinations.
- Inbound Destination Network Address Translation (DNAT) support. Inbound network traffic to your firewall public IP address is translated and filtered to the private IP addresses on your virtual networks.
- Azure Monitor logging. All events are integrated with Azure Monitor, allowing you to archive logs to a storage account, stream events to your Event Hub, or send them to Azure Monitor logs.

Grouping the features above into logical groups reveals that Azure Firewall has three rule types: NAT rules, network rules, and application rules. The application order precedence for the rules are that network rules are applied first, then application rules. Rules are terminating, which means if a match is found in network rules, then application rules are not processed. If there's no network rule match, and if the packet protocol is HTTP/HTTPS,
the packet is then evaluated by the application rules. If no match continues to be found, then the packet is evaluated against the infrastructure rule collection. If there's still no match, then the packet is denied by default.

NAT rules
You can configure inbound connectivity by configuring Destination Network Address Translation (DNAT) as

described in the tutorial:**Filter inbound traffic with Azure Firewall DNAT using the Azure portal.**[122] DNAT rules are applied first.
If a match is found, an implicit corresponding network rule to allow the translated traffic is added. You can
override this behavior by explicitly adding a network rule collection with deny rules that match the translated traffic.
No application rules are applied for these connections.

Firewall rules to secure Azure Storage

Azure Storage provides a layered security model, which enables you to secure your storage accounts to a specific set of supported networks. When network rules are configured, only applications requesting data from over the specified set of networks can access a storage account.

An application that accesses a storage account when network rules are in effect requires proper authorization on the request. Authorization is supported with Azure AD credentials (for blobs and queues) (preview), a valid account access key, or a SAS token.

By default, storage accounts accept connections from clients on any network. To limit access to selected networks, you must first change the default action. Making changes to network rules can impact your applications' ability to connect to Azure Storage. Setting the default network rule to Deny blocks all access to the data unless specific network rules that grant access are also applied. Be sure to grant access to any allowed networks using network rules before you change the default rule to deny access.

Exercise

In this exercise you will manage default network access rules for storage accounts through the Azure portal, PowerShell,
or Azure CLI. For this exercise we will use the Azure portal.

1.Sign in to the Azure Portal

2.Navigate to the storage account you want to secure.

3.Select the settings menu called Firewalls and virtual networks.

4.To deny access by default, choose to allow access from Selected networks. To allow traffic from all networks, choose to allow access from All networks.

5.Select Save to apply your changes.

For information on the other two methods, see:

- **Configure firewalls and virtual networks: PowerShell**[123]
- **Configure firewalls and virtual networks: CLIv2**[124]

Grant access from a virtual network

You can configure storage accounts to allow access only from specific VNets.

You enable a service endpoint for Azure Storage within the VNet. This endpoint gives traffic an optimal route to the Azure Storage service. The identities of the virtual network and the subnet are also transmitted with each request. Administrators can then configure network rules for the storage account that allow requests to be received from specific subnets in the VNet. Clients granted access via these network rules must continue to meet the authorization requirements of the storage account to access the data.

122 https://docs.microsoft.com/en-us/azure/firewall/tutorial-firewall-dnat
123 https://docs.microsoft.com/en-us/azure/storage/common/storage-network-security
124 https://docs.microsoft.com/en-us/azure/storage/common/storage-network-security

Each storage account supports up to 100 virtual network rules, which could be combined with IP network rules.

More information is available at:

- **Virtual Network Service Endpoints**[125]
- **Configure firewalls and virtual networks: Grant access from an internet IP range**[126]

Exercise

Although you can manage virtual network rules for storage accounts through the Azure portal, PowerShell, or CLIv2, for this exercise we will use the Azure portal:

1.Go to the storage account you want to secure.

2.Select the settings menu called **Firewalls and virtual networks**.

3.Verify that you've selected to allow access from **Selected network**s.

4.To grant access to a virtual network with a new network rule, under **Virtual networks**, select **Add existing virtual network**, select both the **Virtual networks** and **Subnets** options, and then select **Add**.

5.To create a new virtual network and grant it access, select **Add new virtual network**. Provide the information necessary to create the new virtual network, and then select **Create**.

6.To remove a virtual network or subnet rule, select the ellipses (...) to open the context menu for the virtual network or subnet, and then select **Remove**.

7.Select **Save** to apply your changes.

Storage analytics data access

In some cases, access to read diagnostic logs and metrics is required from outside the network boundary. You can grant exceptions to the network rules to allow read-access to storage account log files, metrics tables, or both.

Exercise

Although you can manage network rule exceptions through the Azure portal, PowerShell, or Azure CLI v2, for this exercise we will use the Azure portal:

1.Go to the storage account you want to secure.

2.Select the settings menu called **Firewalls and virtual networks**.

3.Verify that you've selected to allow access from **Selected networks**.

4.Under **Exceptions**, select the exceptions you want to grant.

5.Select **Save** to apply your changes.

Configure Azure services to protect web apps

One way to protect an Azure service is to make it highly available, very responsive, and monitor its performance for
unresponsiveness. **Azure Front Door Service (AFD)** offers these protections.

AFD enables you to define, manage, and monitor the global routing for your web traffic by optimizing for best performance and instant global failover for high availability. With AFD, you can transform your

125 https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-service-endpoints-overview
126 https://docs.microsoft.com/en-us/azure/storage/common/storage-network-security

multiple-region consumer and enterprise applications into robust, high-performance, personalized modern applications, APIs, and content that reaches a global audience with Azure.

AFD works at Layer 7 or HTTP/HTTPS layer, and uses anycast protocol with split TCP and Microsoft's global network for improving global connectivity. Based on your routing method selection in the configuration, you can ensure that AFD is routing your client requests to the fastest and most available application backend. An *application backend* is any internet-facing service hosted inside or outside of Azure.

AFD provides a range of traffic-routing methods and backend health monitoring options to suit different application needs and automatic failover models. Similar to Traffic Manager, AFD is resilient to failures, including an entire Azure region failure. AFD is not a new service; it was developed years ago to enhance its Bing and Microsoft Office 365 services.

