Microsoft Official Course

AZ-100T01

Managing Azure Subscriptions and Resources

AZ-100T01

Managing Azure Subscriptions and Resources

Contents

Module 0 Welcome

Start Here

Azure Administrator Curriculum

This course is part of a series of courses to help you prepare for Microsoft's Azure Administrator certification tests. There are two exams:

- AZ-100, **Microsoft Azure Infrastructure and Deployment**[1], and
- AZ-101, **Microsoft Azure Integration and Security**[2].

Each exam measures your ability to accomplish certain technical tasks. For example, AZ-100 includes five study areas, as shown in the table. The percentages indicate the relative weight of each area on the exam. The higher the percentage, the more questions you are likely to see in that area.

AZ-100 Study Areas	Weights
Manage Azure subscriptions and resources	**15-20%**
Implement and manage storage	20-25%
Deploy and manage virtual machines	20-25%
Configure and manage virtual networks	20-25%
Manage identities	15-20%

✓ This course will focus on preparing you for the **Manage Azure subscriptions and resources** area of the AZ-100 certification exam.

About This Course

Course Description

This course teaches IT Professionals how to manage their Azure subscriptions, including access, policies, and compliance, as well as how to track and estimate service usage and related costs. Students also learn how cloud resources are managed in Azure through user and group accounts. Students learn how to

[1] https://www.microsoft.com/en-us/learning/exam-az-100.aspx

[2] https://www.microsoft.com/en-us/learning/exam-az-101.aspx

grant appropriate access to Azure AD users, groups, and services through Role-based access control (RBAC). Students also discover the core monitoring tools and capabilities provided by Azure, including Azure Alerts and Activity Log. Students are then introduced to Log Analytics as a broad data analytics solution, and use this service to query and analyze operational data. Students then learn about the Azure Resource Manager deployment model, and how to work with resources, resource groups and ARM templates.

Because this course is the first course in the series for the Azure Administrator exams, there is a considerable amount of foundational content that is covered here in order to prepare students for the remaining courses in the curriculum. So students are provided with a lesson that covers tips and tricks for working in the Azure portal, as well as an introduction to key tools used in the Azure environment, such as the Cloud Shell and Resource Explorer. Emphasis is laid on PowerShell and the command line interface (CLI) as important skills to acquire not only in preparation for the exam but for the job role itself.

Level: Intermediate

Audience

This course is for Azure Administrators. Azure Administrators manage the cloud services that span storage, networking, and compute cloud capabilities, with a deep understanding of each service across the full IT lifecycle. They take end-user requests for new cloud applications and make recommendations on services to use for optimal performance and scale, as well as provision, size, monitor and adjust as appropriate. This role requires communicating and coordinating with vendors. Azure Administrators use the Azure Portal and as they become more proficient they use PowerShell and the Command Line Interface.

Prerequisites

Successful Azure Administrators start this role with experience on operating systems, virtualization, cloud infrastructure, storage structures, and networking.

Expected learning

- Manage Azure subscriptions and billing, and implement Azure policies.
- Implement access management with Azure users, groups, and role-based access control.
- Use Azure Monitor to configure Azure alerts and review the Azure Activity Log.
- Query and analyze Log Analytics data.
- Deploy resources with ARM templates and organize Azure resources.
- Optimize your use of Azure tools like the Azure portal, Azure PowerShell, Cloud Shell and the Azure CLI.

Syllabus

This course includes content that will help you prepare for the certification exam. Other content is included to ensure you have a complete picture of Azure subscriptions and resources. The course content includes a mix of videos, graphics, reference links, module review questions, and practice labs.

Module 1 – Managing Azure Subscriptions

In this module, you'll learn about the components that make up an Azure subscription and how management groups are used to organize subscriptions into containers to allow you to control organizational governance and policy management across subscriptions. As well as learning about the different available types of subscription, you'll see how to apply tags to your Azure resources to logically organize them by categories. Lessons in this module include:

- Overview of Azure Subscriptions
- Billing
- Azure Policy

Module 2 – Access Management for Cloud Resources

In this module you will learn the basics of role-based access control as it applies to users and groups. Focus on the administrator role and how it used in Azure. Lessons include:

- Azure Users and Groups
- Role-based Access Control

✓ The Managing Identities course provides additional coverage of Azure AD access management.

Module 3 – Monitoring and Diagnostics

In this module, you learn about the Azure Monitor and the many capabilities to ensure your Azure architecture is working correctly. Monitoring skills are explained in this first course and then demonstrated in the following courses. The two main elements explained in this module are Azure Alerts and Azure Activity Log. Lessons include:

- Exploring Monitoring Capabilities in Azure
- Azure Alerts
- Azure Activity Log

Module 4 – Log Analytics

In this module, you will focus on Log Analytics. Log Analytics provides a way for you to collect, analyze, and query all types of connected data. It is a very powerful tool and the lessons include:

- Introduction to Log Analytics
- Querying and Analyzing Log Analytics Data

Module 5 – Azure Resource Manager

In this module, you will learn about how resources are organized into resource groups and how ARM templates are used to deploy those resources. This module introduces the concepts and then they are applied in the other courses. Lessons include:

- ARM Templates
- Resource Groups

Module 6 – Azure Tips, Tricks, and Tools

This last module is provided to help you get the most from your administrative tools. This include the Azure Portal, Cloud Shell, Azure CLI, Azure PowerShell, and Resource Explorer. Take time to master these two lessons:

- Azure Portal
- Azure Tools and Environment

Study Guide

The Manage Azure subscriptions and resources objective of the AZ-100 exam, consists of three main areas of study: Manage Azure subscriptions and resources, Analyze resource utilization and consumption, and Manage resource groups. These tables show you what may be included in each test area and where it is covered in this course.

✓ We recommend you use these tables as a checklist to ensure you are prepared in each area.

✓ We also recommend supplementing your study with a **practice test.**[3] Also, hands-on practice is critical to understanding these concepts and passing the certification exams. There are several ways to get an **Azure subscription**[4].

Manage Azure subscriptions and resources

Testing May Include	Course Content
Administer administrator permissions	Module 2 - Access Management for Cloud Resources
Configure cost center quotas and tags	Module 1 - Managing Azure Subscriptions
Configure Azure subscription policies at Azure subscription level	Module 1 - Managing Azure Subscriptions

Analyze resource utilization and consumption

Testing May Include	Course Content
Configure diagnostic settings on resources	Module 3 - Monitoring and Diagnostics
Create baseline for resources	Module 5 - Azure Resource Manager
Create and test alerts	Module 3 - Monitoring and Diagnostics
Analyze alerts across subscriptions	Module 3 - Monitoring and Diagnostics
Analyze metrics across subscriptions	Module 3 - Monitoring and Diagnostics
Create action groups	Module 3 - Monitoring and Diagnostics
Monitor for unused resources	Module 1 - Managing Azure Subscriptions
Report on spend	Module 1 - Managing Azure Subscriptions
Utilize Log Search query functions	Module 4 - Log Analytics
View Alerts in Log Analytics	Module 4 - Log Analytics

Manage resource groups

Testing May Include	Course Content
Use Azure policies for resource groups	Module 1 - Managing Azure Subscriptions
Configure resource locks	Module 5 - Azure Resource Manager
Configure resource policies	Module 1 - Managing Azure Subscriptions
Implement/Set tagging on resource groups	Module 1 - Managing Azure Subscriptions
Move resources across resource groups	Module 5 - Azure Resource Manager
Remove resource groups	Module 5 - Azure Resource Manager

[3] https://us.mindhub.com/az-100-microsoft-azure-infrastructure-deployment-microsoft-official-practice-test/p/MU-AZ-100

[4] https://azure.microsoft.com/en-us/offers/ms-azr-0044p/

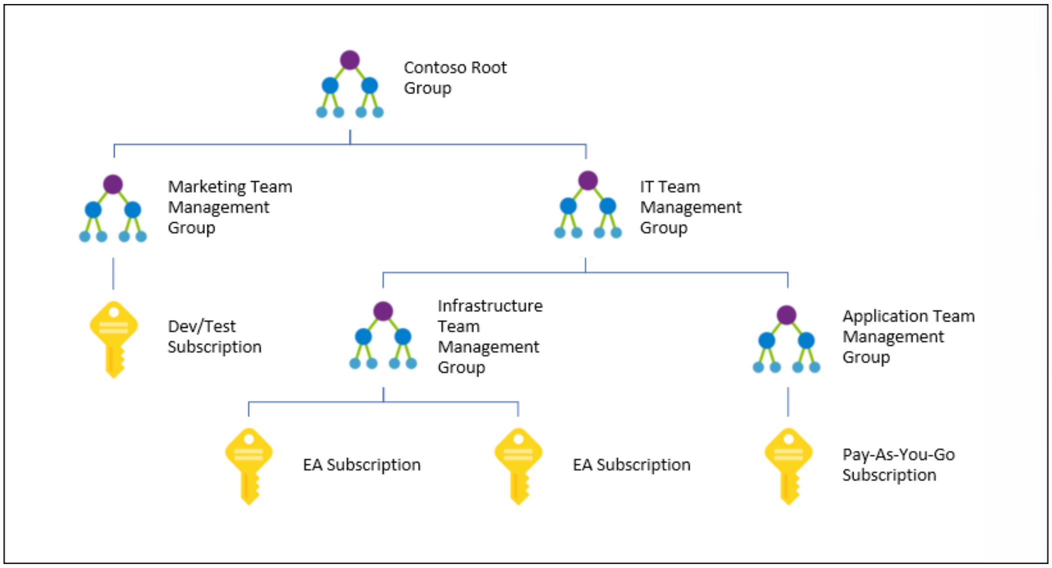
Module 1 Managing Azure Subscriptions

Overview of Azure Subscriptions

Management Groups

If your organization has several subscriptions, you may need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups provide a level of scope above subscriptions. You organize subscriptions into containers called "management groups" and apply your governance conditions to the management groups. Management group enable:

- Organizational alignment for your Azure subscriptions through custom hierarchies and grouping.
- Targeting of policies and spend budgets across subscriptions and inheritance down the hierarchies.
- Compliance and cost reporting by organization (business/teams).


Contoso Root Group
Marketing Team Management Group
IT Team Management Group
Dev/Test Subscription
Infrastructure Team Management Group
Application Team Management Group
EA Subscription
EA Subscription
Pay-As-You-Go Subscription

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 limits the regions available for virtual machine (VM) creation. This policy would be applied to all management groups, subscriptions, and resources under that management group by only allowing VMs to be created in that region.

✓ Management groups is a relatively new concept in Azure. Take time to review the reference links.

For more information, you can see:

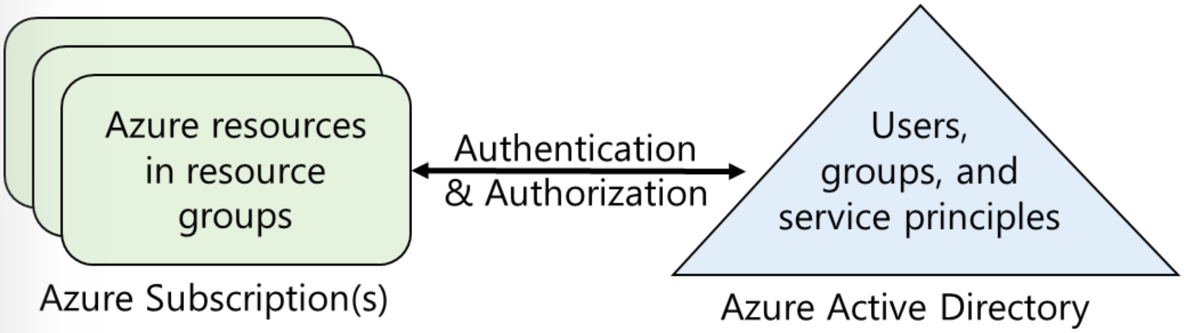
Organize your resources with Azure management groups - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/management-groups-overview**

Create management groups for resource organization and management - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/management-groups-create?toc=/azure/billing/TOC.json**

Azure Subscriptions

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, then you are responsible for 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 the subscription ID may be required for programmatic operations.


Azure resources in resource groups
Authentication & Authorization
Users, groups, and service principles
Azure Subscription(s)
Azure Active Directory

Azure accounts

Subscriptions have accounts. An Azure account is simply an identity in Azure Active Directory (Azure AD) or in 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

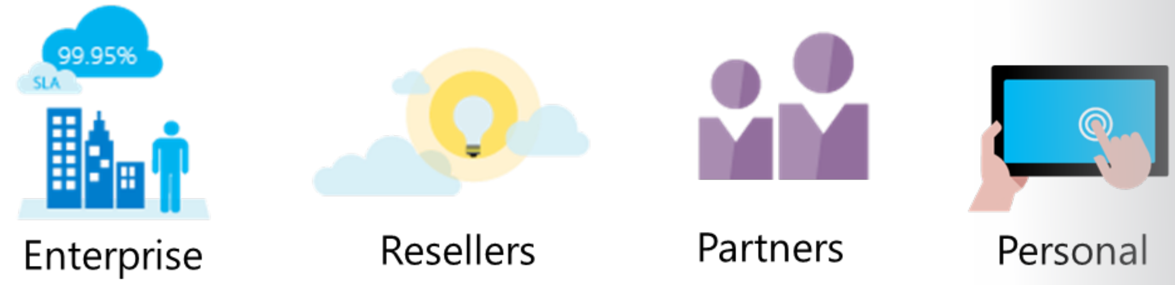
Every Azure subscription is associated with an Azure Active Directory. Users and services that access resources of the subscription first need to authenticate with Azure Active Directory.

Typically to grant a user access to your Azure resources, you would add them to the Azure AD directory associated with your subscription. The user will now have access to all the resources in your subscription. This is an all-or-nothing operation that may give that user access to more resources than you anticipated.

✓ Do you know how many subscriptions your organization has? Do you know how resources are organized into resource groups?

Getting a Subscription

There are several ways to get an Azure subscription: Enterprise agreements, Microsoft resellers, Microsoft partners, and a personal free account.



Enterprise agreements

Any **Enterprise Agreement**[1] customer can add Azure to their agreement by making an upfront monetary commitment to Azure. That commitment is consumed throughout the year by using any combination of the wide variety of cloud services Azure offers from its global datacenters. Enterprise agreements have a 99.95% monthly SLA.

Reseller

Buy Azure through the **Open Licensing program**[2], which provides a simple, flexible way to purchase cloud services from your Microsoft reseller. If you already purchased an Azure in Open license key, **activate a new subscription or add more credits now**[3].

Partners

Find a **Microsoft partner**[4] who can design and implement your Azure cloud solution. These partners have the business and technology expertise to recommend solutions that meet the unique needs of your business.

Personal free account

With a **free trial account**[5] you can get started using Azure right away and you won't be charged until you choose to upgrade.

✓ Which subscription model are you most interested in?

For more information, you can see:

Solution providers - **https://www.microsoft.com/en-us/solution-providers/home**

Video - Subscription Access

This is an older video that is still valuable, but the Account Center (3:20) is no longer in use.

[1] https://azure.microsoft.com/en-us/pricing/enterprise-agreement/
[2] https://www.microsoft.com/en-us/licensing/licensing-programs/open-license.aspx
[3] https://azure.microsoft.com/en-us/offers/ms-azr-0111p/
[4] https://azure.microsoft.com/en-us/partners/directory/
[5] https://azure.microsoft.com/en-us/free/

Visual Studio Administrative Portal

Video: Visual Studio Administrative Portal

In this video you will get a walkthrough of the features and capabilities in the new Visual Studio Administration Portal. You'll learn how to navigate the new portal, add, edit, remove, bulk add and bulk edit subscribers. If your company uses Azure Active Directory or AAD, you'll find out how this makes it even easier to manage subscriptions.

Video: Personal Free Account

Personal Free Account

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.

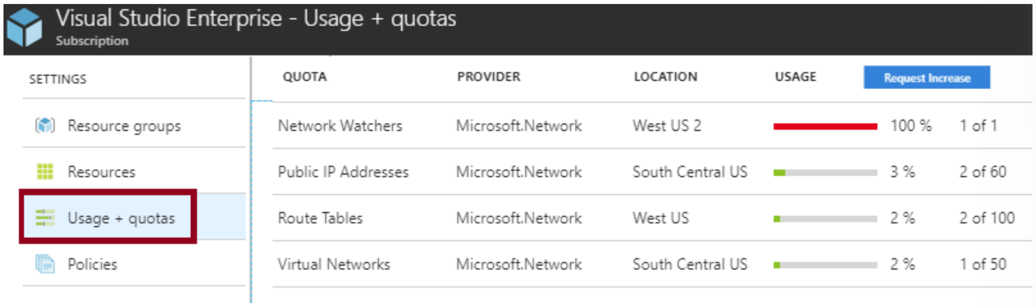
When you sign up, you'll also get 12-months of free compute, storage, network, and database services, and over 30 services that are continuously free, to learn and build your next ideas into prototypes.

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

Video: EA and Dev Test Subscriptions

Check Resource Limits

Azure provides the ability to see the number 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 to track current usage, and plan for future use. In this example, there are two Public IP Addresses in South Central US and the limit is 60.



QUOTA	PROVIDER	LOCATION	USAGE	
Network Watchers	Microsoft.Network	West US 2	100 %	1 of 1
Public IP Addresses	Microsoft.Network	South Central US	3 %	2 of 60
Route Tables	Microsoft.Network	West US	2 %	2 of 100
Virtual Networks	Microsoft.Network	South Central US	2 %	1 of 50

The limits shown are the limits for your subscription. If you need to increase a default limit, there is a Request Increase link. You will complete and submit the support request. All resources have a maximum limit listed in Azure **limits**[6]. If your current limit is already at the maximum number, the limit can't be increased.

□ You can also check your resource limits with PowerShell and the CLI. Learn more at the reference link.

For more information, you can see:

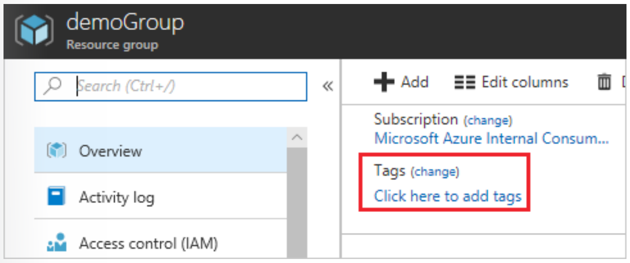
Check resource usage against limits - **https://docs.microsoft.com/en-us/azure/networking/check-usage-against-limits**

Resource Tags

You can apply tags to your Azure resources to logically organize them by categories. Each tag consists of a name and a value. For example, you can apply the name “Environment” and the value "Production" or “Development” to your resources. After creating your tags, you associate them with the appropriate resources.

With tags in place, you can retrieve all the resources in your subscription with that tag name and value. This means, you can retrieve related resources from different resource groups.

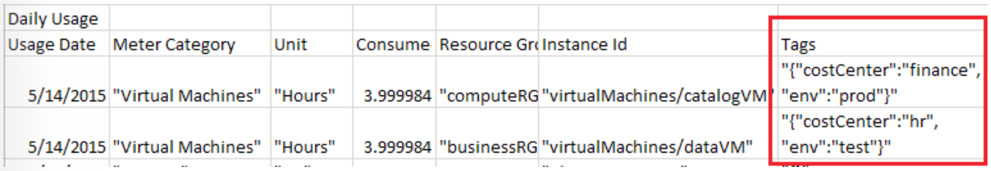
[6] https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits?toc=%2fazure%2fnetworking%2ftoc.json



Subscription (change)
Microsoft Azure Internal Consum...

Tags (change)
Click here to add tags

Perhaps one of the best uses of tags is to group billing data. When you download the usage CSV for services, the tags appear in the Tags column. For example, you could group virtual machines by cost center and production environment.



Daily Usage						
Usage Date	Meter Category	Unit	Consume	Resource Gr	Instance Id	Tags
5/14/2015	"Virtual Machines"	"Hours"	3.999984	"computeRG	"virtualMachines/catalogVM	"{"costCenter":"finance", "env":"prod"}"
5/14/2015	"Virtual Machines"	"Hours"	3.999984	"businessRG	"virtualMachines/dataVM"	"{"costCenter":"hr", "env":"test"}"

There are a few things to consider about tagging (more at the reference link):

- Each resource or resource group can have a maximum of 15 tag name/value pairs.
- Tags applied to the resource group are not inherited by the resources in that resource group.

✓ If you must create a lot of tags you will want to do that programmatically. You can use PowerShell or the CLI. Learn more at the reference links.

For more information, you can see:

Use tags to organize your Azure resources - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags**

PowerShell (Tagging) - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags#powershell**[7]

CLI (Tagging) - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags#azure-cli**[8]

Video: Enforcing Tags with Policy

[7] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags
[8] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags

Billing

Azure Accounts

An Azure account determines how Azure usage is reported and who the Account Administrator is. Accounts and subscriptions are created in the Azure Account Center. The person who creates the account is the Account Administrator for all subscriptions created in that account. That person is also the default Service Administrator for the subscription.

Subscription User Types

There are three roles related to Azure accounts and subscriptions:

Administrative role	Limit	Summary
Account Administrator	1 per Azure account	Authorized to access the Account Center (create subscriptions, cancel subscriptions, change billing for a subscription, change Service Administrator). This role has full control over the subscription and is the account that is responsible for billing.
Service Administrator	1 per Azure subscription	Authorized to access Azure Management Portal for all subscriptions in the account. By default, same as the Account Administrator when a subscription is created. This role has control over all the services in the subscription.
Co-administrator	200 per subscription (in addition to Service Administrator)	Same as Service Administrator but can't change the association of subscriptions to Azure directories.

Account administrator

The Account Administrator for a subscription is the only person with access to the Account Center. The Account Administrator does not have any other access to services in that subscription; they need to also be the Service Administrator or a co-administrator for that. For security reasons, the Account Administrator for a subscription can only be changed with a call to Azure support. The Account Administrator can easily reassign the Service Administrator for a subscription at the Account Center at any time.

Service administrator and co-administrator

The Service Administrator is the first co-administrator for a subscription. Like other co-administrators, the Service Administrator has management access to cloud resources using the Azure Management Portal, as well as tools like Visual Studio, other SDKs, and command line tools like PowerShell. The Service Administrator can also add and remove other co-administrators.

Additionally, Co-administrators can't delete the Service Administrator from the Azure Management Portal. Only the Account Administrator can change this assignment at the Account Center. The Service Administrator is the only user authorized to change a subscription's association with a directory in the Azure Management Portal.

✓ Account Administrators using a Microsoft account must log in every 2 years (or more frequently) to keep the account active. Inactive accounts are cancelled, and the related subscriptions removed. There are no login requirements if using a work or school account. Take a few minutes to look through the list of available roles at the reference link.

For more information, you can see:

Assigning administrator roles in Azure Active Directory - **https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles**

Service Usage

In the move from on-premises computing to cloud-hosted services, tracking and estimating service usage and related costs are significant concerns. It's important to be able to estimate what new resources will cost to run monthly and be able to project how the billing will look for a given month based on the current spending.

Azure provides a wealth of tools to help you predict and manage monthly billing. Some of these tools are described in the three sections below.

Get resource usage data

Azure provides a set of Billing REST APIs that give access to resource consumption and metadata information for Azure subscriptions. This gives you the ability to better predict and manage Azure costs. These Billing APIs enable you to track and analyze spending in hourly increments, create spending alerts, and predict future billing based on current usage trends.

Predict future costs

Although it's challenging to estimate costs ahead of time, Azure has a pricing calculator that you can use when you estimate the cost of deployed resources. You can also use the Billing blade in the portal and the Billing REST APIs to estimate future costs, based on current consumption.

Set up billing alerts

After you've deployed your application or solution on Azure, you can create alerts that send you email when you approach the spending limits that are defined in the alert.

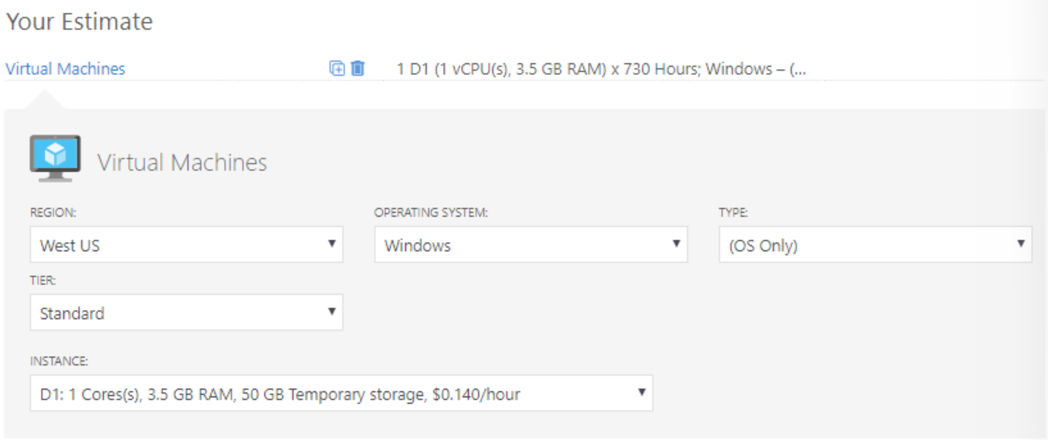
For more information, you can see:

Azure Cost Management Documentation - **https://docs.microsoft.com/en-us/azure/cost-management/**

Pricing Calculator

When you sign up for Azure, there are several things you can do to get a better idea of your spend. The pricing calculator can provide an estimate of costs before you create an Azure resource.

The Pricing Calculator provides estimates in all areas of Azure including compute, networking, storage, web, and databases.



Your Estimate

Virtual Machines 1 D1 (1 vCPU(s), 3.5 GB RAM) x 730 Hours; Windows – (...

Virtual Machines

REGION: West US

OPERATING SYSTEM: Windows

TYPE: (OS Only)

TIER: Standard

INSTANCE: D1: 1 Cores(s), 3.5 GB RAM, 50 GB Temporary storage, $0.140/hour

Prices are estimates and are not intended as actual price quotes. Actual prices may vary depending upon the date of purchase, currency of payment, and type of agreement you enter with Microsoft.

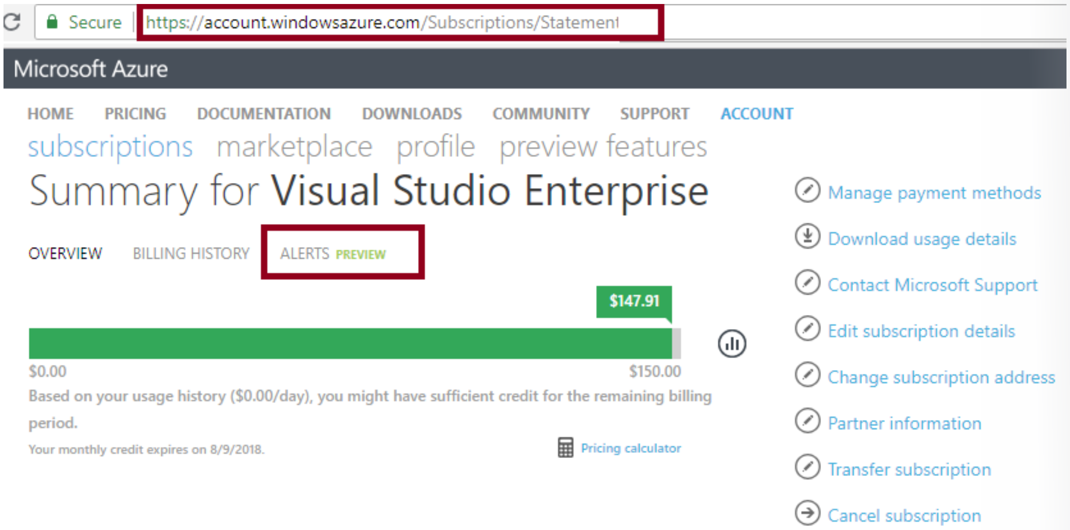
✓ Take a few minutes to access the Pricing Calculator and try a few scenarios.

For more information, you can see:

Pricing Calculator - **https://azure.microsoft.com/en-us/pricing/calculator/**

Billing Alert Service

If you’re the Account Admin for an Azure subscription, you can use the Azure Billing Alert Service to create customized billing alerts that help you monitor and manage billing activity for your Azure accounts. Billing alerts is available from the Account portal.



Secure https://account.windowsazure.com/Subscriptions/Statement

Microsoft Azure

HOME PRICING DOCUMENTATION DOWNLOADS COMMUNITY SUPPORT ACCOUNT

subscriptions marketplace profile preview features

Summary for Visual Studio Enterprise

OVERVIEW BILLING HISTORY ALERTS PREVIEW

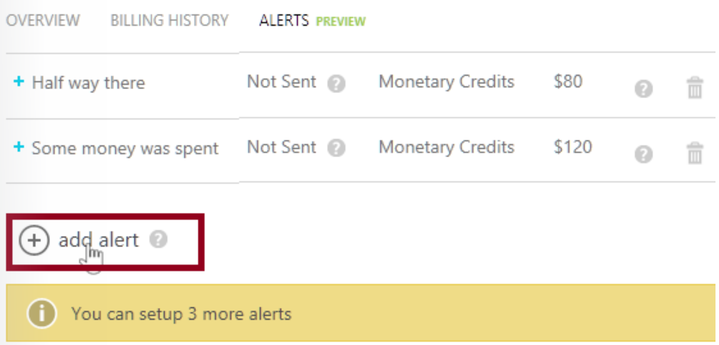
$147.91

$0.00 $150.00

Based on your usage history ($0.00/day), you might have sufficient credit for the remaining billing period.

Your monthly credit expires on 8/9/2018.

You can set up a total of five billing alerts per subscription, with a different threshold and up to two email recipients for each alert.



+ Half way there	Not Sent	Monetary Credits	$80
+ Some money was spent	Not Sent	Monetary Credits	$120

You can setup 3 more alerts

✓ EA subscriptions are not supported by this service, instead EA customers can get alerts for each department under an enrollment by setting spending quotas.

For more information, you can see:

Set up billing or credit alerts for your Microsoft Azure subscriptions - **https://docs.microsoft.com/en-us/azure/billing/billing-set-up-alerts**

Video: Understand Your Bill

Additional Practice - View Your Bill

Take a few minutes and try the **Download or view your Azure billing invoice and daily usage data**[9] steps. In this practice, you learn how to:

- Get your invoice in email.
- Download an invoice from the Azure portal.
- Download usage from the Account Center.

✓ An invoice is only generated when you owe money. If you have a monthly credit amount or if you have a Free Trial then you may not have an invoice.

For more information, you can see:

Understand terms on your Microsoft Azure detailed usage charges - **https://docs.microsoft.com/en-us/azure/billing/billing-understand-your-usage**

Understand terms on your Microsoft Azure invoice - **https://docs.microsoft.com/en-us/azure/billing/billing-understand-your-invoice**

[9] https://docs.microsoft.com/en-us/azure/billing/billing-download-azure-invoice-daily-usage-date

Video: Azure Billing

Azure Policy

Video: Azure Policy

Azure Policy

Azure Policy is a service in Azure that you use to create, assign and, manage policies. These policies enforce different rules over your resources, so those resources stay compliant with your corporate standards and service level agreements. Azure Policy does this by running evaluations of your resources and scanning for those not compliant with the policies you have created.

The main advantages of Azure policy are in the areas of enforcement and compliance, scaling, and remediation.

- **Enforcement and compliance**. Turn on built-in policies or build custom ones for all resource types. Real time policy evaluation and enforcement. Periodic and on-demand compliance evaluation.
- **Apply policies at scale**. Apply policies to a Management Group with control across your entire organization. Apply multiple policies and aggregate policy states with policy initiative. Define an exclusion scope.
- **Remediation**. Real time remediation, and remediation on existing resources (coming soon).

Azure Policy will be important to you if your team runs an environment where you need to govern:

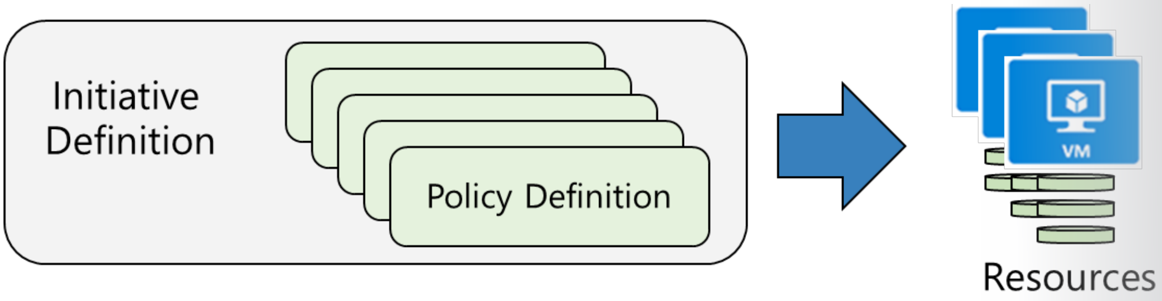
- Multiple engineering teams (deploying to and operating in the environment)
- Multiple subscriptions
- Need to standardize/enforce how cloud resources are configured
- Manage regulatory compliance, cost control, security, or design consistency

For more information, you can see:

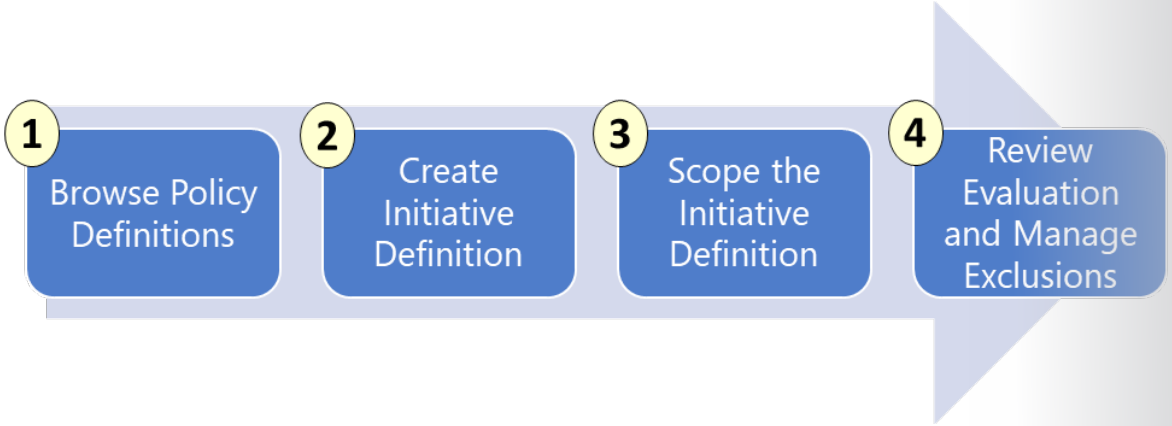
Azure Policy Documentation - **https://docs.microsoft.com/azure/azure-policy/** [10]

[10] https://docs.microsoft.com/azure/azure-policy/

Implementing Azure Policy



To implement Azure Policies, you can follow these steps.


1
Browse Policy Definitions
2
Create Initiative Definition
3
Scope the Initiative Definition
4
Review Evaluation and Manage Exclusions

1. **Browse Policy Definitions**. A Policy Definition expresses what to evaluate and what actions to take. Every policy definition has conditions under which it is enforced. And, it has an accompanying effect that takes place if the conditions are met. For example, you could prevent VMs from being deployed if they are exposed to a public IP address.
2. **Create Initiative Definitions**. An initiative definition is a set of Policy Definitions to help track your compliance state for a larger goal. For example, ensuring a branch office is compliant.
3. **Scope the Initiative Definition**. You can limit the scope of the Initiative Definition to Management Groups, Subscriptions, or Resource Groups.
4. **View Policy Evaluation results**. Once an Initiative Definition is assigned, you can evaluate the state of compliance for all your resources. Individual resources, resource groups, and subscriptions within a scope can be exempted from the having policy rules affect it. Exclusions are handled individually for each assignment.

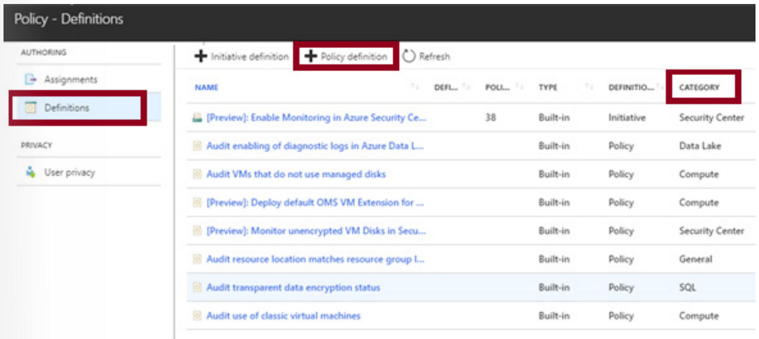
✓ Even if you have only a few Policy Definitions, we recommend creating an Initiative Definition.

Browse Policy Definitions

There are many Built-in Policy Definitions for you to choose from. Sorting by Category will help you locate what you need. For example,

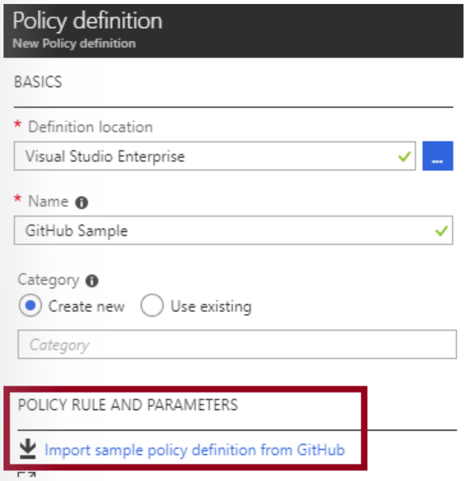
- The Allowed Virtual Machine SKUs enables you to specify a set of virtual machine SKUs that your organization can deploy.

- The Allowed Locations policy enables you to restrict the locations that your organization can specify when deploying resources. This can be used to enforce your geo-compliance requirements.