Exercise

Create an AFD for a highly available global web application

This exercise includes two instances of a web application running in different Azure regions. An AFD configuration based on equal weighted and same priority backends is created that helps direct user traffic to the nearest set of site backends running the application. AFD continuously monitors the web application and provides automatic failover to the next available backend when the nearest site is unavailable.

Prerequisites

This exercise requires that you have deployed two instances of a web application running in different Azure regions (East US and West Europe). Both the web application instances run in Active/Active mode, which means either of them can take traffic at any time, unlike an Active/Stand-By configuration where one acts as a failover.

To create an AFD for a highly available global web application:

1.Sign in to the Azure portal.

2.On the top, left side of the screen, select **Create a resource > Web > Web App > Create**.

3.In **Web App**, enter or select the following information, and enter default settings where none are specified.

Setting	Value
Name	Enter a unique name for your web app.
Resource group	Select New, and then type myResourceGroupFD1.
App Service plan/Location	Select New. In the App Service plan, enter myAppServicePlanEastUS, and then select OK.
Location	Enter East US.

4.Select **Create**. A default website is created when the Web App is successfully deployed.

5.Repeat steps 2-4 to create a second website in a different Azure region with the following settings:

Setting	Value
Name	Enter a unique name for your web app.
Resource Group	Select New, and then type myResourceGroupFD2.
App Service plan/Location	Select New. In the App Service plan, enter myAppServicePlanWestEurope, and then select OK.
Location	Enter West Europe.

Create an AFD for the web apps

Add a frontend host for AFD

To create an AFD configuration that directs user traffic based on lowest latency between the two backends, complete the following steps:

1.On the top, left side of the screen, select **Create a resource > Networking > Front Door > Create**.

2.In the **Create a Front Door**, you start by adding the basic information and providing a subscription where you want AFD to be configured. Similarly, like any other Azure resource you also need to provide a ResourceGroup, and a Resource Group region if you are creating a new one. Lastly, you need to provide a name for your AFD.

3.After you have filled in the basic information, you need to define the frontend host for the configuration. The result should be a valid domain name such as myappfrontend.azurefd.net. This hostname needs to be globally unique, but AFD will manage the validation.

Add application backend and backend pools

Next, you need to configure your application backend (or backends) in a backend pool for AFD to know where your application resides:

1.Select the plus (+) icon to add a backend pool, and then specify a name for your backend pool, such as myBackendPool.

2.Next, select **Add Backends** to add your websites created earlier.

3.Select **Target host type** as **App Service**, select the subscription in which you created the website, and then choose the first website from the **Target host name**, for example myAppServicePlanEastUS.azurewebsites.net.

4.Leave the remaining fields as is for now, and select **Add**.

5.Repeat steps 2 through 4 to add the other website, myAppServicePlanWestEurope.azurewebsites.net.

6.You can optionally choose to update the health probes and load balancing settings for the backend pool, but the default values should also work.

7.Select **Add**.

Add a routing rule

1.Finally, select the plus (+) icon on Routing rules to configure a routing rule. This is needed to map your frontend host to the backend pool, which basically is configuring that if a request comes to myappfrontend.azurefd.net, then forward it to the backend pool myBackendPool.

2.Select **Add** to add the routing rule for your AFD.

3.You should now be ready to create the AFD, so select **Review**, and then select **Create**.

Note: You must ensure that each of the frontend hosts in your AFD has a routing rule with a default path ('/') *associated with it. That is, across all of your routing rules there must be at least one routing rule for each of your frontend hosts defined at the default path ('/'*). Failing to do so could result in your end-user traffic not getting routed correctly.

View AFD in action

After you create an AFD, it will take a few minutes for the configuration to be deployed globally. After configuration finishes deploying, open a web browser and type the URL myappfrontend.azurefd.net to access the frontend host you created. Your request will automatically get routed to the nearest backend to you from the specified backends in the backend pool.

View AFD handle application failover

If you want to test AFD's instant global failover in action, you can go to one of the websites you created and stop it. Based on the health probe setting defined for the backend pool, the traffic will instantly fail over to the other website deployments. You can also test behavior by disabling the backend in the backend pool configuration for your AFD.

Exercise summary

In this exercise, you created an AFD that allows you to direct user traffic for web applications that require high availability and maximum performance. To learn more about routing traffic, read **Front Door routing methods**[127].

Clean up resources

When no longer needed, delete the resource groups, web applications, and all related resources.

Configure Azure Application Security Groups

In this lesson we look at Application Security Groups (ASGs), which are built on network security groups. A quick review of Security groups reminds us that You can filter network traffic to and from Azure resources in an Azure virtual network with a network security group. A network security group contains security rules that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources.

For each rule, you can specify source and destination, port, and protocol. You can enable network security group flow logs to analyze network traffic to and from resources that have an associated network security group.

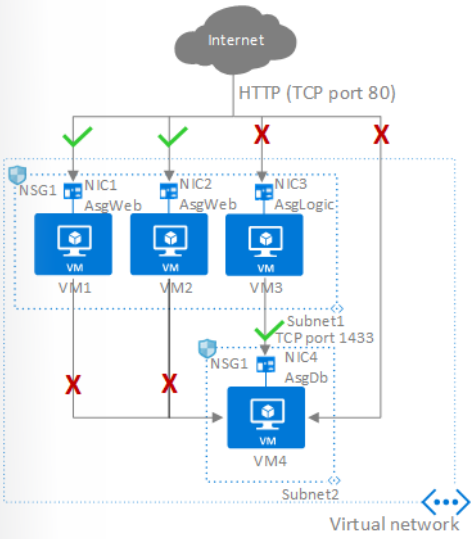
- To learn about which Azure resources can be deployed into a virtual network and have network security groups associated to them, see **Virtual network integration for Azure services.**[128]
- If you've never created a network security group, to get experience creating one you can complete a quick tutorial here: **Tutorial: Filter network traffic with a network security group using the Azure Portal**[129].
- If you're familiar with network security groups and need to manage them, see **Create, change, or delete a network security group**[130].
- If you're having communication problems and need to troubleshoot network security groups, see **Diagnose a virtual machine network traffic filter problem**[131].
- You can read more about enabling network security group flow logs to analyze network traffic to and from resources that have an associated network security group at:
  - **Tutorial: Log network traffic to and from a virtual machine using the Azure portal**[132]
  - **Traffic Analytics**[133]

ASGs

127 https://docs.microsoft.com/en-us/azure/frontdoor/front-door-routing-methods
128 https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-for-azure-services
129 https://docs.microsoft.com/en-us/azure/virtual-network/tutorial-filter-network-traffic
130 https://docs.microsoft.com/en-us/azure/virtual-network/manage-network-security-group
131 https://docs.microsoft.com/en-us/azure/virtual-network/diagnose-network-traffic-filter-problem
132 https://docs.microsoft.com/en-us/azure/network-watcher/network-watcher-nsg-flow-logging-portal?toc=%2fazure%2fvirtual-network%2ftoc.json
133 https://docs.microsoft.com/en-us/azure/network-watcher/traffic-analytics?toc=%2fazure%2fvirtual-network%2ftoc.json

ASGs enable you to configure network security as a natural extension of an application's structure. You then can group VMs
and define network security policies based on those groups.

You also can reuse your security policy at scale without manual maintenance of explicit IP addresses. The platform manages
the complexity of explicit IP addresses and multiple rule sets, allowing you to focus on your business logic. Consider the following illustration.



In the illustration, NIC1 and NIC2 are members of the AsgWeb ASG. NIC3 is a member of the AsgLogic ASG. NIC4 is a member of the AsgDb ASG.
Though each network interface in this example is a member of only one ASG, a network interface can be a member of multiple ASGs, up
to the Azure limits. None of the network interfaces have an associated network security group. NSG1 is associated to both subnets
and contains the following rules:

- **Allow-HTTP-Inbound-Internet** - This rule is needed to allow traffic from the internet to the web servers.
  Because inbound traffic from the internet is denied by the DenyAllInbound default security rule, no additional rule is
  needed for the AsgLogic or AsgDb application security groups.
- **Deny-Database-All** - Because the AllowVNetInBound default security rule allows all communication between resources
  in the same virtual network, this rule is needed to deny traffic from all resources.
- **Allow-Database-BusinessLogic** - This rule allows traffic from the AsgLogic application security group to the AsgDb
  application security group. The priority for this rule is higher than the priority for the Deny-Database-All rule. As a
  result, this rule is processed before the Deny-Database-All rule, so traffic from the AsgLogic application security group
  is allowed, whereas all other traffic is blocked.

The rules that specify an ASG as the source or destination are only applied to the network interfaces that are members of the ASG. If the network interface is not a member of an ASG, the rule is not applied to the network interface even though the network security group is associated to the subnet.

ASGs have the following constraints:

- There are limits to the number of ASGs you can have in a subscription, in addition to other limits related to ASGs.
- You can specify one ASG as the source and destination in a security rule. You cannot specify multiple ASGs in the source or destination.
- All network interfaces assigned to an ASG have to exist in the same virtual network that the first network interface assigned to the ASG is in. For example, if the first network interface assigned to an ASG named AsgWeb is in the virtual network named VNet1, then all subsequent network interfaces assigned to ASGWeb must exist in VNet1. You cannot add network interfaces from different virtual networks to the same ASG.
- If you specify an ASG as the source and destination in a security rule, the network interfaces in both ASGs must exist in the same virtual network. For example, if AsgLogic contained network interfaces from VNet1, and AsgDb contained network interfaces from VNet2, you could not assign AsgLogic as the source and AsgDb as the destination in a rule. All network interfaces for both the source and destination ASGs need to exist in the same virtual network.

Configure and Manage Azure Key Vault

Introduction

Protecting your keys is essential to protecting your data in the cloud.

Azure Key Vault helps safeguard cryptographic keys and secrets that cloud applications and services use. Key Vault streamlines the key management process and enables you to maintain control of keys that access and encrypt your data. Developers can create keys for development and testing in minutes, and then migrate them to production keys. Security administrators can grant (and revoke) permission to keys, as needed.

You can use Key Vault to create multiple secure containers, called vaults. Vaults help reduce the chances of accidental loss of security information by centralizing application secrets storage. Key vaults also control and log the access to anything stored in them.

Azure Key Vault can manage requesting and renewing TLS certificates. It provides features for a robust solution for certificate lifecycle management.

Azure Key Vault helps address the following issues:

- Secrets management. You can use Azure Key Vault to securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets.
- Key management. You use Azure Key Vault as a key management solution, making it easier to create and control the encryption keys used to encrypt your data.
- Certificate management. Azure Key Vault is also a service that lets you easily provision, manage, and deploy public and private SSL/TLS certificates for use with Azure and your internal connected resources.
- Store secrets backed by hardware security modules (HSMs). The secrets and keys can be protected either by software, or FIPS 140-2 Level 2 validates HSMs.

Azure Key Vault is designed to support application keys and secrets. Key Vault is not intended as storage for user passwords.

The following table lists security best practices for using Key Vault.

Best practice	Solution
Grant access to users, groups, and applications at a specific scope.	Use RBAC's predefined roles. For example, to grant access to a user to manage key vaults, you would assign the predefined role Key Vault Contributor to this user at a specific scope. The scope in this case would be a subscription, a resource group, or just a specific key vault. If the predefined roles don't fit your needs, you can define your own roles.