NAME	DEFI...	POLI...	TYPE	DEFINITIO...	CATEGORY
[Preview]: Enable Monitoring in Azure Security Ce...		38	Built-in	Initiative	Security Center
Audit enabling of diagnostic logs in Azure Data L...			Built-in	Policy	Data Lake
Audit VMs that do not use managed disks			Built-in	Policy	Compute
[Preview]: Deploy default OMS VM Extension for ...			Built-in	Policy	Compute
[Preview]: Monitor unencrypted VM Disks in Secu...			Built-in	Policy	Security Center
Audit resource location matches resource group l...			Built-in	Policy	General
Audit transparent data encryption status			Built-in	Policy	SQL
Audit use of classic virtual machines			Built-in	Policy	Compute

If you don't see what you need you can add a Policy Definition. The easiest way to do this is to Import a policy from **GitHub**[11]. New Policy Definitions are added almost every day.


Policy definition
New Policy definition
BASICS
POLICY RULE AND PARAMETERS
Import sample policy definition from GitHub

* Definition location
Visual Studio Enterprise

* Name
GitHub Sample

Category
● Create new ○ Use existing
Category

✓ Policy Definitions have a **specific JSON format**[12]. As a Azure Administrator you will not need to create files in this format, but you may want to take a look just, so you are familiar.

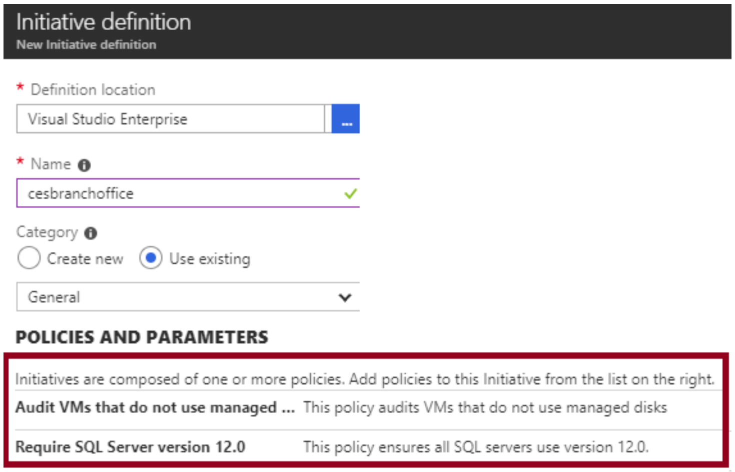
For more information, you can see:

Policy Definition - **https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction#policy-definition**[13]

Create Initiative Definitions

Once you have determined which Policy Definitions you need, you create an Initiative Definition. This definition will include one or more policies. There is a pick list on the right side of the New Initiative definition page (not shown) to make your selection.

[11] https://github.com/Azure/azure-policy/tree/master/samples
[12] https://docs.microsoft.com/en-us/azure/azure-policy/policy-definition
[13] https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction



Initiative definition
New Initiative definition

* Definition location
Visual Studio Enterprise

* Name
cesbranchoffice

Category
○ Create new ● Use existing

General

POLICIES AND PARAMETERS

Initiatives are composed of one or more policies. Add policies to this Initiative from the list on the right.

Audit VMs that do not use managed ...	This policy audits VMs that do not use managed disks
Require SQL Server version 12.0	This policy ensures all SQL servers use version 12.0.

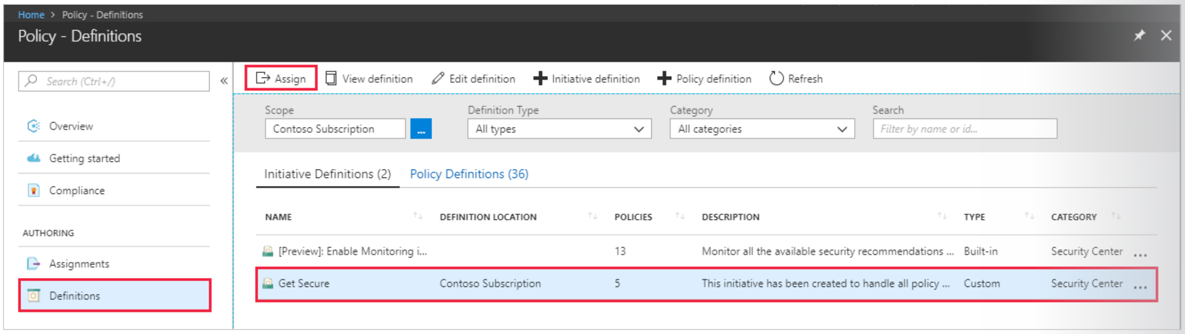
✓ Can you see how this will require some planning to organize your policies?

For more information, you can see:

Initiative definition - **https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction#initiative-definition**[14]

Scope the Initiative Definition

Once our Initiative Definition is created, you can assign the definition to establish its scope. A scope determines what resources or grouping of resources the policy assignment gets enforced on.

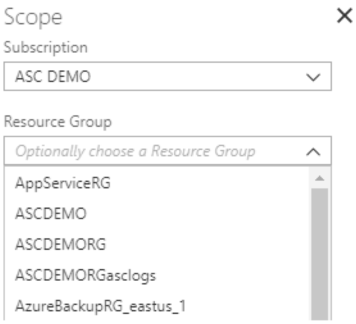


Scope: Contoso Subscription
Definition Type: All types
Category: All categories
Search: Filter by name or id...

Initiative Definitions (2) | Policy Definitions (36)

NAME	DEFINITION LOCATION	POLICIES	DESCRIPTION	TYPE	CATEGORY
[Preview]: Enable Monitoring i...		13	Monitor all the available security recommendations ...	Built-in	Security Center
Get Secure	Contoso Subscription	5	This initiative has been created to handle all policy ...	Custom	Security Center

You can select the Subscription, and then optionally a Resource Group.



Scope

Subscription
ASC DEMO

Resource Group
Optionally choose a Resource Group

AppServiceRG
ASCDEMO
ASCDEMORG
ASCDEMORGasclogs
AzureBackupRG_eastus_1

[14] https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction

✓ Currently, an Initiative Definition can have up to 100 policies.

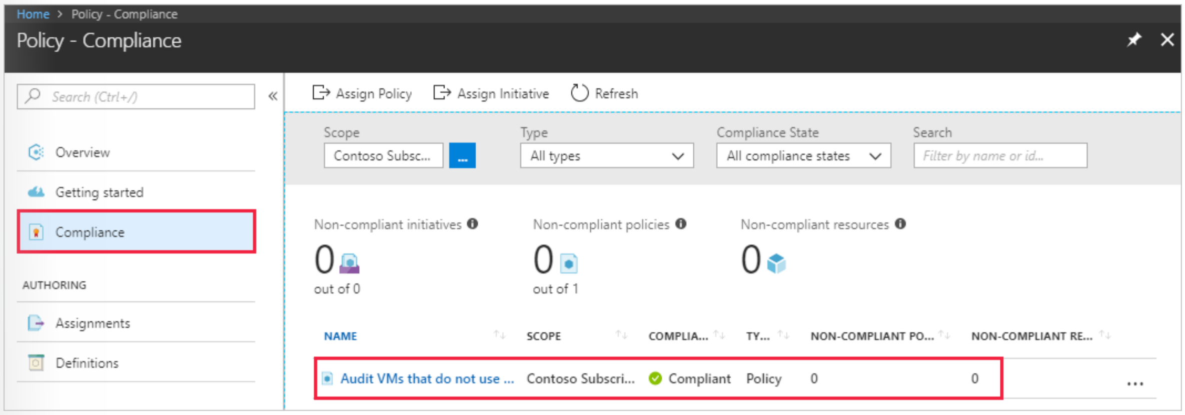
For more information, you can see:

Initiative assignment - **https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction#initiative-assignment**[15]

Recommendations for managing policies - **https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction#recommendations-for-managing-policies**[16]

Determine Compliance

Once your policy is in place you can use the Compliance blade to review non-compliant initiatives, non-compliant policies, and non-compliant resources.



Scope: Contoso Subsc...
Type: All types
Compliance State: All compliance states

Non-compliant initiatives: 0 out of 0
Non-compliant policies: 0 out of 1
Non-compliant resources: 0

NAME	SCOPE	COMPLIA...	TY...	NON-COMPLIANT PO...	NON-COMPLIANT RE...
Audit VMs that do not use ...	Contoso Subscri...	Compliant	Policy	0	0

When a condition is evaluated against your existing resources and found true, then those resources are marked as non-compliant with the policy. Although you don't see the evaluation logic in the Azure portal, the compliance state results are shown. The compliance state result is either compliant or non-compliant.

✓ Policy evaluation happens about once an hour, which means that if you make changes to your policy definition and create a policy assignment then it will be re-evaluated over your resources within the hour.

For more information, you can see:

Identify non-compliant resources - **https://docs.microsoft.com/en-us/azure/azure-policy/assign-policy-definition#identify-non-compliant-resources**

Additional Practice - Create and Manage Policies

Take a few minutes to try the **Tutorial: Create and manage policies to enforce compliance**[17]. In this tutorial, you learn how to:

- Assign a policy to enforce a condition for resources you create in the future.
- Create and assign an initiative definition to track compliance for multiple resources.
- Resolve a non-compliant or denied resource.
- Implement a new policy across an organization.

✓ You can also create a policy with PowerShell or the CLI.

[15] https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction
[16] https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction
[17] https://docs.microsoft.com/en-us/azure/azure-policy/create-manage-policy

For more information, you can see:

Quickstart: Create a policy assignment to identify non-compliant resources using the Azure RM PowerShell module - **https://docs.microsoft.com/en-us/azure/azure-policy/assign-policy-definition-ps**

Create a policy assignment to identify non-compliant resources in your Azure environment with the Azure CLI - **https://docs.microsoft.com/en-us/azure/azure-policy/assign-policy-definition-cli**

Module 1 Review Questions

Module 1 Review Questions

Management Groups

You manage several Azure subscriptions for an organization. You need to be able to efficiently manage the subscriptions. What are management groups? What advantages does management groups provide? How could you use management groups in this situation?

Click for suggested answer ↓

If your organization has several subscriptions, you may need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups give a level of scope above 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.
- Targeting of policies and spend budgets across subscriptions and inheritance down the hierarchies.
- Compliance and cost reporting by organization (business/teams).

Tagging

You manage Azure resources for an organization. Many distinct groups within the organization use Azure resources. You need to organize the resources based on which group is using the resources. How can you use tagging to help organize your resources? What are the benefits and limitations of tagging?

Click for suggested answer ↓

You can apply tags to your Azure resources to logically organize them by categories. Each tag consists of a name and a value. For example, you can apply the name “Environment” and the value “Production” or “Development” to your resources. After creating your tags, you associate them with the proper resources.

Benefits. With tags in place, you can retrieve all the resources in your subscription with that tag name and value. This means, you can retrieve related resources from different resource groups. One of the best uses of tags is to group billing data. When you download the usage CSV for services, the tags appear in the Tags column. For example, you could group virtual machines by cost center and production environment.

Limitations. Each resource or resource group can have a maximum of 15 tag name/value pairs. Tags applied to the resource group are not inherited by the resources in that resource group.

Azure Policies

You are managing Azure resources for an organization. You need to ensure that all resources follow corporate standards and service level agreements (SLA). You have decided to use Azure policies. What are Azure policies and what are the main advantages to using them?

Click for suggested answer ↓

Azure Policy is a service in Azure that you use to create, assign and, manage policies. These policies enforce different rules over your resources, so those resources stay compliant with your corporate standards and service level agreements.

The main advantages of Azure policy are in the areas of enforcement and compliance, scaling, and remediation.

- **Enforcement and compliance**. Turn on built-in policies or build custom ones for all resource types. Real time policy evaluation and enforcement. Periodic and on-demand compliance evaluation.
- **Apply policies at scale**. Apply policies to a Management Group with control across your entire organization. Apply multiple policies and aggregate policy states with policy initiative. Define an exclusion scope.
- **Remediation**. Real time remediation, and remediation on existing resources (coming soon).

Module 2 Access Management for Cloud Resource

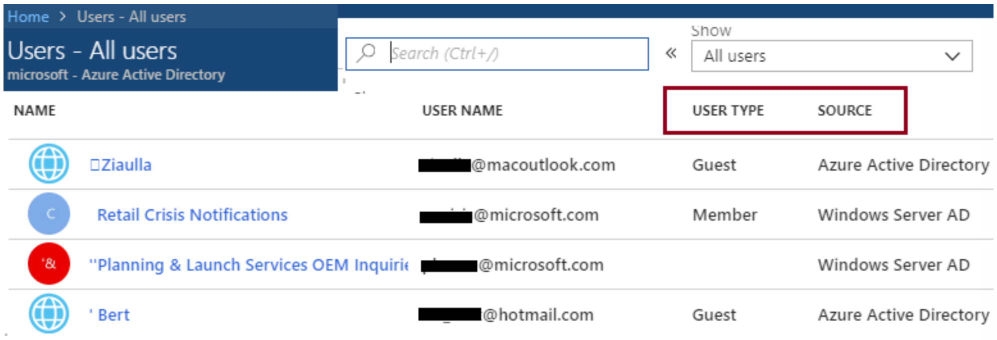
Azure Users and Groups

Video: Managing Users and Groups

User Accounts

In Azure AD, all users who require access to resources must have a user account. A user account is an Azure AD user object that contains all the information that's required to authenticate and authorize the user during the sign-in process and build the user's access token.

To view the Azure AD users, simply access the All users blade.



Show: All users

NAME	USER NAME	USER TYPE	SOURCE
Ziaulla	[redacted]@macoutlook.com	Guest	Azure Active Directory
Retail Crisis Notifications	[redacted]@microsoft.com	Member	Windows Server AD
"Planning & Launch Services OEM Inquirie	[redacted]@microsoft.com		Windows Server AD
' Bert	[redacted]@hotmail.com	Guest	Azure Active Directory

Notice the Source in the above screenshot. There are different sources depending on the types of identity, including:

- **Cloud identities (Azure Active Directory)**. Users that only exist in Azure AD. For example, administrator accounts or users you are managing yourself.
- **Directory-synchronized identities (Windows Server AD)**. Users brought in to Azure through a synchronization activity using Azure AD Connect. These are users that exist in Windows Server AD.
- **Guest users (Azure Active Directory)**. Users from outside Azure. For example, Google and Microsoft accounts.

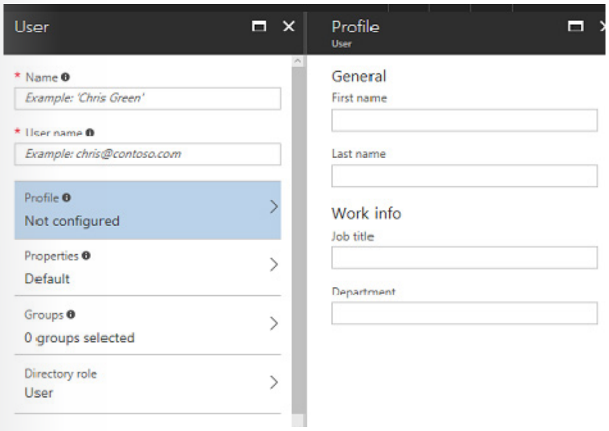
✓ Take a minute to access the Portal and view your users. Notice the User Type and Source columns. Have you given any thought as to the type of users you will need?

Adding User Accounts

There are multiple ways to add cloud identities to Azure AD.

Azure Portal

You can add new users through the Azure Portal. In addition to Name and User name, there is profile information like Job Title and Department.



User

* Name: *Example: 'Chris Green'*

* User name: *Example: chris@contoso.com*

Profile: Not configured

Properties: Default

Groups: 0 groups selected

Directory role: User

Profile
User

General

First name

Last name

Work info

Job title

Department

Azure PowerShell

```
    You can use the PowerShell New-AzureADUser command to add cloud-based
users.
    # Create a password object
    $PasswordProfile = New-Object -TypeName Microsoft.Open.AzureAD.Model.
PasswordProfile
    # Assign the password
    $PasswordProfile.Password = "<Password>"
    # Create the new user
    New-AzureADUser -AccountEnabled $True -DisplayName "Abby Brown" -Pass-
wordProfile $PasswordProfile -MailNickName "AbbyB" -UserPrincipalName <a
href="mailto:AbbyB@contoso.com" title="" target="_blank" data-generat-
ed=''>AbbyB@contoso.com</a>
```

✓ Users can also be added to Azure AD through Office 365 Admin Center, Microsoft Intune admin console, and the CLI. Which of the options mentioned in this topic do you prefer?

For more information, you can see:

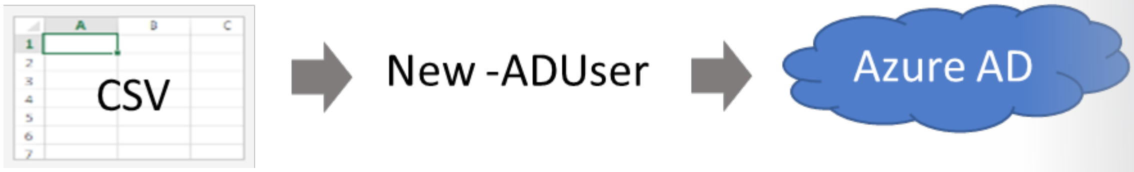
Add or change profile information for a user in Azure Active Directory - **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-users-profile-azure-portal**

Creating a new user in Azure AD - **https://docs.microsoft.com/en-us/powershell/azure/active-directory/new-user-sample?view=azureadps-2.0**

az ad user create - **https://docs.microsoft.com/en-us/cli/azure/ad/user?view=azure-cli-latest#az_ad_user_create**[1]

Bulk User Accounts

There are several ways you can use PowerShell to import data into your directory, but the most commonly used method is to use a CSV file. This file can either be manually created, for example using Excel, or it can be exported from an existing data source such as a SQL database or an HR application.



If you are going to use a CSV file here are some things to think about:

- **Naming conventions**. Establish or implement a naming convention for usernames, display names and aliases. For example, a user name could consist of last name, period, first name: Smith.John@contoso.com.
- **Passwords**. Implement a convention for the initial password of the newly created user. Figure out a way for the new users to receive their password in a secure way. Methods commonly used for this are generating a random password and emailing it to the new user or their manager.

The steps for using the CSV file are very straightforward. Use the reference link to see a sample PowerShell script.

1. Use **Connect-AzureAD** to create a PowerShell connection to your directory You should connect with an admin account that has privileges on your directory.
2. Create a new Password Profile for the new users. The password for the new users needs to conform to the password complexity rules you have set for your directory.
3. Use **Import-CSV** to import the csv file. You will need to specify the path and file name of the CSV file.
4. Loop through the users in the file constructing the user parameters required for each user. For example, 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, you can see:

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

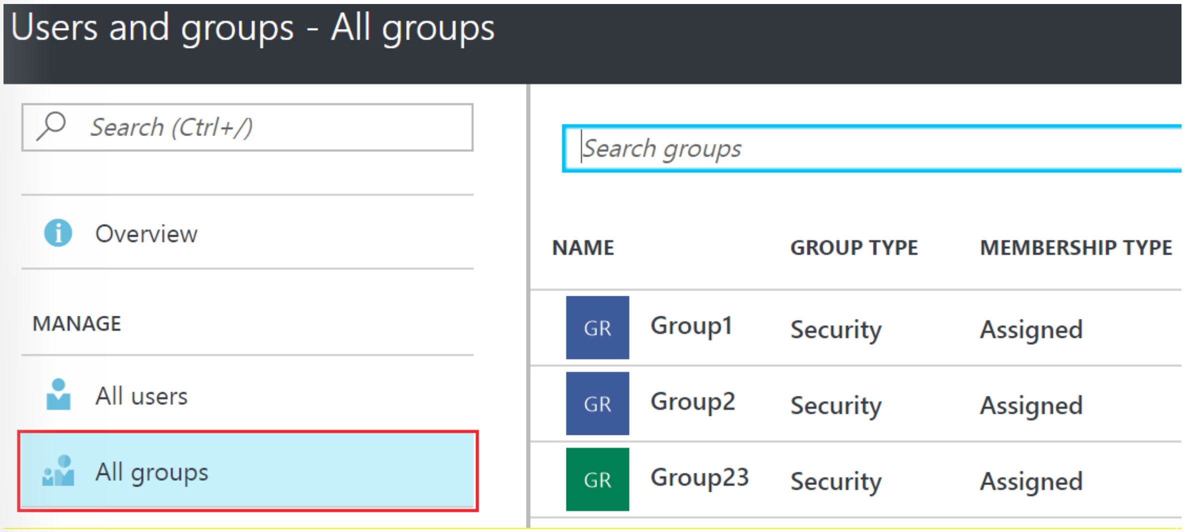
New-ADUser - **https://docs.microsoft.com/en-us/powershell/module/azuread/new-azureaduser?view=azureadps-2.0**

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

Group Accounts

A group helps organize users to make it easier to manage permissions. Groups can be easily added through the portal. There are two types of groups: security groups and distribution groups.

- **Security groups** are security-enabled and are used to assign permissions and control access to various resources.
- **Distribution groups** are used mainly by email applications and are not security enabled. You can easily add groups in the portal.



NAME	GROUP TYPE	MEMBERSHIP TYPE
GR Group1	Security	Assigned
GR Group2	Security	Assigned
GR Group23	Security	Assigned

Adding Groups

You can also use PowerShell to add a group with the New-AzureADGroup command.

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

Adding Members to Groups

There are two ways to add members to Azure groups.

- **Directly Assigned**. In this situation you create the group then you manually add individual user accounts to the group.
- **Dynamically Assigned**. In this situation you create rules to enable attribute-based dynamic memberships for groups based on characteristics. For example, if a user's Department is Sales, then they are dynamically assigned to the Sales group. You can set up a rule for dynamic membership on security groups or Office 365 groups. This feature requires an Azure AD Premium P1 license.

✓ Have you given any thought to which groups you need to create? Would you directly assign or dynamically assign membership?

For more information, you can see:

Manage group membership for users in your Azure Active Directory tenant - **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-members-azure-portal**

Create attribute-based rules for dynamic group membership in Azure Active Directory - **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-dynamic-membership-azure-portal**

Create a group and add members in Azure Active Directory - **https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-create-azure-portal**

New-AzureADGroup - **https://docs.microsoft.com/en-us/powershell/module/azuread/new-azuread-group?view=azureadps-2.0**

Demonstration: Create User and Group Accounts

Additional Practice - Users and Groups

✓ For the Quickstarts in this practice, you will to sign in to Azure with an account that's a global admin for the directory.

Add New Users to Active Directory

Try the **Quickstart: Add new users to Azure Active Directory**[2]. This Quickstart explains how to delete or add users in your organization into your organization's Azure Active Directory (Azure AD) tenant using the Azure portal or by synchronizing your on-premises Windows Server AD user account data.

Manage Group Membership

Try the **Manage group membership for users in your Azure Active Directory tenant**[3]. This article explains how to manage the members for a group in Azure Active Directory (Azure AD).

Create a group and add members

Try the **Create a group and add members in Azure Active Directory**[4]. This article explains how to create and populate a new group in Azure Active Directory. Use a group to perform management tasks such as assigning licenses or permissions to several users or devices at once.

Manage profile information

Try the **Add or change profile information for a user in Azure Active Directory**[5] article. This article explains how to add user profile information, such as a profile picture or phone and email authentication information, in Azure Active Directory (Azure AD).

✓ As you have time, experiment with other user and group administrative tasks. Also, if you want to try some of these tasks using PowerShell, see the Azure Active Directory PowerShell 2.0 cmdlet reference for **AzureAD**[6].

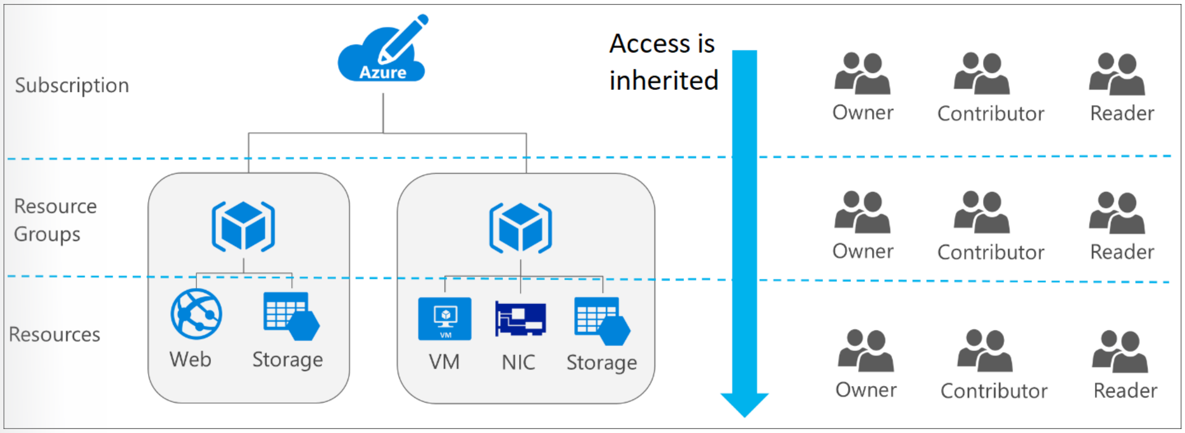
[2] https://docs.microsoft.com/en-us/azure/active-directory/add-users-azure-active-directory
[3] https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-members-azure-portal
[4] https://docs.microsoft.com/en-us/azure/active-directory/active-directory-groups-create-azure-portal
[5] https://docs.microsoft.com/en-us/azure/active-directory/active-directory-users-profile-azure-portal
[6] https://docs.microsoft.com/en-us/powershell/module/Azuread/?view=azureadps-2.0

Role-based Access Control

Video: Role-based Access Control

RBAC Concepts

Managing access to resources in Azure is a critical part of an organization's security and compliance requirements. Role-based access control (RBAC) is the capability for you to grant appropriate access to Azure AD users, groups, and services. RBAC is configured by selecting a role (the definition of what actions are allowed and/or denied), then associating the role with a user, group or service principal. Finally, this combination of role and user/group/service principal is scoped to either the entire subscription, a resource group, or specific resources within a resource group.



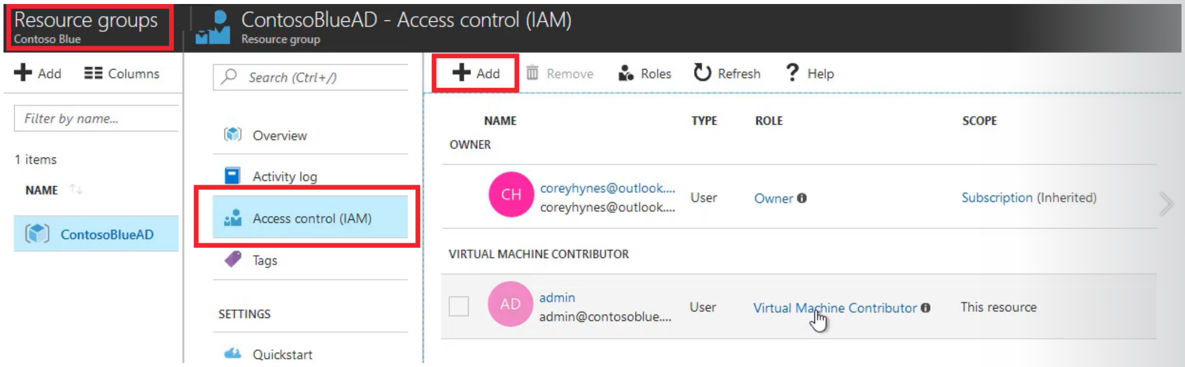
Roles

A role is a collection of actions that can be performed on Azure resources. A user or a service can perform an action on an Azure resource if they have been assigned a role that contains that action. There are many built-in roles. Three of the most common roles are Owner, Contributor and Reader.

Role name	Description
Owner	Owner can manage everything, including access.
Contributor	Contributors can manage everything except access.
Reader	Readers can view everything but can't make changes.

Using the Portal to implement RBAC

You can use the Azure Portal to make your role assignments. In this example, the ContosoBlueAD resource group shows on the Access Control (IAM) blade the current roles and scopes. You can add or remove roles as you need. You can add synced users and groups to Azure roles, which enables organizations to centralize the granting of access.



NAME	TYPE	ROLE	SCOPE
OWNER			
CH coreyhynes@outlook.... coreyhynes@outlook....	User	Owner	Subscription (Inherited)
VIRTUAL MACHINE CONTRIBUTOR			
AD admin admin@contosoblue....	User	Virtual Machine Contributor	This resource

✓ Users and groups are sourced from Azure Active Directory, which is commonly populated with credentials from on-premises directories, such as Active Directory. Note that RBAC access that you grant at parent scopes is inherited at child scopes.

For more information, see:

Get started with access management in the Azure portal: **https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-control-what-is**

Administrator Permissions

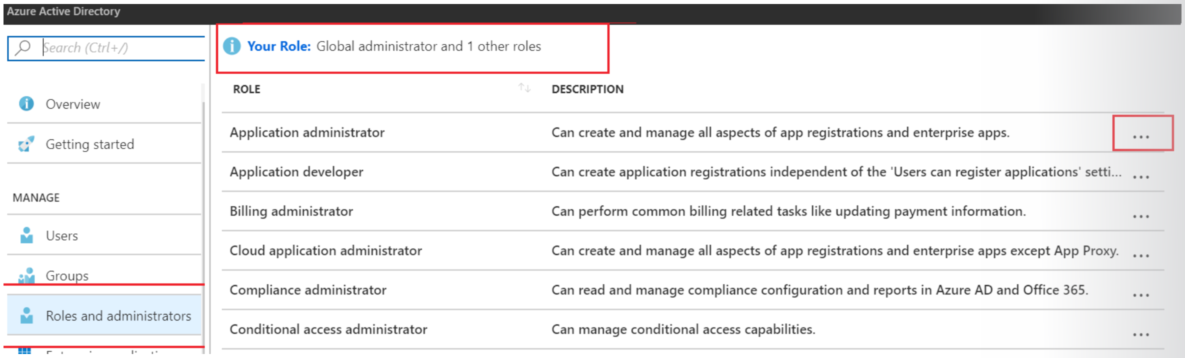
Using Azure AD, you can designate separate administrators to serve different functions. Administrators can be designated in the Azure AD portal to perform tasks such as adding or changing users, assigning administrative roles, resetting user passwords, managing user licenses, and managing domain names.

Global administrator

The global administrator has access to all administrative features. By default, the person who signs up for an Azure subscription is assigned the global administrator role for the directory. Only global administrators can assign other administrator roles.

Viewing role membership

You can see and manage all the members of the administrator roles in the Azure Active Directory portal. When you're viewing a roles members, you can see the complete list of permissions granted by the role assignment. This includes links to relevant documentation to help guide you through managing directory roles.



Your Role: Global administrator and 1 other roles

ROLE	DESCRIPTION
Application administrator	Can create and manage all aspects of app registrations and enterprise apps.
Application developer	Can create application registrations independent of the 'Users can register applications' setti...
Billing administrator	Can perform common billing related tasks like updating payment information.
Cloud application administrator	Can create and manage all aspects of app registrations and enterprise apps except App Proxy.
Compliance administrator	Can read and manage compliance configuration and reports in Azure AD and Office 365.
Conditional access administrator	Can manage conditional access capabilities.

✓ Are you using the Azure forums to find information and post questions? If not, try the reference link.

For more information, you can see:

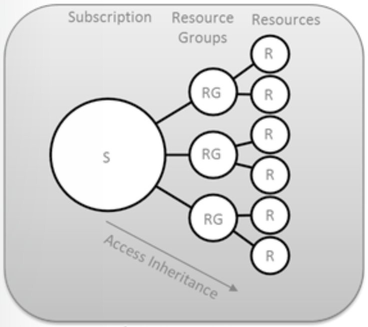
Assigning administrator roles in Azure Active Directory - **https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles**

Available roles - **https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles#available-roles**

Active Directory Forum - **https://feedback.azure.com/forums/169401-azure-active-directory?category_id=166032**

Resource Scope

Access does not need to be granted to the entire subscription. Roles can also be assigned for resource groups as well as for individual resources. In Azure RBAC, a resource inherits role assignments from its parent resources. So if a user, group, or service is granted access to only a resource group within a subscription, they will be able to access only that resource group and resources within it, and not the other resources groups within the subscription. As another example, a security group can be added to the Reader role for a resource group, but be added to the Contributor role for a database within that resource group.



Role Assignment Scopes

Role Assignment

A role assignment is created that associates a security principal to a role. The role is further used to grant access to a resource scope. This decoupling allows you to specify that a specific role has access to a resource in your subscription and add/remove security principals from that role in a loosely connected manner. Roles can be assigned to the following types of Azure AD security principals:

- **Users**. Roles can be assigned to organizational users that are in the Azure AD with which the Azure subscription is associated. Roles can also be assigned to external Microsoft accounts that exist in the same directory.
- **Groups**. Roles can be assigned to Azure AD security groups. A user is automatically granted access to a resource if the user becomes a member of a group that has access. The user also automatically loses access to the resource after getting removed from the group. A best practice is to manage access through groups by assigning roles to those groups and adding users – instead of assigning roles directly to users.
- **Service principals**. Service identities are represented as service principals in the directory. They authenticate with Azure AD and securely communicate with one another. Services can be granted access to Azure resources by assigning roles through the Azure module for Windows PowerShell to the Azure AD service principal representing that service.

Role Definitions

In a previous topic, you were introduced to three of the most common built-in roles: Owner, Contributor, and Reader. Azure AD provides many other **built-in roles**[7] to cover the most common security scenarios.

Role definitions

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

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

In this example the Owner role means all (*) actions, no denied actions, and all (/) scopes. This information is available with the Get-AzureRmRoleDefinition cmdlet.

Actions and NotActions

The Actions and NotActions properties can be tailored to grant and deny the exact permissions you need. Review this table to see how Owner, Contributor, and Reader are defined.

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

✓ Take a minute to open the Azure Portal, open the Subscriptions or Resource Group blade, and click Access Control (IAM). Click Add and take a few minutes to review the built-in roles and see which role you would be most interested in using.

For more information, you can see:

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

Create custom roles for Azure Role-Based Access Control - **https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-control-custom-roles**

Get-AzureRmRoleDefinition - **https://docs.microsoft.com/en-us/previous-versions/azure/mt603792(v=azure.100)**