Best practice	Solution
Control what users have access to.	Access to a key vault is controlled through two separate interfaces: management plane, and data plane. The management plane and data plane access controls work independently. Use RBAC to control what users have access to. For example, if you want to grant an application access to use keys in a key vault, you only need to grant data plane access permissions by using key vault access policies, and no management plane access is needed for this application. Conversely, if you want a user to be able to read vault properties and tags but not have any access to keys, secrets, or certificates, you can grant this user read access by using RBAC, and no access to the data plane is required.
Store certificates in your key vault.	Azure Resource Manager can securely deploy certificates stored in Azure Key Vault to Azure VMs when the VMs are deployed. By setting appropriate access policies for the key vault, you also control who gets access to your certificate. Another benefit is that you manage all your certificates in one place in Azure Key Vault.
Ensure that you can recover a deletion of key vaults or key vault objects.	Deletion of key vaults or key vault objects can be either inadvertent or malicious. Enable the soft delete and purge protection features of Key Vault, particularly for keys that are used to encrypt data at rest. Deletion of these keys is equivalent to data loss, so you can recover deleted vaults and vault objects if needed. Practice Key Vault recovery operations on a regular basis.

Note: If a user has contributor permissions (RBAC) to a key vault management plane, they can grant themselves access to the data plane by setting a key vault access policy. We recommend that you tightly control who has contributor access to your key vaults, to ensure that only authorized persons can access and manage your key vaults, keys, secrets, and certificates.

Configure an HSM key-generation solution

For added assurance, when you use Azure Key Vault, you can import or generate keys in hardware security modules (HSMs) that never leave the HSM boundary. This scenario is often referred to as Bring Your Own Key (BYOK). The HSMs are FIPS 140-2 Level 2 validated. Azure Key Vault uses Thales nShield family of HSMs to protect your keys. (This functionality is not available for Azure China.)

More information about generating and transferring an HSM-protected key over the Internet:

1.You generate the key from an offline workstation, which reduces the attack surface.

2.The key is encrypted with a Key Exchange Key (KEK), which stays encrypted until transferred to the Azure Key Vault HSMs. Only the encrypted version of your key leaves the original workstation.

3.The toolset sets properties on your tenant key that binds your key to the Azure Key Vault security world. After the Azure Key Vault HSMs receive and decrypt your key, only these HSMs can use it. Your key cannot be exported. This binding is enforced by the Thales HSMs.

4.The KEK that encrypts your key is generated inside the Azure Key Vault HSMs, and is not exportable. The HSMs enforce that there can be no clear version of the KEK outside the HSMs. In addition, the toolset includes attestation from Thales that the KEK is not exportable and was generated inside a genuine HSM that was manufactured by Thales.

5.The toolset includes attestation from Thales that the Azure Key Vault security world was also generated on a genuine HSM manufactured by Thales.

6.Microsoft uses separate KEKs and separate security worlds in each geographical region. This separation ensures that your key can be used only in data centers in the region in which you encrypted it. For example, a key from a European customer cannot be used in data centers in North American or Asia.

For more information, see **Implementing bring your own key (BYOK) for Azure Key Vault**[134].

If you have access to Thales HSM, smartcards, and support software you can walk through an exercise detailed at the link above. It is suggested to review the steps even if you cannot perform the exercise.

Manage access, and permissions to secrets, certificates, and keys to Key Vault

Access to a key vault is controlled through two interfaces: the management plane, and the data plane. The management plane is where you manage Key Vault itself. Operations in this plane include creating and deleting key vaults, retrieving Key Vault properties, and updating access policies. The data plane is where you work with the data stored in a key vault. You can add, delete, and modify keys, secrets, and certificates from here.

To access a key vault in either plane, all callers (users or applications) must have proper authentication and authorization. Authentication establishes the identity of the caller. Authorization determines which operations the caller can execute.

Both planes use Azure AD for authentication. For authorization, the management plane uses RBAC, and the data plane uses a Key Vault access policy.

Active Directory authentication

When you create a key vault in an Azure subscription, its automatically associated with the Azure AD tenant of the subscription. All callers in both planes must register in this tenant and authenticate to access the key vault. In both cases, applications can access Key Vault in two ways:

- User plus application access. The application accesses Key Vault on behalf of a signed-in user. Examples of this type of access include Azure PowerShell and the Azure portal. User access is granted in two ways. They can either access Key Vault from any application, or they must use a specific application (referred to as compound identity).
- Application-only access. The application runs as a daemon service or background job. The application identity is granted access to the key vault.

For both types of access, the application authenticates with Azure AD. The application uses any supported authentication method based on the application type. The application acquires a token for a resource in the plane to grant access. The resource is an endpoint in the management or data plane, based on the

134 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-hsm-protected-keys

Azure environment. The application uses the token and sends a REST API request to Key Vault. To learn more, review the whole authentication flow.

For more information, see:

- **What is authentication?**[135]
- **Resource endpoints**[136]
- **Authorize access to Azure Active Directory web applications using the OAuth 2.0 code grant flow**[137]

The model of a single mechanism for authentication to both planes has several benefits:

- Organizations can centrally control access to all key vaults in their organization.
- If a user leaves, they instantly lose access to all key vaults in the organization.
- Organizations can customize authentication by using the options in Azure AD, such as to enable multi-factor authentication for added security.