Assignable Scopes

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

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

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 in only the subscriptions or resource groups that require it, and not clutter 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]/
[resource]
```

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

Makes a role available for assignment only in the Network resource group.

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

✓ Take a minute to open the Azure Portal and use the Access Control blade to add a role and then assign it to a user. For your organization which role assignments would you need?

For more information, you can see:

Custom roles access control - **https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-control-custom-roles#custom-roles-access-control**[8]

Demonstration: Role-based Access Control

Additional Practice - RBAC

Role-based access control (RBAC) is the way that you manage access to resources in Azure. In this Quickstart, you grant a user access to create and manage virtual machines in a resource group. Take a few minutes to work through the **Grant access for a user using RBAC and the Azure portal**[9]. This Quickstart steps through the basics of:

- Creating a resource group in the Azure portal.
- Assign a user to a role.

[8] https://docs.microsoft.com/en-us/azure/active-directory/role-based-access-control-custom-roles
[9] https://docs.microsoft.com/en-us/azure/role-based-access-control/quickstart-assign-role-user-portal

- Remove the created role assignment.

Using PowerShell

Next, try the following **tutorial**[10] to grant a user access to view all resources in a subscription and manage everything in a resource group using Azure PowerShell. In this tutorial you will:

- Create a user
- Create a resource group
- Use the Get-AzureRMRoleAssignment command to list the role assignments
- Use the Remove-AzureRmResourceGroup command to remove access

For more information, you can see:

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

Video: Enterprise Admins

[10] https://docs.microsoft.com/en-us/azure/role-based-access-control/tutorial-role-assignments-user-powershell

Module 2 Review Questions

Module 2 Review Questions

User Accounts

You are managing Azure AD for an organization. You engage a third-party consulting company to assist with technical issues. You need to give the consultant access to the Azure resources.What is an Azure user account? What are the three identity sources for user accounts? Which type of account would you use for the consultant?

Click for suggested answer ↓

In Azure AD, all users who need access to resources must have a user account. A user account is an Azure AD user object that has all the information that is needed to authenticate and authorize the user during the sign-in process and build the user's access token. There are various sources depending on the types of identity, including:

- Cloud identities (Azure Active Directory). Users that only exist in Azure AD. For example, administrator accounts or users you are managing yourself.
- Directory-synchronized identities (Windows Server AD). Users brought in to Azure through a synchronization activity using Azure AD Connect. These are users that exist in Windows Server AD.
- Guest users (Azure Active Directory). Users from outside Azure. For example, Google and Microsoft accounts.

Group Accounts

You need to manage access permissions for an Azure tenant. What are Azure groups? What are the two basic types of groups? What are the two ways to assign members to Azure groups?

Click for suggested answer ↓

A group helps organize users to make it easier to manage permissions. You can easily create and configure groups by using the Azure portal. There are two types of groups: security groups and distribution groups. Security groups are security-enabled and are used to assign permissions and control access to various resources. Distribution groups are used by email applications and are not security enabled. You can easily add groups in the portal.

There are two ways to add members to Azure groups.

- Directly Assigned. In this situation you create the group then you manually add individual user accounts to the group.
- Dynamically Assigned. In this situation you create rules to enable attribute-based dynamic memberships for groups based on characteristics. For example, a user in the sales department is dynamically assigned to a Sales group. You can set up a rule for dynamic membership on security groups or Office 365 groups. This feature needs an Azure AD Premium P1 license.

Role-Based Access Control

You are managing permissions for an Azure tenant. You need to define several levels of control for groups of users. The requirements for the groups are:

- Group 1: Manage everything, including access.
- Group 2: Manage everything except access.
- Group 3: View everything but do not allow changes.

What is RBAC? What are the three most common roles in Azure? Which role would you use for each group? How do you view the roles in the Azure portal?

Click for suggested answer ↓

Managing access to resources in Azure is a critical part of an organization’s security and compliance requirements. Role-based access control (RBAC) is the capability for you to grant proper access to Azure AD users, groups, and services.

Three of the most common roles are Owner, Contributor and Reader. Owner can manage everything, including access. Contributors can manage everything except access. Readers can view everything but can't make changes.

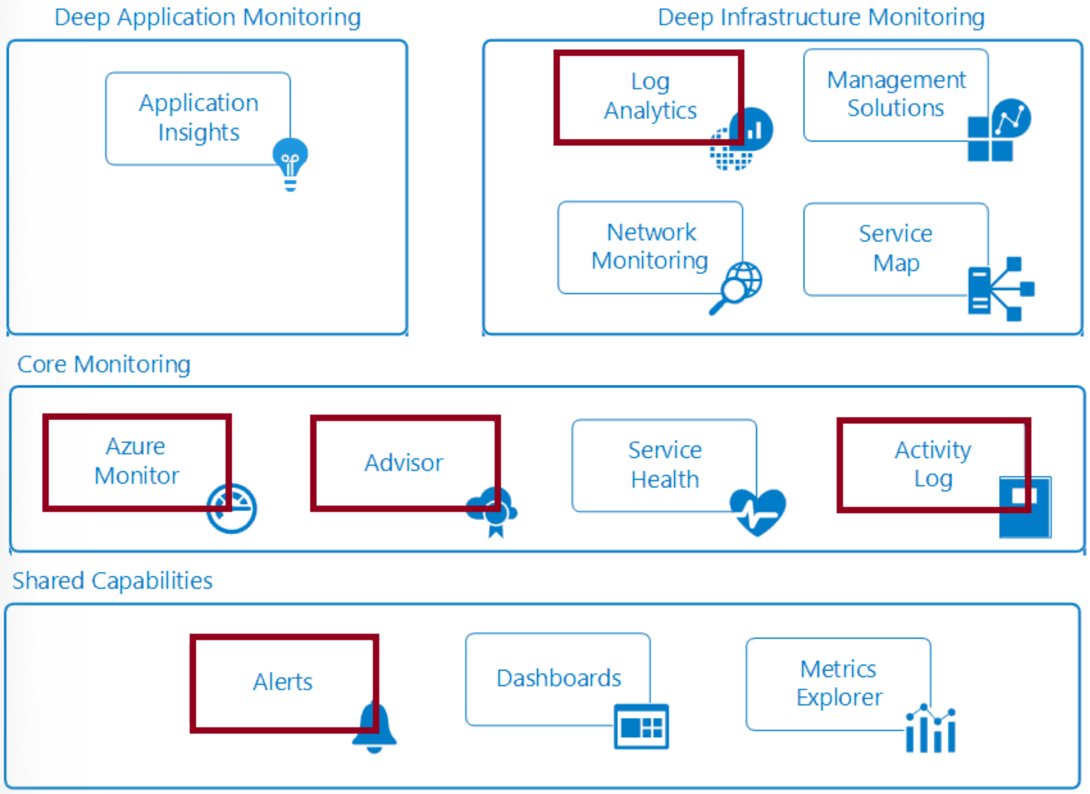
Module 3 Monitoring and Diagnostics

Exploring Monitoring Capabilities in Azure

Introducing Azure Monitor Service

Monitoring is the act of collecting and analyzing data to determine the performance, health, and availability of your business application and the resources that it depends on. An effective monitoring strategy helps you understand the detailed operation of the components of your application. It also helps you increase your uptime by proactively notifying you of critical issues so that you can resolve them before they become problems.

Azure includes multiple services that individually perform a specific role or task in the monitoring space. Together, these services deliver a comprehensive solution for collecting, analyzing, and acting on telemetry from your application and the Azure resources that support them. They can also work to monitor critical on-premises resources to provide a hybrid monitoring environment. Understanding the tools and data that are available is the first step in developing a complete monitoring strategy for your application.



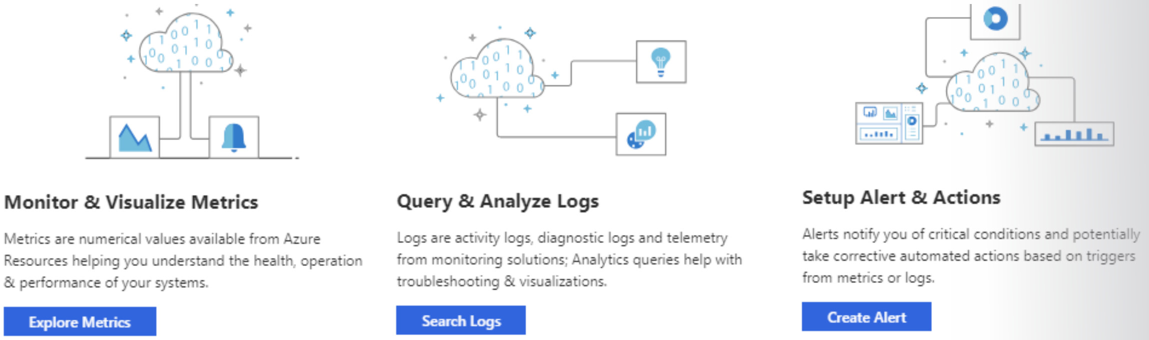
✓ In this course we will cover the services that will help you with materials in the other courses. Specifically, the highlighted items in the diagram are covered in this course.

For more information, you can see:

Monitoring Azure applications and resources - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview**

Video: An Overview of Azure Monitor

Azure Monitor - Key Capabilities



Monitor & Visualize Metrics

Metrics are numerical values available from Azure Resources helping you understand the health, operation & performance of your systems.

Query & Analyze Logs

Logs are activity logs, diagnostic logs and telemetry from monitoring solutions; Analytics queries help with troubleshooting & visualizations.

Setup Alert & Actions

Alerts notify you of critical conditions and potentially take corrective automated actions based on triggers from metrics or logs.

Azure Monitor enables core monitoring for Azure services by allowing the collection of metrics, activity logs, and diagnostic logs. For example, the activity log tells you when new resources are created or modified.

Metrics are available that provide performance statistics for different resources and even the operating system inside a virtual machine. You can view this data with one of the explorers in the Azure portal and create alerts based on these metrics. Azure Monitor provides the fastest metrics pipeline (5 minute down to 1 minute), so you should use it for time critical alerts and notifications.

You can also send these metrics and logs to Azure Log Analytics for trending and detailed analysis, or create additional alert rules to proactively notify you of critical issues as a result of that analysis.

For more information, see:

Get started with Azure Monitor – **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-get-started**

Video: Centralized Monitoring with Azure Monitor

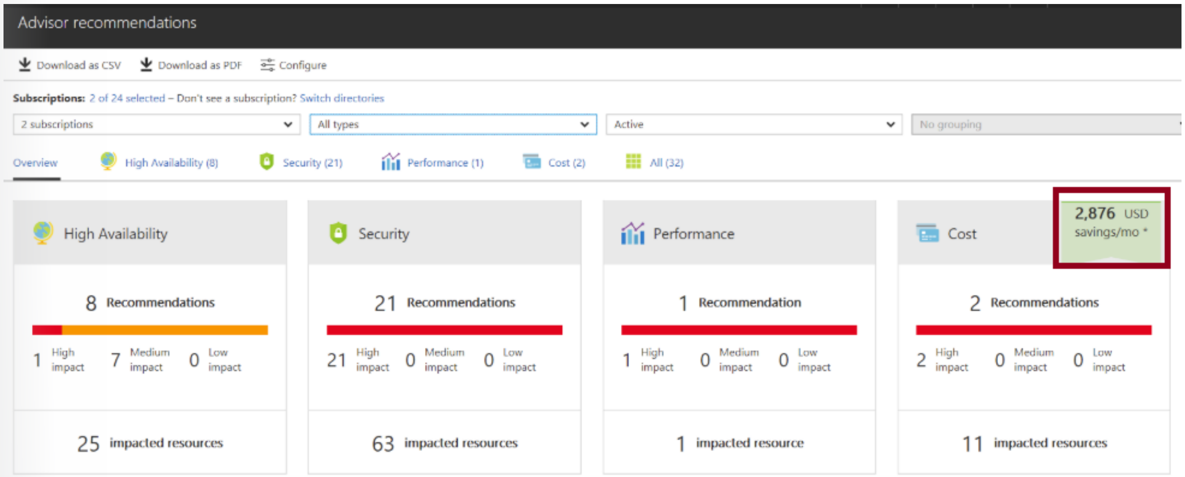
Centralized Monitoring with Azure Monitor

 You can think of Azure Monitor as a platform service which provides a pipeline for metric and log data coming from any Azure resource providers. Alerts and Activity Log are covered in more details in separate lessons. Log Analytics is covered in a separate module.

Azure Advisor

Advisor is a personalized cloud consultant that helps you follow best practices to optimize your Azure deployments. It analyzes your resource configuration and usage telemetry and then recommends solutions that can help you improve the cost effectiveness, performance, high availability, and security of your Azure resources.

The Advisor cost recommendations page helps you optimize and reduce your overall Azure spend by identifying idle and underutilized resources.



Select the recommended action for a recommendation to implement the recommendation. A simple interface will open that enables you to implement the recommendation or refer you to documentation that assists you with implementation.

✓ Advisor provides recommendations for virtual machines, availability sets, application gateways, App Services, SQL servers, and Redis Cache.

For more information, you can see:

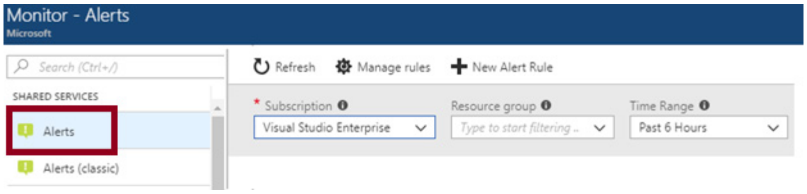
Introduction to Azure Advisor - **https://docs.microsoft.com/en-us/azure/advisor/advisor-overview**

Advisor Cost recommendations - **https://docs.microsoft.com/en-us/azure/advisor/advisor-cost-recommendations**

Azure Alerts

Azure Monitor Alerts

Alerting is now available with Azure Monitor.



* Subscription: Visual Studio Enterprise

Resource group: Type to start filtering ...

Time Range: Past 6 Hours

The Monitor Alerts experience has many benefits.

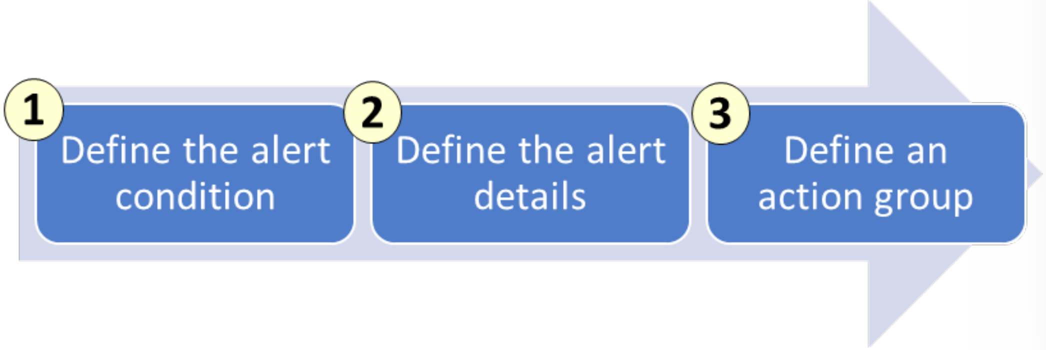
- **Better notification system**. All newer alerts use action groups, which are named groups of notifications and actions that can be reused in multiple alerts.
- **A unified authoring experience**. All alert creation for metrics, logs and activity log across Azure Monitor, Log Analytics, and Application Insights is in one place.
- **View Log Analytics alerts in Azure portal**. You can now also see Log Analytics alerts in your subscription. Previously these were in a separate portal.
- **Separation of Fired Alerts and Alert Rules**. Alert Rules (the definition of the condition that triggers an alert), and Fired Alerts (an instance of the alert rule firing) are differentiated, so the operational and configuration views are separated.
- **Better workflow**. The new alerts authoring experience guides the user along the process of configuring an alert rule, which makes it simpler to discover the right things to get alerted on.

For more information, you can see:

The new alerts experience in Azure Monitor - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-unified-alerts**

Alert Rules

Creating an alert is a three-step task: define the alert condition, define alert details, and define an action group.



1 Define the alert condition

2 Define the alert details

3 Define an action group

1. Define alert condition includes:

- **Target selection**. For example, storage account.

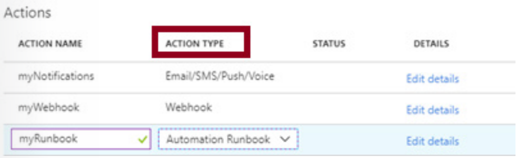
- **Alert criteria**. For example, Used Capacity.
- **Alert logic**. For example, over a six-hour period whenever the Used Capacity is over 1000000 bytes.

1. Define alert details includes: Alert rule name, description, and severity. There are five severity levels, Severity 0 to Severity 4.
2. Define action group. Create an action group to notify your team via email and text messages, or automate actions using webhooks and runbooks.

✓ Take a few minutes to create an alert rule and look at the options.

Action Groups

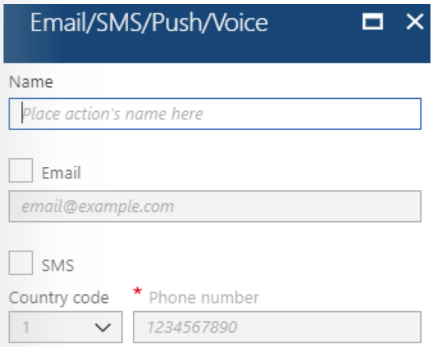
Action groups enable you to configure a list of actions to take when the alert is triggered. Action groups ensure that the same actions are taken each time an alert is triggered. There are several action types you can select when defining the group: Select Email/**SMS**[1]/Push/Voice, **Logic App**[2], **Webhook**[3], **IT Service Management**[4], or Automation Runbook.



Actions

ACTION NAME	ACTION TYPE	STATUS	DETAILS
myNotifications	Email/SMS/Push/Voice		Edit details
myWebhook	Webhook		Edit details
myRunbook ✓	Automation Runbook		Edit details

Each action type is different in the details that must be provided. Here is a screenshot for the Email and SMS configuration.



Email/SMS/Push/Voice

Name

Place action's name here

□ Email

email@example.com

□ SMS

Country code: 1

* Phone number: 1234567890

✓ Take a few minutes to create an action group using the link below.

For more information, you can see:

Create an action group by using the Azure portal - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-action-groups#create-an-action-group-by-using-the-azure-portal**[5]

Action specific information - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-action-groups#action-specific-information**[6]

[1] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-sms-alert-behavior

[2] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-action-groups-logic-app

[3] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-activity-log-alerts-webhook

[4] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-itsmc-overview

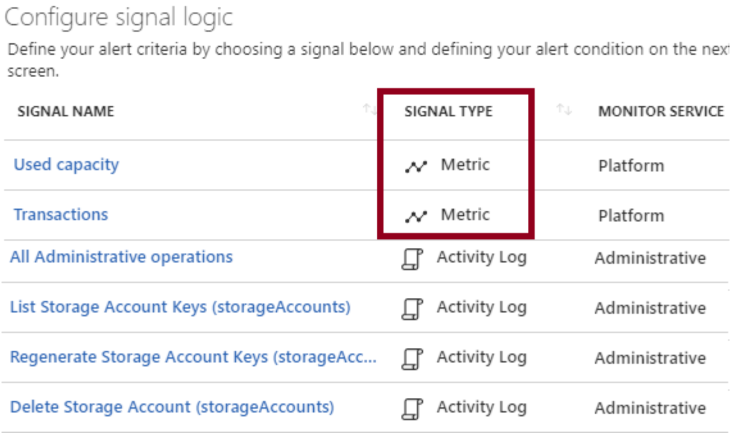
[5] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-action-groups

[6] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-action-groups

Rate limiting for Voice, SMS, emails, Azure App push notifications and webhook posts - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-alerts-rate-limiting**

Signal Types and Metrics

Signals are emitted by the Target resource and can be of several types. Metric, Activity log, Application Insights, and Log are supported Signal types.



Configure signal logic

Define your alert criteria by choosing a signal below and defining your alert condition on the next screen.

SIGNAL NAME	SIGNAL TYPE	MONITOR SERVICE
Used capacity	Metric	Platform
Transactions	Metric	Platform
All Administrative operations	Activity Log	Administrative
List Storage Account Keys (storageAccounts)	Activity Log	Administrative
Regenerate Storage Account Keys (storageAcc...	Activity Log	Administrative
Delete Storage Account (storageAccounts)	Activity Log	Administrative

Newer metric alerts specifically have the following improvements:

- **Improved latency**. Newer metric alerts can run as frequently as every minute. Log alerts still have a longer than 1-minute delay due to the time is takes to ingest the logs.
- **Support for multi-dimensional metrics**. You can alert on dimensional metrics allowing you to monitor an interesting segment of the metric.