Manage certificates

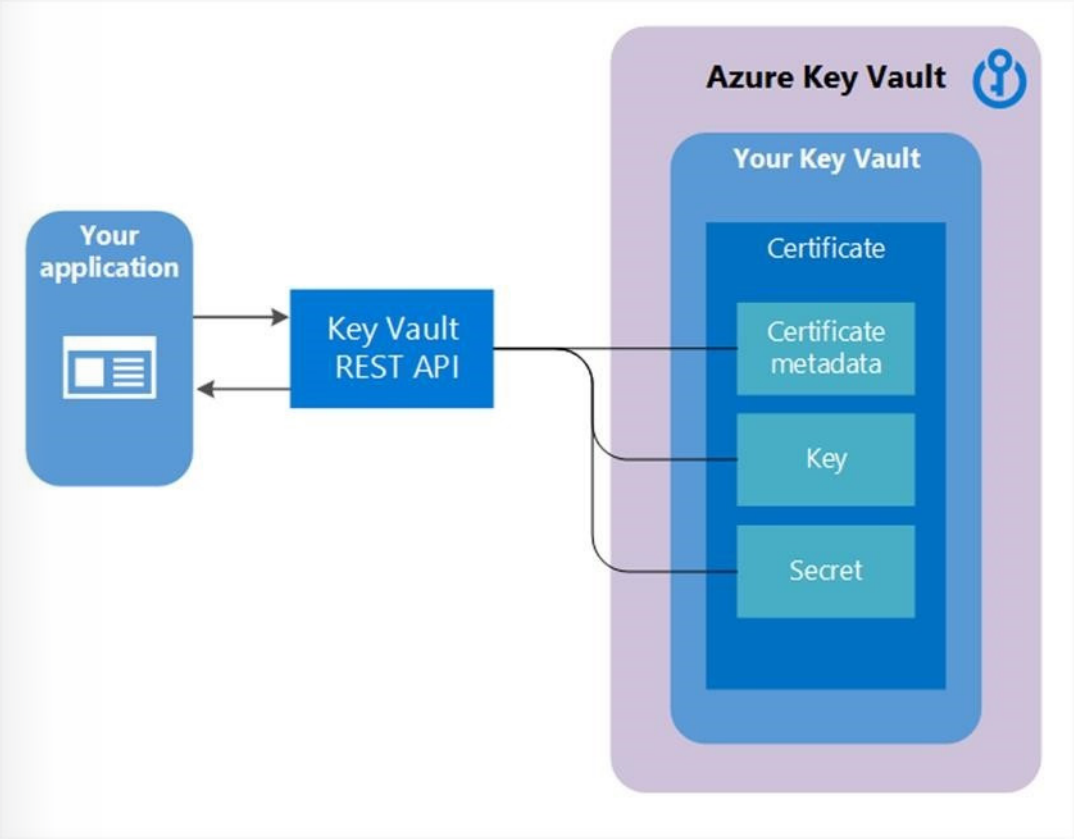
Key Vault certificates support provides for management of your x509 certificates and enables:

- A certificate owner to create a certificate through a Key Vault creation process or through the import of an existing certificate. Includes both self-signed and CA-generated certificates.
- A Key Vault certificate owner to implement secure storage and management of X509 certificates without interaction with private key material.
- A certificate owner to create a policy that directs Key Vault to manage the life-cycle of a certificate.
- Certificate owners to provide contact information for notification about lifecycle events of expiration and renewal of certificate.
- Automatic renewal with selected issuers - Key Vault partner X509 certificate providers and CAs.

When a Key Vault certificate is created, an addressable key and secret are also created with the same name. The Key Vault key allows key operations and the Key Vault secret allows retrieval of the certificate value as a secret. A Key Vault certificate also contains public x509 certificate metadata.

The identifier and version of certificates is similar to that of keys and secrets. A specific version of an addressable key and secret created with the Key Vault certificate version is available in the Key Vault certificate response.

135 https://docs.microsoft.com/en-us/azure/active-directory/develop/authentication-scenarios
136 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-secure-your-key-vault
137 https://docs.microsoft.com/en-us/azure/active-directory/develop/v1-protocols-oauth-code



When a Key Vault certificate is created, it can be retrieved from the addressable secret with the private key in either PFX or PEM format. However, the policy used to create the certificate must indicate that the key is exportable. If the policy indicates non-exportable, then the private key isn't a part of the value when retrieved as a secret.

The addressable key becomes more relevant with non-exportable Key Vault certificates. The addressable Key Vault key's operations are mapped from the keyusage field of the Key Vault certificate policy used to create the Key Vault certificate. If a Key Vault certificate expires, it's addressable key and secret become inoperable.

Two types of key are supported – RSA or RSA HSM with certificates. Exportable is only allowed with RSA, and is not supported by RSA HSM.

Certificate policy

A certificate policy contains information on how to create and manage the Key Vault certificate lifecycle. When a certificate with private key is imported into the Key Vault, a default policy is created by reading the x509 certificate.

When a Key Vault certificate is created from scratch, a policy needs to be supplied. This policy specifies how to create the Key Vault certificate version, or the next Key Vault certificate version. After a policy has been established, it's not required with successive create operations for future versions. There's only one instance of a policy for all the versions of a Key Vault certificate.

At a high level, a certificate policy contains the following information:

- X509 certificate properties. Contains subject name, subject alternate names, and other properties used to create an x509 certificate request.

- Key Properties. Contains key type, key length, exportable, and reuse key fields. These fields instruct key vault on how to generate a key.
- Secret properties. Contains secret properties such as content type of addressable secret to generate the secret value, for retrieving certificate as a secret.
- Lifetime Actions. Contains lifetime actions for the Key Vault certificate. Each lifetime action contains:
  - Trigger, which specifies via days before expiry or lifetime span percentage.
  - Action, which specifies the action type: emailContacts, or autoRenew.
- Issuer: Contains the parameters about the certificate issuer to use to issue x509 certificates.
- Policy attributes: Contains attributes associated with the policy.

Certificate Issuer

Before you can create a certificate issuer in a Key Vault, the following two prerequisite steps must be completed successfully:

1.Onboard to CA providers:

- An organization administrator must onboard their company with at least one CA provider.

2.Admin creates requester credentials for Key Vault to enroll (and renew) SSL certificates:

- Provides the configuration to be used to create an issuer object of the provider in the key vault.

Certificate contacts

Certificate contacts contain contact information to send notifications triggered by certificate lifetime events. The contacts information is shared by all the certificates in the key vault. A notification is sent to all the specified contacts for an event for any certificate in the key vault.

If a certificate's policy is set to auto renewal, then a notification is sent for the following events:

- Before certificate renewal
- After certificate renewal, and stating if the certificate was successfully renewed, or if there was an error, requiring manual renewal of the certificate
- When it’s time to renew a certificate for a certificate policy that is set to manually renew (email only)

Certificate access control

The Key Vault that contains certificates manages access control for those same certificates. The access control policy for certificates is distinct from the access control policies for keys and secrets in the same Key Vault. Users might create one or more vaults to hold certificates, to maintain scenario appropriate segmentation and management of certificates.

The following permissions closely mirror the operations allowed on a secret object, and can be used on a per-principal basis in the secrets access control entry on a key vault:

- Permissions for certificate management operations:
  - get: Get the current certificate version, or any version of a certificate.
  - list: List the current certificates, or versions of a certificate.
  - update: Update a certificate.
  - create: Create a Key Vault certificate.
  - import: Import certificate material into a Key Vault certificate.
  - delete: Delete a certificate, its policy, and all of its versions.

- recover: Recover a deleted certificate.
- backup: Back up a certificate in a key vault.
- restore: Restore a backed-up certificate to a key vault.
- managecontacts: Manage Key Vault certificate contacts.
- manageissuers: Manage Key Vault certificate authorities/issuers.
- getissuers: Get a certificate's authorities/issuers.
- listissuers: List a certificate's authorities/issuers.
- setissuers: Create or update a Key Vault certificate's authorities/issuers.
- deleteissuers: Delete a Key Vault certificate's authorities/issuers.

- Permissions for privileged operations:
  - purge: Purge (permanently delete) a deleted certificate.

For more information, see **Azure Key Vault REST API**[138].

For information on establishing permissions, see:

- **Vaults - Create Or Update**[139]
- **Vaults - Update Access Policy**[140]

Upload a secret

Azure Key Vault is a cloud service that works as a secure secrets store. You can securely store keys, passwords, certificates, and other secrets.

Exercise

In this exercise, you use PowerShell to create a key vault. You then store a secret in the newly created vault. However,
instead of typing all of the PowerShell commands, you are provided with **Copy & Try It**.

To complete the exercise, follow the steps at **Quickstart: Set and retrieve a secret from Azure Key Vault using PowerShell**[141].

Note: The same Quickstart location includes steps for performing the same tasks using Azure CLI, Azure portal, .NET, and Node.js

Configure key rotation

After you have a key vault, you can start using it to store keys and secrets. Your applications no longer need to persist your keys or secrets, but can request them from the vault as needed. A key vault allows you to update keys and secrets without affecting the behavior of your application, which opens up a breadth of possibilities for your key and secret management.

Exercise

Perform the exercise steps for the following tasks:

1.Create the secret and upload it to your vault using the steps at **Set up Key Vault**[142].

138 https://docs.microsoft.com/en-us/rest/api/keyvault
139 https://docs.microsoft.com/en-us/rest/api/keyvault/vaults/createorupdate
140 https://docs.microsoft.com/en-us/rest/api/keyvault/vaults/updateaccesspolicy
141 https://docs.microsoft.com/en-us/azure/key-vault/quick-create-powershell
142 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-key-rotation-log-monitoring

2.Set up an application at **Set up the application**[143].

3.Set up a rotation strategy for the values you store as Key Vault secrets using the steps at **Key rotation using Azure Automation**[144].

4.Create a pipeline that sends an email when an app that doesn't match the app ID of the web app retrieves any secrets from the vault using the steps at **Key Vault auditing pipeline**[145].

5.Create an Azure logic app that picks up events, parses the content, and sends an email based on a condition being matched using the steps at **Azure logic app**[146].

Note: Working through all of the exercise steps will take about 1.5 hours.

After completing this exercise, you will now have an end-to-end pipeline that looks for new key-vault audit logs once every minute. It pushes new logs it finds to a Service Bus queue. The logic app is triggered when a new message lands in the queue. If the app ID within the event doesn't match the app ID of the calling application, it sends an email.

Additional Information

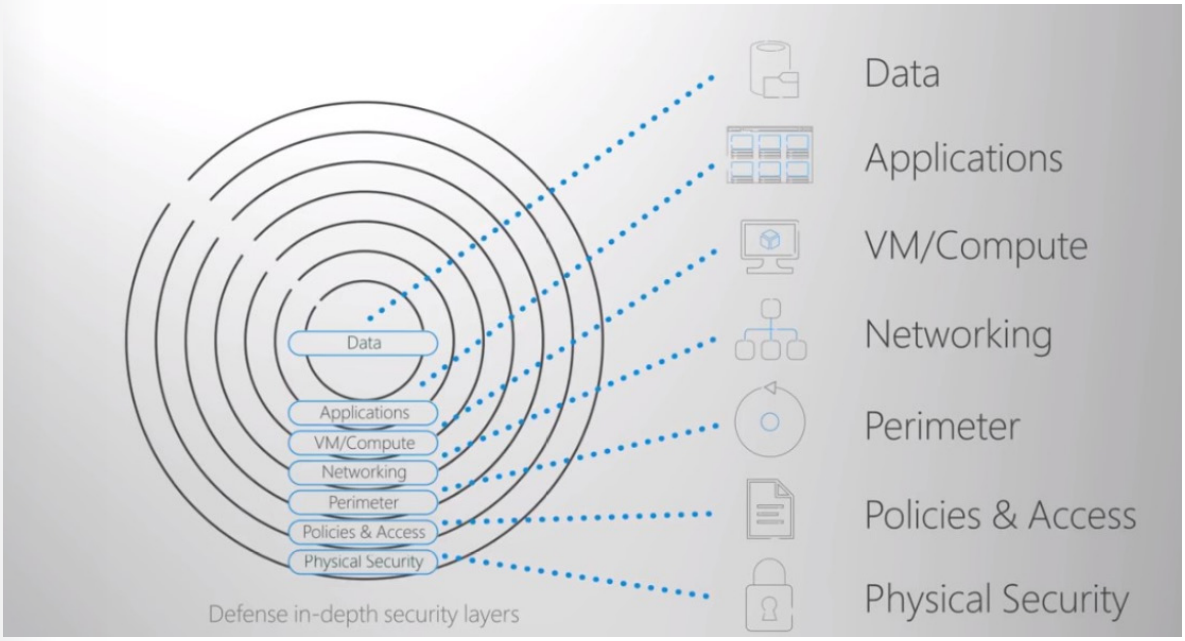
- For a list of the latest Azure PowerShell Certificate cmdlets for Azure Key Vault, see **Azure Key Vault Cmdlets**[147].
- Key Vault certificate management operations are discussed more in depth at **Certificate operations**[148].
- Read more about Key Vault at **What is Azure Key Vault?**[149]