- **More control over metric conditions**. You can define richer alert rules. The newer alerts support monitoring the maximum, minimum, average, and total values of metrics.
- **Combined monitoring of multiple metrics**. You can monitor multiple metrics (currently, up to two metrics) with a single rule. An alert is triggered if both metrics breach their respective thresholds for the specified time-period.
- **Metrics from Logs (limited public preview)**. Some log data going into Log Analytics can now be extracted and converted into Azure Monitor metrics and then alerted on just like other metrics.

For more information, you can see:

Alert rule terminology - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-unified-alerts#alert-rules-terminology**[7]

Additional Practice - Alerts

The Azure Activity Log provides a history of subscription-level events in Azure. It offers information about who created, updated, or deleted what resources and when they did it. You can create an Activity Log alert to receive email, SMS, or webhook notifications when an activity occurs that match your alert conditions.

Take a minute to review and try the **Audit and receive notifications about important actions in your Azure subscription**[8] Quickstart. This Quickstart steps through creating a simple network security group,

[7] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-unified-alerts

[8] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitor-quick-audit-notify-action-in-subscription

browsing the Activity Log to understand the event that occurred, and then authoring an Activity Log alert to become notified when any network security group is created going forwards.

You will learn how to:

- Create a network security group
- Browse the Activity Log in the portal
- Browse an event in the Activity log
- Create an Activity log alert
- Test the Activity log alert

✓ Can you see how the activity log lets you monitor activity at the subscription level? If you like, try another practice using the reference link.

For more information, you can see:

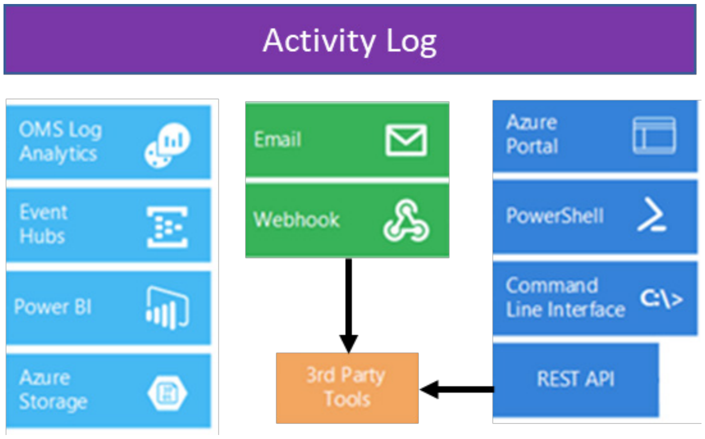
Create, view, and manage alerts using Azure Monitor - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitor-alerts-unified-usage**

Azure Activity Logs

Overview of Activity Log

The Azure Activity Log is a subscription log that provides insight into subscription-level events that have occurred in Azure. This includes a range of data, from Azure Resource Manager operational data to updates on Service Health events. The Activity Log was previously known as “Audit Logs” or “Operational Logs”.

Using the Activity Log, you can determine the ‘what, who, and when’ for any write operation taken on the resources in your subscription. For example, who stopped a service. It provides an audit trail of the activities or operations performed on your resources by someone working on the Azure platform. You can also understand the status of the operation and other relevant properties.



This diagram shows many of the things you can do with the activity log including:

- Send data to Log Analytics for advanced search and alerts.
- Query or manage events in the Portal, PowerShell, CLI, and REST API.
- Stream information to Event Hub.
- Archive data to a storage account.
- Analyze data with Power BI.

✓ The Activity Log differs from **Diagnostic Logs**[9]. Activity Logs provide data about the operations on a resource from the outside (the “control plane”). Diagnostics Logs are emitted by a resource and provide information about the operation of that resource (the "data plane").

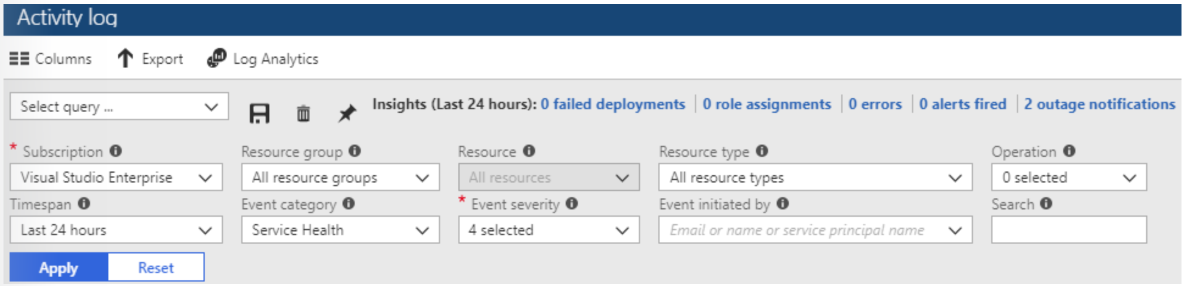
For more information, you can see:

Monitor Subscription Activity with the Azure Activity Log - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs**

[9] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-of-diagnostic-logs

Video: Activity Log

Query the Activity Log



Activity log

Select query ...

Insights (Last 24 hours): 0 failed deployments | 0 role assignments | 0 errors | 0 alerts fired | 2 outage notifications

* Subscription: Visual Studio Enterprise

Resource group: All resource groups

Resource: All resources

Resource type: All resource types

Operation: 0 selected

Timespan: Last 24 hours

Event category: Service Health

* Event severity: 4 selected

Event initiated by: Email or name or service principal name

Search:

In the Azure portal, you can filter your Activity Log by these fields:

- **Subscription**. One or more Azure subscription names.
- **Resource group**. One or more resource groups within those subscriptions.
- **Resource (name)**. The name of a specific resource.
- **Resource type**. The type of resource, for example, Microsoft.Compute/virtualmachines.
- **Operation name**. The name of an Azure Resource Manager operation, for example, Microsoft.SQL/servers/Write.
- **Timespan**. The start and end time for events.
- **Category**. The event category is described in the next topic.
- **Severity**. The severity level of the event (Informational, Warning, Error, Critical).
- **Event initiated by**. The 'caller,' or user who performed the operation.
- **Search**. This is an open text search box that searches for that string across all fields in all events.

✓ Once you have defined a set of filters, you can save it as a query that is persisted across sessions if you ever need to perform the same query with those filters applied again in the future. You can also pin a query to your Azure dashboard to always keep an eye on specific events.

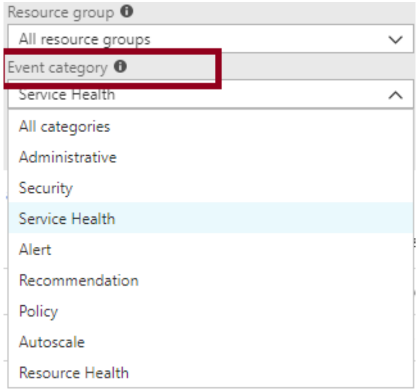
For more information, you can see:

Query the Activity Log in the Azure portal - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs#query-the-activity-log-in-the-azure-portal**[10]

[10] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs

Event Categories

The Activity Log provides several event categories. You may select one or more.



Resource group
All resource groups
Event category
Service Health

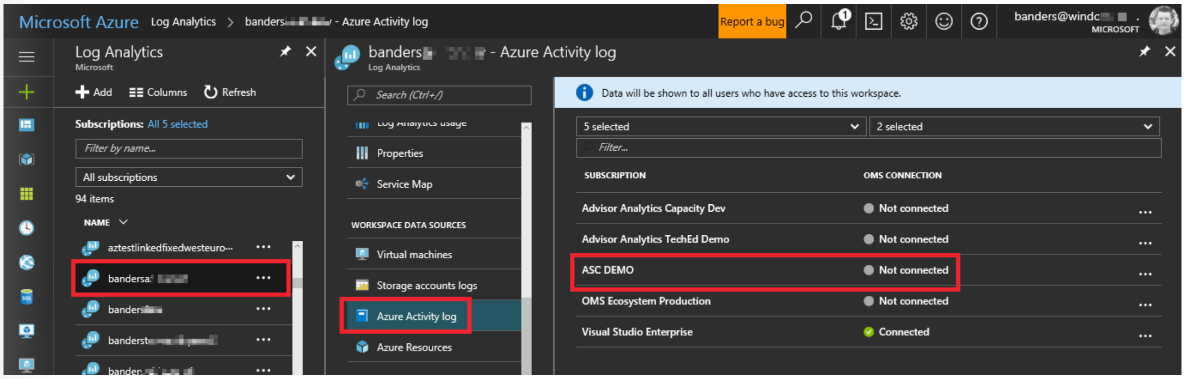
- **Administrative**. This category contains the record of all create, update, delete, and action operations performed through Resource Manager. Examples of the types of events you would see in this category include “create virtual machine” and "delete network security group". The Administrative category also includes any changes to role-based access control in a subscription.
- **Service Health**. This category contains the record of any service health incidents that have occurred in Azure. An example of the type of event you would see in this category is “SQL Azure in East US is experiencing downtime.” Service health events come in five varieties: Action Required, Assisted Recovery, Incident, Maintenance, Information, or Security.
- **Alert**. This category contains the record of all activations of Azure alerts. An example of the type of event you would see in this category is “CPU % on myVM has been over 80 for the past 5 minutes.”
- **Autoscale**. This category contains the record of any events related to the operation of the autoscale engine based on any autoscale settings you have defined in your subscription. An example of the type of event you would see in this category is “Autoscale scale up action failed.”
- **Recommendation**. This category contains recommendation events from certain resource types, such as web sites and SQL servers. These events offer recommendations for how to better utilize your resources.
- **Security**. This category contains the record of any alerts generated by Azure Security Center. An example of the type of event you would see in this category is “Suspicious double extension file executed.”
- **Policy and Resource Health**. These categories do not contain any events; they are reserved for future use.

For more information, you can see:

Categories in the Activity Log - **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs#categories-in-the-activity-log**

Activity Log and Log Analytics

This image shows how easy it is to add the Activity Log Analytics solution to your workspace. Once that is completed the Azure Activity Logs tile will be added to your Overview dashboard. This process applies for when the Log Analytics workspace is in the same Azure subscription, or in a different subscription but in the same Azure Active Directory.



Data will be shown to all users who have access to this workspace.

SUBSCRIPTION	OMS CONNECTION
Advisor Analytics Capacity Dev	● Not connected
Advisor Analytics TechEd Demo	● Not connected
ASC DEMO	● Not connected
OMS Ecosystem Production	● Not connected
Visual Studio Enterprise	✓ Connected

With the Azure Activity Logs tile, you can do many things:

- Analyze the activity logs with pre-defined views.
- Analyze and search activity logs from multiple Azure subscriptions.
- Keep activity logs for longer than the default of 90 days.
- Correlate activity logs with other Azure platform and application data.
- See operational activities aggregated by status.
- View trends of activities happening on each of your Azure services.
- Report on authorization changes on all your Azure resources.
- Identify outage or service health issues impacting your resources.
- Use Log Search to correlate user activities, auto-scale operations, authorization changes, and service health to other logs or metrics from your environment.

✓ Log Analytics collects activity logs and stores the logs for 90 days free of charge. If you store logs for longer than 90 days, you will incur data retention charges for the data stored longer than that period. When you're on the Free pricing tier, activity logs do not apply to your daily data consumption.

For more information, you can see:

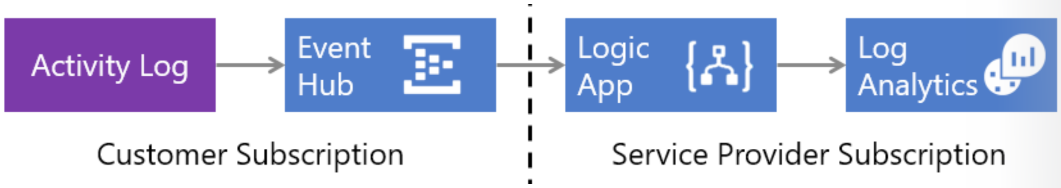
Collect and analyze Azure activity logs in Log Analytics - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-activity**

Collect Across Subscriptions

This topic covers the strategy to collect Azure Activity Logs into a Log Analytics workspace using the Azure Log Analytics Data Collector connector for Logic Apps. Use this strategy when you need to send logs to a workspace in a different Azure Active Directory. For example, if you are a managed service provider, you may want to collect activity logs from a customer's subscription and store them in a Log Analytics workspace in your own subscription.

The basic strategy is to have Azure Activity Log send events to an **Event Hub**[11] where a **Logic App**[12] sends them to your Log Analytics workspace.

[11] https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-what-is-event-hubs
[12] https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-overview



Advantages of this approach include:

- Low latency since the Azure Activity Log is streamed into the Event Hub. The Logic App is then triggered and posts the data to Log Analytics.
- Minimal code is required, and there is no server infrastructure to deploy.

✓ Do you think your organization would benefit from this strategy?

For more information, you can see:

Collect Azure Activity Logs into Log Analytics across subscriptions - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-activity-logs-subscriptions**

Video: Activity Log Alerts

Additional Practice - Activity Log

Azure Activity Logs are a platform service for working with logs and metrics across your subscription. In this practice, try configuring the tasks in the Azure portal first. In most cases, you can also perform the tasks using PowerShell or the CLI.

- **Create an activity log alert**[13]
- **View the Activity Log in the Azure portal**[14]
- **Configure log profiles using the Azure portal**[15]
- **Enable streaming of the Activity Log**[16]
- **Archive the Activity Log using the portal**[17]
- **Configure the Activity Log Analytics solution for your workspaces**[18]

✓ The tasks listed are only a representative sampling of what you can do with Activity Logs. Explore some of the other tasks as you have time. Don't forget to view the associated Activity Log dashboards. (Click the Azure Activity Logs tile to open the Azure Activity Logs dashboard.)

[13] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-activity-log-alerts
[14] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-audit
[15] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview-activity-logs
[16] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-stream-activity-logs-event-hubs
[17] https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-archive-activity-log
[18] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-activity

For more information, see:

Create activity log alerts – **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-activity-log-alerts**

Stream the Azure Activity Log to Event Hubs – **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-stream-activity-logs-event-hubs**

Archive the Azure Activity Log – **https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-archive-activity-log**

Collect and analyze Azure activity logs in Log Analytics - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-activity**

Module 3 Review Questions

Module 3 Review Questions

Azure Monitor Alerting

Your organization has a hybrid Azure infrastructure, with several business-critical resources in Azure as well as on-premises. You need to ensure that you receive a notification when any of the systems are unavailable or experience latency. You decide to use Alerting in Azure Monitor. What metric alerts types are available?

Click for suggested answer ↓

Alerting is now available with Azure Monitor. Signals are emitted by the Target resource and can be of several types including Metric, Activity log, Application Insights, and Log types. Newer metric alerts specifically have the following improvements:

- Improved latency: Newer metric alerts can run as often as every minute. Log alerts still have a longer than 1-minute delay due to the time is takes to ingest the logs.
- Support for multi-dimensional metrics: You can alert on dimensional metrics allowing you to monitor an interesting segment of the metric.
- More control over metric conditions: You can define richer alert rules. The newer alerts support monitoring the maximum, minimum, average, and total values of metrics.
- Combined monitoring of multiple metrics: You can monitor multiple metrics (currently, up to two metrics) with a single rule. An alert is triggered if both metrics breach their respective thresholds for the specified time-period.
- Metrics from Logs (limited public preview): Some log data going into Log Analytics can now be extracted and converted into Azure Monitor metrics and then alerted on just like other metrics.

Log Analytics

Your organization is a Managed Service Provider that manages Azure subscription for several clients. You need to monitor your client’s Azure environments. You decide to use Log Analytics. How should it be used, and what are the benefits for your clients?

Click for suggested answer ↓

Collect Azure Activity Logs into a Log Analytics workspace using the Azure Log Analytics Data Collector connector for Logic Apps. Use this strategy when you need to send logs to a workspace in a different Azure Active Directory. For example, if you are a managed service provider, you may want to collect activity logs from a customer's subscription and store them in a Log Analytics workspace in your own subscription.

The basic strategy is to have Azure Activity Log send events to an Event Hub where a Logic App sends them to your Log Analytics workspace. The benefits for the clients are:

- Analysis of resource configuration.

- Usage telemetry.
- Solution recommendations that can help you improve cost effectiveness, performance, high availability, and security.

Monitoring and Logging

You manage security for an organization. A business-critical system has been shut down. Management suspects that an employee is responsible. You need to audit the environment and find the user responsible. Which tool should you use and why?

Click for suggested answer ↓

The Azure Activity Log is a subscription log that gives insight into subscription-level events that have occurred in Azure. This includes a range of data, from Azure Resource Manager operational data to updates on Service Health events. The Activity Log was previously known as “Audit Logs” or “Operational Logs”.

Using the Activity Log, you can determine the “what, who, and when” for any write operation taken on the resources in your subscription. It gives an audit trail of the activities or operations performed on your resources by someone working on the Azure platform. You can also understand the status of the operation and other relevant properties.

Module 4 Log Analytics

Introduction to Log Analytics

Video: Log Analytics

Log Analytics

Log Analytics helps you collect, correlate, search, and act on log and performance data generated by operating systems and applications. It gives you real-time operational insights using integrated search and custom dashboards to readily analyze millions of records across all your workloads and servers regardless of their physical location. Log Analytics gives you a single interface for consuming and correlating the data, covering both Linux and Windows Server.

Log Analytics Scenarios

One of the challenges with any broad data analytics solution is figuring out where you're going to see value for your organization. Out of all the things that are possible, what does your business need? What we hear from customers is that the following areas all have the potential to deliver significant business value:

Example 1 - Assessing updates

An important part of the daily routine for any IT administrator is assessing systems update requirements and planning patches. Accurate scheduling is critical, as it directly relates to SLAs to the business and can seriously impact business functions. In the past, you had to schedule an update with only limited knowledge of how long the patching would take. Operations Management Suite collects data from all customers performing patches and uses that data to provide an average patching time for specific missing updates. This use of “crowd-sourced” data is unique to cloud systems, and is a great example of how Log Analytics can help meet strict SLAs.

Example 2 - Change tracking

Troubleshooting an operational incident is a complex process, requiring access to multiple data streams. With Operations Management Suite, you can easily perform analysis from multiple angles, using data from a wide variety of sources through a single interface for correlation of information. By tracking changes throughout the environment, Log Analytics helps to easily identify things like abnormal behavior from a specific account, users installing unapproved software, unexpected system reboots or shutdowns, evidence of security breaches, or specific problems in loosely coupled applications.

For more information, you can see:

What is Log Analytics? **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-overview**

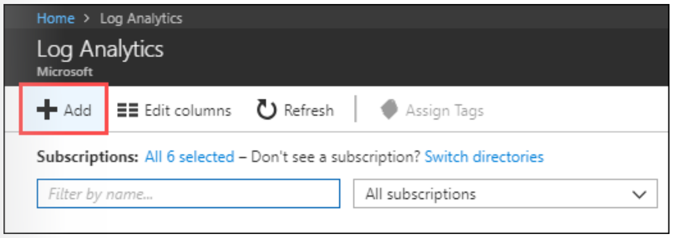
What’s new in Microsoft Operations Management Suite: Log Analytics - **https://blog.tyang.org/wp-content/uploads/2016/04/Whats-New-in-OMS.pdf**

Log Analytics FAQ - **https://docs.microsoft.com/en-us/azure/azure-monitor/platform/log-faq**

Unified Alerts in Log Analytics - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-alerts**

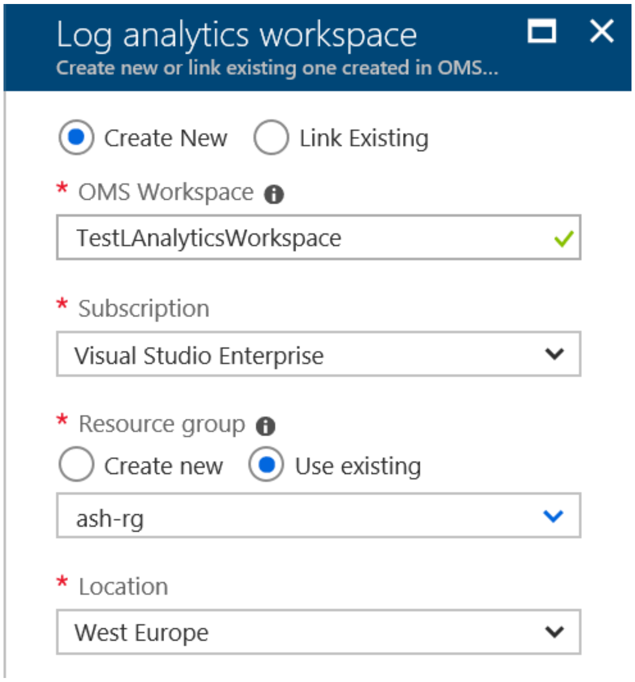
Create a Workspace

To get started with Log Analytics you need to add a workspace. 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.



Subscriptions: All 6 selected – Don't see a subscription? Switch directories

You can them click Create and select your choices for the new workspace.



Log analytics workspace

Create new or link existing one created in OMS...

● Create New ○ Link Existing

* OMS Workspace

TestLAnalyticsWorkspace ✓

* Subscription

Visual Studio Enterprise

* Resource group

○ Create new ● Use existing

ash-rg

* Location

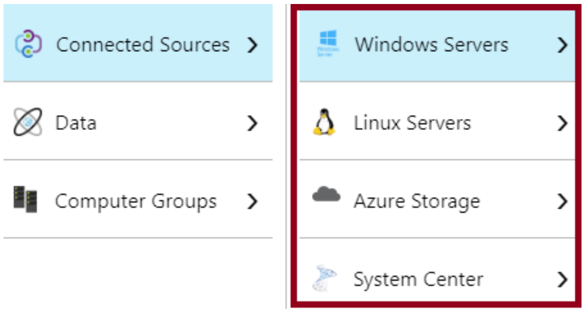
West Europe

For more information, see:

Create a Workspace – **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-quick-collect-azurevm#create-a-workspace**

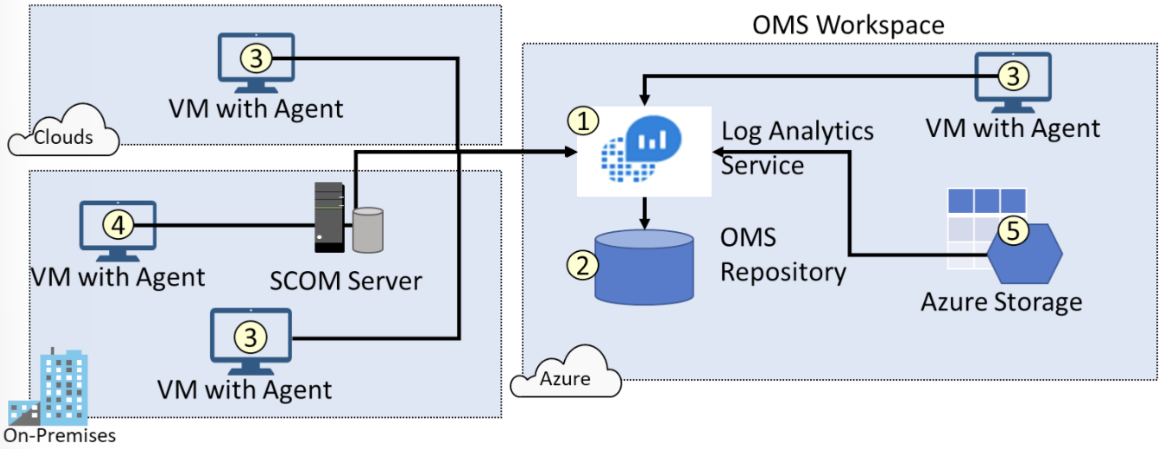
Connected Sources

Connected sources are the computers and other resources that generate data collected by Log Analytics. This can include agents installed on **Windows**[1] and **Linux**[2] computers that connect directly or agents in a connected **System Center Operations Manager management group**[3] . Log Analytics can also collect data from **Azure storage**[4].



This following diagram shows how Connected Sources flow data to the Log Analytics service.

[1] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-windows-agents
[2] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-linux-agents
[3] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-om-agents
[4] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-azure-storage



Ensure you can locate each of the following.

- The Log Analytics service (1) collects data and stores it in the OMS repository (2). The OMS Repository is hosted in Azure. Connected Sources provide information to the Log Analytics service.
- Computer agents (3) generate data to the Log Analytics service. These agents can run on Windows or Linux computers, virtual or physical computers, on-premises or cloud computers, and Azure or other cloud providers.
- A System Center Operations Manager (SCOM) management group can be connected to Log Analytics. SCOM agents (4) communicate with management servers which forward events and performance data to Log Analytics.
- An Azure storage account (5) can also collect Azure Diagnostics data from a worker role, web role, or virtual machine in Azure. This information can be sent to the Log Analytics service.

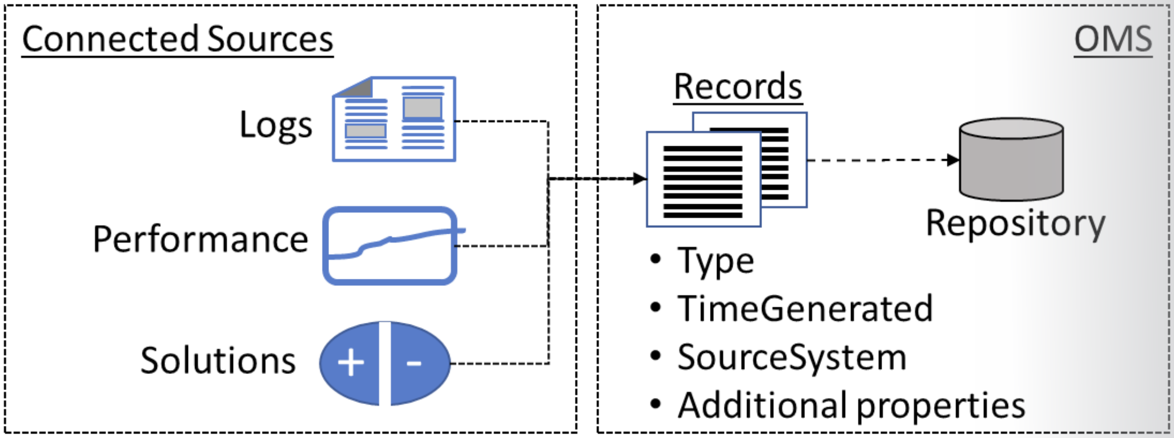
For more information, you can see:

Connecting Computers to the Log Analytics Service - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-windows-agents#system-requirements-and-required-configuration**[5]

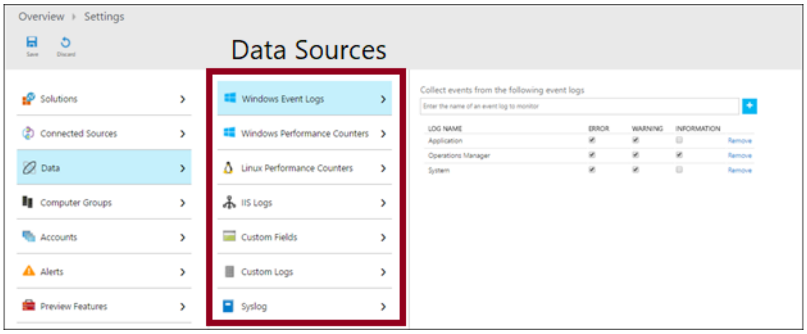
Data Sources

Data sources are the different kinds of data collected from each connected source. These can include events and performance data from Windows and Linux agents, in addition to sources such as 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.

[5] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-windows-agents



When you configure the Log Analytics settings you can see the data sources that are available. Data sources include: Windows Event Logs, Windows Performance Counters, Linux Performance Counters, IIS Logs, Custom Fields, Custom Logs, and Syslog. Each data source has additional configuration options. For example, the Windows Event Log can be configured to forward Error, Warning, or Informational messages.



Collect events from the following event logs

LOG NAME	ERROR	WARNING	INFORMATION	
Application	☑	☑	□	Remove
Operations Manager	☑	☑	☑	Remove
System	☑	☑	□	Remove

For more information, you can see:

Data Sources in Log Analytics - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-data-sources**

Demonstration: Log Analytics

Additional Practice - Visualize Data

Log Analytics dashboards can visualize all your saved log searches, giving you the ability to find, correlate, and share IT operational data in the organization. This practice covers creating a log search that will be used to support a shared dashboard that will be accessed by your IT operations support team.

Take a few minutes to try the **Create and share dashboards of Log Analytics data**[6] tutorial. You learn how to:

- Create a shared dashboard in the Azure portal.
- Visualize a performance log search.
- Add a log search to a shared dashboard.
- Customize a tile in a shared dashboard.

✓ In this tutorial, you learned how to create a dashboard in the Azure portal and add a log search to it. In the next tutorial you will learn the different responses you can implement based on log search results.

Additional Practice - Alert on Data

Azure Alerts automatically runs specified log queries at regular intervals. If the results of the log query match your criteria, then an alert record is created.

Take a few minutes to try the **Respond to events with Azure Monitor Alerts**[7] tutorial. You learn how to:

- Create an alert rule.
- Configure an Action Group to send an e-mail notification.

✓ In this tutorial, you learned to create an alert based on your Log Analytics workspace and then defined a custom log search. You then activated your alert and created an Action Group to send an email notification each time the alert is triggered.

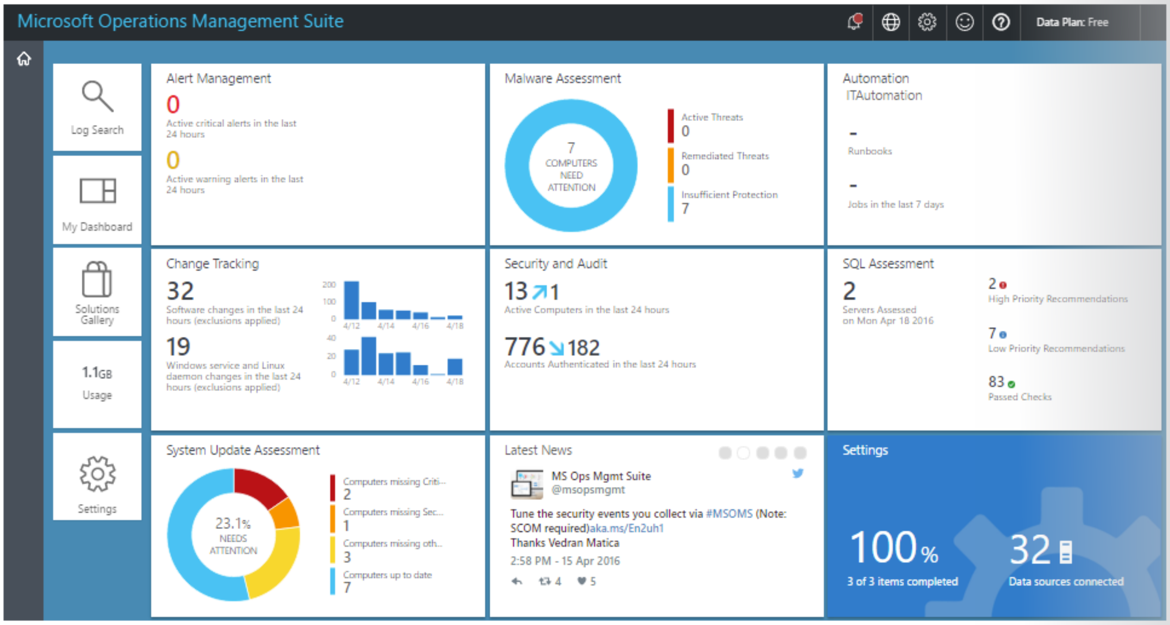
[6] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-tutorial-dashboards

[7] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-tutorial-response

Querying and Analyzing Log Analytics Data

Analyzing Log Analytics Data

Most of your interaction with Log Analytics will be through the OMS portal (see image below) which runs in any browser and provides you with access to configuration settings and multiple tools to analyze and act on collected data. From the portal, you can leverage log searches where you construct queries to analyze collected data, dashboards which you can customize with graphical views of your most valuable searches, and solutions which provide additional functionality and analysis tools.


Microsoft Operations Management Suite
Data Plan: Free
Log Search
My Dashboard
Solutions Gallery
1.1GB Usage
Settings
Alert Management
0
Active critical alerts in the last 24 hours
0
Active warning alerts in the last 24 hours
Malware Assessment
7 COMPUTERS NEED ATTENTION
Active Threats 0
Remediated Threats 0
Insufficient Protection 7
Automation
ITAutomation
-
Runbooks
-
Jobs in the last 7 days
Change Tracking
32
Software changes in the last 24 hours (exclusions applied)
19
Windows service and Linux daemon changes in the last 24 hours (exclusions applied)
200
100
0
4/12
4/14
4/16
4/18
40
20
0
4/12
4/14
4/16
4/18
Security and Audit
13 ↗ 1
Active Computers in the last 24 hours
776 ↘ 182
Accounts Authenticated in the last 24 hours
SQL Assessment
2
Servers Assessed on Mon Apr 18 2016
2 ●
High Priority Recommendations
7 ●
Low Priority Recommendations
83 ●
Passed Checks
System Update Assessment
23.1% NEEDS ATTENTION
Computers missing Criti... 2
Computers missing Sec... 1
Computers missing oth... 3
Computers up to date 7
Latest News
MS Ops Mgmt Suite
@msopsmgmt
Tune the security events you collect via #MSOMS (Note: SCOM required)aka.ms/En2uh1
Thanks Vedran Matica
2:58 PM - 15 Apr 2016
4
5
Settings
100%
3 of 3 items completed
32
Data sources connected

✓ In response to customer feedback to consolidate monitoring and management of both on premises and Azure workloads into a single user experience, OMS portal capabilities have been added into the Azure portal

For more information, you can see:

OMS portal moving to Azure – **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-oms-portal-transition**

Additional Practice - Collect and Analyze Data

This is a two part practice. In the first practice, you will collect performance data from virtual machines. In the second practice, you will create and edit queries to analyze the data.

Part 1

Take a few minutes to try the **Collect data about Azure Virtual Machines**[8] QuickStart. In this QuickStart, you learn how to:

- Create a workspace.
- Enable Log Analytics on virtual machines.
- Collect event and performance data.

[8] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-quick-collect-azurevm

- View the data collected.

Part 2

Take a few minutes to try the **View or analyze data collected with Log Analytics log search**[9] tutorial. In this tutorial, you learn how to:

- Perform a simple search of event data and use features to modify and filter the results.
- Learn how to work with performance data.

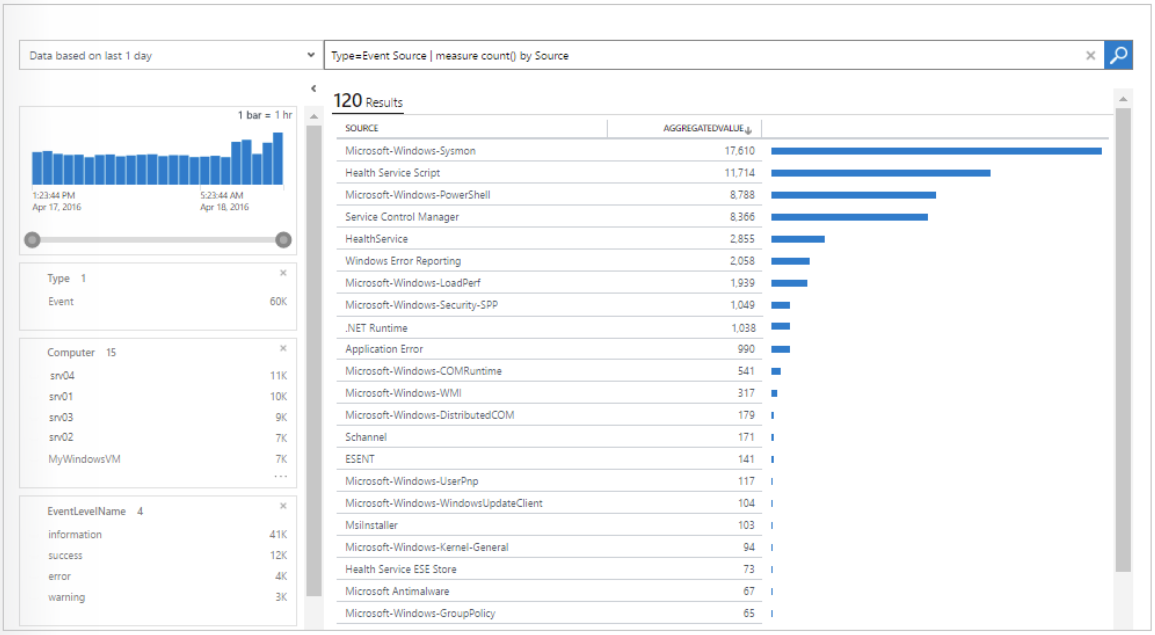
✓ In the next tutorial you will learn how to visualize the data by creating a dashboard.

For more information, you can see:

Writing a query - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-log-search#writing-a-query**[10]

Log Analytics Querying

Log Analytics provides a query syntax to quickly retrieve and consolidate data in the repository. You can create and save Log Searches to directly analyze data in the OMS portal or have log searches run automatically to create an alert if the results of the query indicate an important condition.



Type=Event Source | measure count() by Source

Type	1
Event	60K

Computer	15
srv04	11K
srv01	10K
srv03	9K
srv02	7K
MyWindowsVM	7K

EventLevelName	4
information	41K
success	12K
error	4K
warning	3K

SOURCE	AGGREGATEDVALUE
Microsoft-Windows-Sysmon	17,610
Health Service Script	11,714
Microsoft-Windows-PowerShell	8,788
Service Control Manager	8,366
HealthService	2,855
Windows Error Reporting	2,058
Microsoft-Windows-LoadPerf	1,939
Microsoft-Windows-Security-SPP	1,049
.NET Runtime	1,038
Application Error	990
Microsoft-Windows-COMRuntime	541
Microsoft-Windows-WMI	317
Microsoft-Windows-DistributedCOM	179
Schannel	171
ESENT	141
Microsoft-Windows-UserPnp	117
Microsoft-Windows-WindowsUpdateClient	104
MsiInstaller	103
Microsoft-Windows-Kernel-General	94
Health Service ESE Store	73
Microsoft Antimalware	67
Microsoft-Windows-GroupPolicy	65

To give a quick graphical view of the health of your overall environment, you can add visualizations for saved log searches to your dashboard. To analyze data outside of Log Analytics, you can export the data from the repository into tools such as Power BI or Excel. You can also leverage the Log Search API to build custom solutions that leverage Log Analytics data or to integrate with other systems.

For more information, you can see:

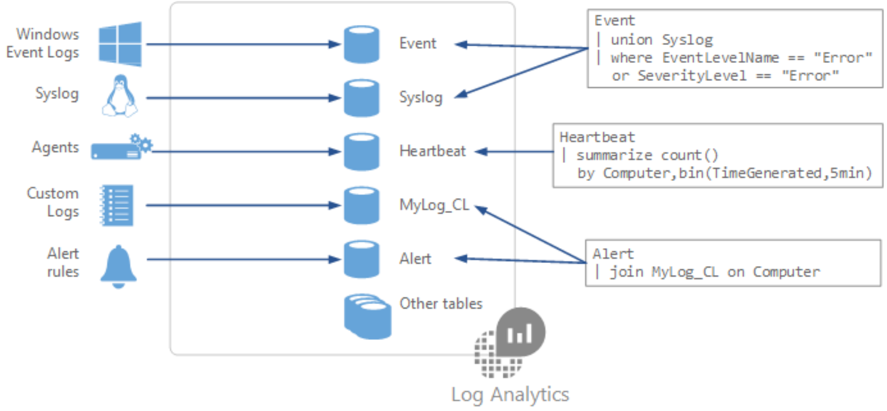
Azure Log Analytics – meet our new query language - **https://azure.microsoft.com/en-us/blog/azure-log-analytics-meet-our-new-query-language-2/**

[9] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-tutorial-viewdata
[10] https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-log-search

Querying Language Syntax

When you build a query, you start by determining which tables have the data that you're looking for. Each data source and solution stores its data in dedicated tables in the Log Analytics workspace. Documentation for each data source and solution includes the name of the data type that it creates and a description of each of its properties. Many queries will only require data from a single table, but others may use a variety of options to include data from multiple tables.

The main query tables are: Event, Syslog, Heartbeat, and Alert.