Course Summary

During this course we explored the defense-in-depth design of Azure services and capabilities to help you securely manage and monitor your cloud data and infrastructure as a managed service. Microsoft designs and operates its cloud services with security at the core and provides you built-in controls and tools to meet your security needs. In addition, with Machine Learning (ML) and Microsoft's significant investments in cyber defense you can benefit from unique intelligence and proactive measures to protect you from threats. Azure offers unified security management and advanced threat protection for
your resources whether they're in the cloud, your data center, or both. Services in Azure are built with security in mind from the ground up to host your infrastructure apps and data. All services are designed and operated to support multiple layers of defense, spanning your data apps, virtual machines, network perimeter related policies, and, of course, physical security within our data centers. This includes how the data sensors and systems that run Azure are architected and operated to the controls you can leverage

143 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-key-rotation-log-monitoring
144 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-key-rotation-log-monitoring
145 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-key-rotation-log-monitoring
146 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-key-rotation-log-monitoring
147 https://docs.microsoft.com/en-us/powershell/module/azurerm.keyvault/?view=azurermps-6.13.0
148 https://docs.microsoft.com/en-us/rest/api/keyvault/certificate-operations?redirectedfrom=MSDN
149 https://docs.microsoft.com/en-us/azure/key-vault/key-vault-overview

as part of your defense in-depth security management. This strategy is illustrated in the following image.



As more and more of a company's digital resources reside outside the corporate network, in the cloud and on
personal devices, a great cloud-based identity and access management solution is the best way to maintain
control over and visibility into how and when users access corporate applications and data. Everything starts
with identity and access control and is governed by Azure Active Directory. With Azure AD, you can create and
manage users and groups, and enable permissions to allow and deny access to enterprise resources. The following
cartoon by Stefan Van Der Wiele is a unique illustration of the many functions of Azure AD.

One of the most important built in services to get familiar with is the Azure Security Center. Across Azure services it provides unified visibility and control adaptive threat protection as well as intelligent threat detection response. This gives you centralized real time monitoring into the security state of your dynamic workloads with actionable recommendations and controls.

Your Azure data is what you'll want to protect the most and is often the core of your apps and services. Data protections are built for both structured and unstructured data. For structured data, all data is encrypted at rest and machine learning can be used to practically look for and alert you on potential security vulnerabilities.
These can be related to data encryption enabling, security telemetry, and extensive capabilities in data services themselves.

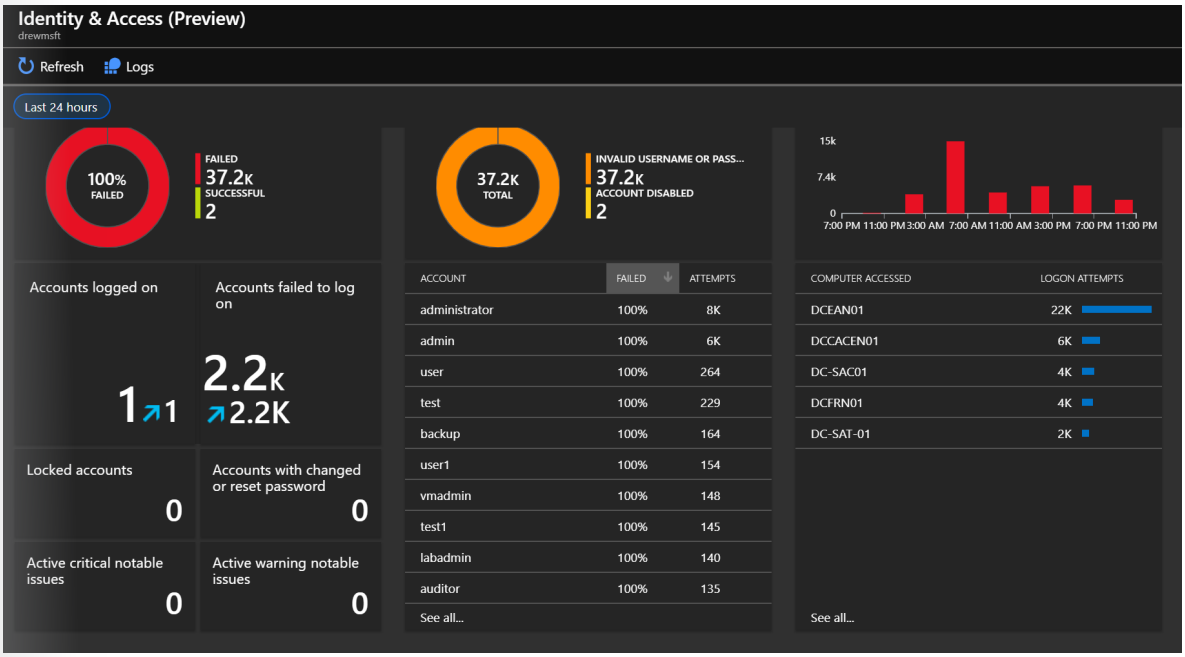
These capabilities can recommend and enable sensitive information discovery and classification for dynamic data masking to obscure data fields. Database services in Azure can be configured to run these checks automatically and
Azure security Center will alert you on any potential issues it finds to keep your data protected. Unstructured data storage accounts spanning blobs, files tables, and queues are also encrypted at rest by default and each account is geographically redundant. Additional protections using access keys to control authentication, shared access signatures for secure delegated access, and granular network firewall controls. Azure Security

Center
will report its findings when Security is at risk.

In Azure, web apps can be configured to use Azure managed service identities to streamline secure communication
with other services. SSL certificates can be managed for your apps, and even require that clients connect into
your apps have valid certificates for inbound requests.

Next are Azure Virtual Machines (VM). Azure Security Center uses machine learning to continuously assess security
and vulnerability levels of your VMs networks and service configurations. It also gives you actionable recommendations
to prevent exploit before they occur and dynamically applies both allow and block lists to keep out unwanted traffic
as you hope for visibility and control. One thing to know is that many of these capabilities are extended to VMs in
other clouds and in your data center.

Azure Security Center leverages the Microsoft intelligent security graph to discover and act against attacks. The
graph combines the cyber intelligence Microsoft collects across all of its services along with industry data to
block known attack patterns. Microsoft also gives the control you need to prioritize alerts and incidents that
are important to your organization. Additionally, we give you a unified view for forensics analysis, and the ability to search across all your computer resources. Threat intelligence can be visualized down to the trending
attack techniques and the geographic regions affected. This is shown below. The following screen shot, provided
by Drew Robinson from the incident response team at Microsoft, highlights the need for these Azure security services.



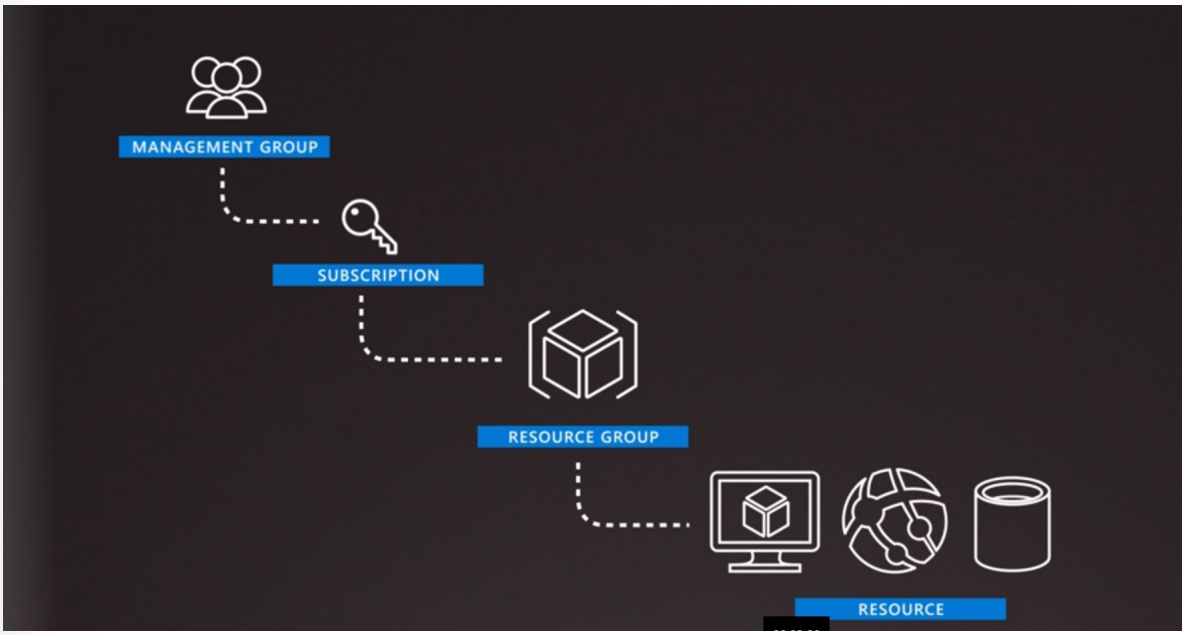
Above is shown the results of some new VMs with RDP services enabled exposed to the Internet. Without monitoring you
would not know there were 37200 login attempts in 24 hours.

Next for network security, the Azure Security Center will assess and report on potential networking security issues
related to open ports and firewall settings and network security groups. Azure provides additional security when
designing and architecting your apps to enforce logical network boundaries and permissions to network security groups.
Network and other resources can be controlled like VMs just-in-time controls for opening management and internet ports
with intelligent recommendations to reduce exposure to brute-force attacks.

Looking at perimeter security, DDoS protection for protection against distributed denial of service attacks is available
in a basic level by default. DDoS protection standard version adds additional protection and mitigations against volumetric
attacks where the attacker's goal is to flood the network with traffic in efforts to disable your services. Protection against
protocol attacks where the attacker tries to find and exploit weaknesses in layer 3 and layer 4 protocol stacks, and application
layer attacks where the web application packets are used to disrupt transmission of data between hosts like cross-site scripting
or HTTP protocol violations.

For security policies and access management Azure has a comprehensive set of services to securely manage security policies
and access to resources where they're accessed by people or programmatically by code. These controls are more than just the
front door to who or which processes can access your apps files or data. They extend how granular access is delegated to
your IT and development teams using role-based access controls to ensure your team members only have access to what they need,

illustrated below.



Identity and Access Management functionality limits Azure administrator to operate with zero standing privilege and use just-in-time approval processes to gain temporary access to sensitive data or controls when needed as your services comply with both international and industry specific compliance standards, and subject to rigorous third party audits that verify Azure security controls to comply with the regulatory requirements, such as GDPR.

Finally, we extend our layered approach to physical security. Data centers managed by Microsoft have extensive layers of protection, access approval at the facilities perimeter, the building's perimeter, inside the building, and on the data center floor. This layered approach reduces the risk of unauthorized users gaining physical access to data and the data center resources.

That was a quick summary of the primary defense and security considerations in Azure covered in this course. At every defense-in-depth layer these controls are all part of the shared responsibility model in Azure comprising the security Microsoft manages as the service provider built-in controls to protect your data, infrastructure, and the intelligence Microsoft can provide from its global scale cyber defense operations.