```
Event
| union Syslog
| where EventLevelName == "Error"
  or SeverityLevel == "Error"
```

```
Heartbeat
| summarize count()
  by Computer,bin(TimeGenerated,5min)
```

```
Alert
| join MyLog_CL on Computer
```

The basic structure of a query is a source table followed by a series of operators separated by a pipe character |. You can chain together multiple operators to refine the data and perform advanced functions. For example, this query returns a count of the top 10 errors in the Event log during the last day. The results are in descending order.

```
Event
| where (EventLevelName == "Error")
| where (TimeGenerated > ago(1days))
| summarize ErrorCount = count() by Computer
| top 10 by ErrorCount desc
```

✓ You can try this query and many others during the Practice: Log Analytics Queries.

For more information, you can see:

Understanding log searches in Log Analytics - **https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-log-search**

Demonstration: Log Analytics Querying

Additional Practice - Log Analytics Queries

Take a few minutes to access the **Log Analytics Querying Demonstration**[11] page. This page provides a live demonstration workspace where you can run and test queries. Some of the testing queries are:

- See the volume of data collected in the last 24 hours in intervals of 30 minutes.
- Chart the distribution of billable data by type, over the last 24 hours.
- Find out which computers were alive in the past 2 days but haven't sent any data in the last 6 hours.

✓ The reference link has additional queries you can try. Is there a specific query that will help with your day to day tasks?

For more information, you can see:

Getting Started with the Analytics Portal - **https://portal.loganalytics.io/demo#/discover/home**[12]

[11] https://portal.loganalytics.io/demo

[12] https://portal.loganalytics.io/demo

Module 4 Review Questions

Module 4 Review Questions

Log Analytics

You work for a cloud solution provider as a technical pre-sales consultant. A customer is inquiring about Log Analytics. You need to help the customer understand why they should invest in the tool and what the benefits are. Can you give some specific examples of how to use Log Analytics?

Click for suggested answer ↓

Example 1 - **Assessing updates**. An important part of the daily routine for any IT administrator is assessing systems update requirements and planning patches. Accurate scheduling is critical, as it relates to SLAs to the business and can seriously affect business functions. In the past, you had to schedule an update with only limited knowledge of how long the patching would take. Operations Management Suite collects data from all customers performing patches and uses that data to give an average patching time for specific missing updates. This use of "crowd-sourced" data is unique to cloud systems and is a great example of how Log Analytics can help meet strict SLAs.

Example 2 - **Change tracking**. Troubleshooting an operational incident is a complex process. You often need access to multiple data streams. With Operations Management Suite, you can easily perform analysis from multiple angles, using data from a wide variety of sources through a single interface for correlation of information. By tracking changes throughout the environment, Log Analytics helps to easily identify things like abnormal behavior from a specific account, users installing unapproved software, unexpected system reboots or shutdowns, evidence of security breaches, or specific problems in loosely coupled applications.

Monitor Resource Usage

An organization has a hybrid Azure infrastructure and has business-critical resources in Azure as well as the on-premises environment. You need to monitor resource usage. Describe how Connected Sources flow data to the Log Analytics service. Can you draw a diagram?

Click for suggested answer ↓

- The Log Analytics service collects data and stores it in the OMS repository. The OMS Repository is hosted in Azure. Connected Sources provide information to the Log Analytics service.
- Computer agents generate data to the Log Analytics service. These agents can run on Windows or Linux computers, virtual or physical computers, on-premises or cloud computers, and Azure or other cloud providers.
- You can connect a System Center Operations Manager (SCOM) management group to Log Analytics. SCOM agents communicate with management servers which forward events and performance data to Log Analytics.
- An Azure storage account can also collect Azure Diagnostics data from a worker role, web role, or virtual machine in Azure. This information can be sent to the Log Analytics service.

Data Visualization

You implement Log Analytics for an organization. You collect substantial amounts of data. You need to visualize the data. What tools can you use?

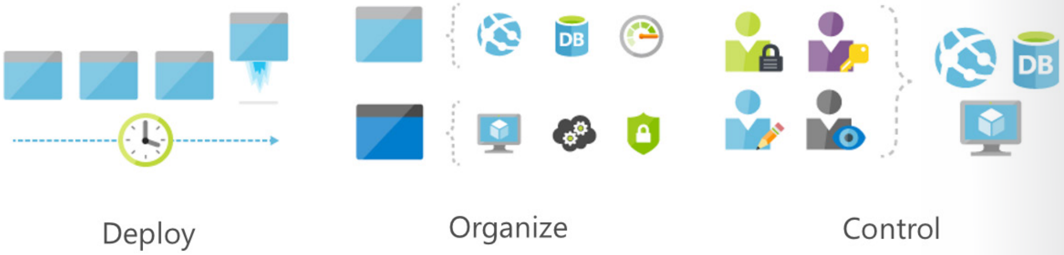
Click for suggested answer ↓

To give a quick graphical view of the health of your overall environment, you can add visualizations for saved log searches to your dashboard. To analyze data outside of Log Analytics, you can export the data from the repository into tools such as Power BI or Excel. You can also use the Log Search API to build custom solutions that leverage Log Analytics data or to integrate with other systems.

Module 5 Azure Resource Manager

ARM Templates

Azure Resource Manager



Azure Resource Manager introduces an entirely new way of thinking about your Azure resources. Instead of creating and managing individual resources, you begin by imagining a complex service, such as a blog, a photo gallery, a SharePoint portal, or a wiki. You use a template – a resource model of the service – to create a resource group with the resources that you need to support the service. Then, you can manage and deploy that resource group as a logical unit. There are three primary concepts in Resource Manager:

- **Resource**. A resource is simply a single service instance in Azure. Most services in Azure can be represented as a resource. For example, a Web App instance is a resource. An App Service Plan is also a resource. Even a SQL Database instance is a resource.
- **Resource Group**. A resource group is a logical grouping of resources. For example, a Resource Group where you would deploy a VM compute instance may be composed of a Network Interface Card (NIC), a Virtual Machine, a Virtual Network, and a Public IP Address.
- **Resource Group Template**. A resource group template is a JSON file that allows you to declaratively describe a set of resources. These resources can then be added to a new or existing resource group.

For example, a template could contain the configuration necessary to create two API App instances, a Mobile App instance and a Document DB instance.

For more information, you can see:

Azure Resource Manager - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-overview**

Video: Azure Resource Manager

Azure Resource Manager

The Azure portal interface has changed slightly since this video was recorded. However, it is still a good overview of what is available through Resource Manager.

Video: ARM Templates

ARM Templates

As a System Administrator you will work with ARM templates every day. Although this is an older video it does a good job of explaining what ARM Templates and how they are used.

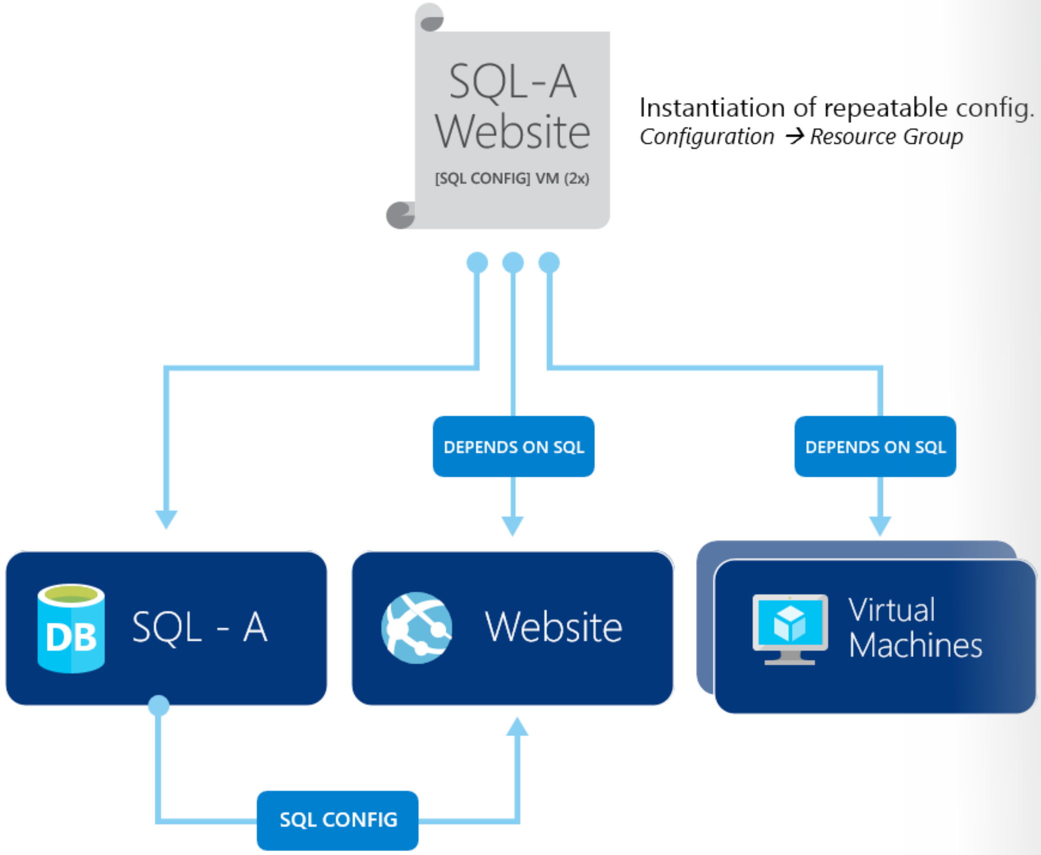
Template Advantages

Templates are generally preferred to manually deploying resources for a number of reasons:

- A template can ensure idempotency, which from a RESTful service standpoint means that multiple identical requests produce the same results as a single request. This results in no side effects on the server, and the result of the request may differ, because the resource state has changed between requests. If you deploy an identical template to multiple resource groups, they would functionally be the same.
- A template can simplify orchestration as you only need to deploy the template to deploy all of your resources. Normally this would take multiple operations.
- A template allows you to configure multiple resources simultaneously and use variables/parameters/functions to create dependencies between resources. For example you can require that a VM is

created before a Web App because you need the VM's public IP address for one of the Web App's settings. Another example is to require that a Storage account is created before a VM so that you can place the VHDs in that storage account.

- A template is a JSON file so it can be configured and managed using a source control provider, and used as part of any continuous integration process.
- Templates can parameterize input and output values so they can be reused across many different scenarios. Templates can also be nested so you can reuse smaller templates as part of a larger orchestration.



Instantiation of repeatable config.
Configuration → Resource Group

Demonstration: ARM Templates

Demonstration ARM Templates

In this introduction to using ARM templates, explore the different deployment methods for services within Microsoft Azure.

Additional Practice - Deploy and ARM Template

Take a few minutes to try this exercise where you'll **deploy an Azure Resource Manager template using PowerShell**[1]. The template that you create deploys a single virtual machine running Windows Server in a new virtual network with a single subnet.

In this exercise, you will:

- Launch the 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 ARM template JSON code. Simply use the copy functionality that is provided in the exercise.

For more information, you can see:

Azure Resource Manager overview – **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-overview?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json**

Troubleshoot common ARM deployment errors – **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-common-deployment-errors**

[1] https://docs.microsoft.com/en-us/azure/virtual-machines/windows/ps-template

Resource Groups

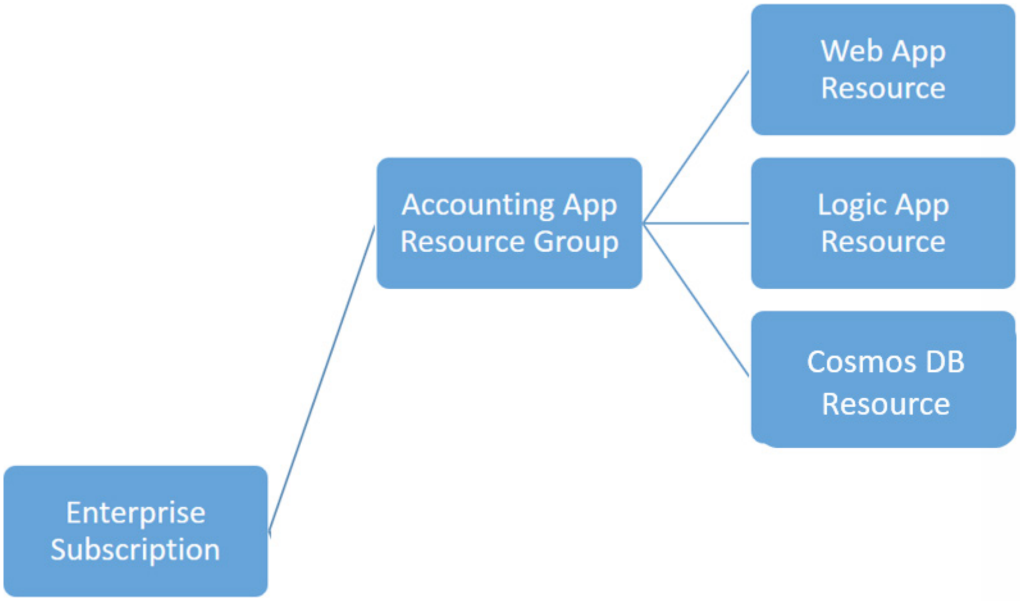
Resource Group Deployments

Resources can be deployed to any new or existing resource group. Deployment of resources to a resource group becomes a job where you can track the template execution. If deployment fails, the output of the job can describe why the deployment failed. Whether the deployment is a single resource to a group or a template to a group, you can use the information to fix any errors and redeploy. Deployments are incremental; if a resource group contains 2 web apps and you decide to deploy a third, the existing web apps will not be removed. Currently, immutable deployments are not supported in a resource group. To implement an immutable deployment, you must create a new resource group.

Resource Groups

Resource Groups are at their simplest a container for multiple resources. There are a couple of small rules for resource groups.

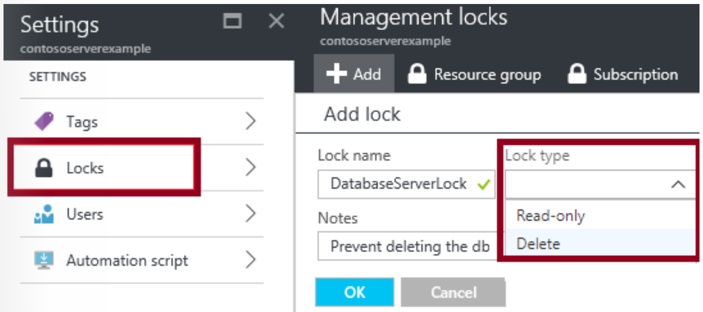
- Resources can only exist in one resource group.
- Resource Groups cannot be renamed.
- Resource Groups can have resources of many different types (services).
- Resource Groups can have resources from many different regions.



By scoping permissions to a resource group, you can add/remove and modify resources easily without having to recreate assignments and scopes.

Resource Manager Locks

A common concern with resources provisioned in Azure is the ease with which they can be deleted. An over-zealous or careless administrator can accidentally erase months of work with a few clicks. Resource manager locks allow organizations to put a structure in place that prevents the accidental deletion of resources in Azure. You can associate the lock with a subscription, resource group, or resource. Locks are inherited by child resources.



Lock name: DatabaseServerLock ✓

Notes: Prevent deleting the db

Lock type: Delete

Locks come in two varieties.

- Read-Only locks, which prevent any changes to the resource.
- Delete locks, which prevent deletion.

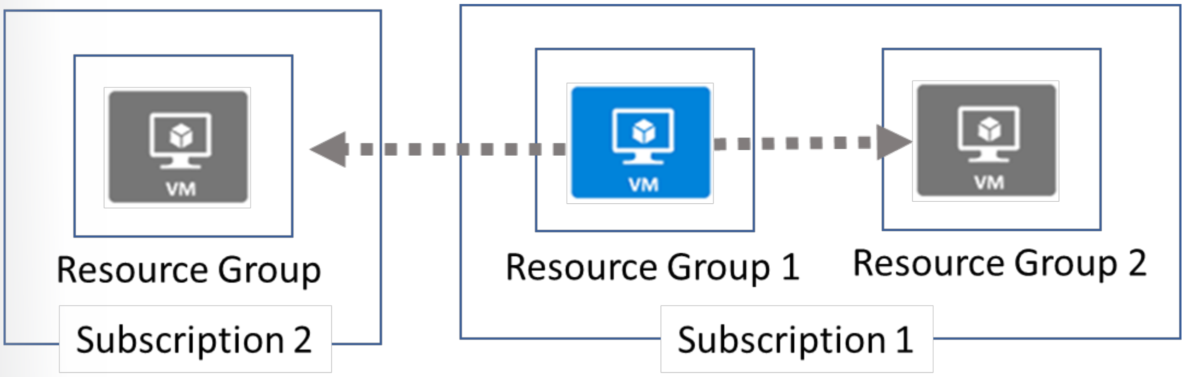
✓ Only Owner and User Access Administrator roles can create or delete management locks.

For additional information, see:

Lock resources to prevent unexpected changes: **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources**

Moving Resources

Sometimes you may need to move resources to either a new subscription or a new resource group in the same subscription.



When moving resources, both the source group and the target group are locked during the operation. Write and delete operations are blocked on the resource groups until the move completes. This lock means you can't add, update, or delete resources in the resource groups, but it doesn't mean the resources are frozen. For example, if you move a virtual machine to a new resource group, an application accessing the virtual machine experiences no downtime.

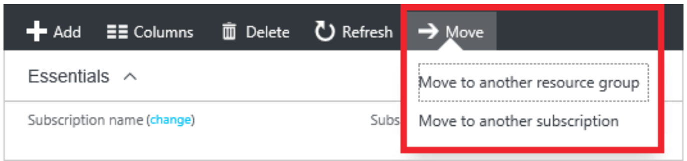
Before beginning this process:

- Review **services that can be moved**[2].
- Review **services that cannot be moved**[3].

[2] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-move-resources

[3] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-move-resources

To move resources, select the resource group containing those resources, and then select the Move button. Select the resources to move and the destination resource group. Acknowledge that you need to update scripts.


Add
Columns
Delete
Refresh
Move
Essentials
Subscription name (change)
Subs
Move to another resource group
Move to another subscription

✓ Just because a service can be moved doesn't mean there aren't restrictions. For example, you can move a virtual network, but you must also move its dependent resources, like gateways. Learn more at the reference link.

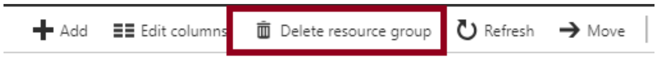
For more information, you can see:

Move resources to new resource group or subscription - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-move-resources**

Remove Resource Groups and Resources

Resource Groups

Use caution when deleting a resource group. Deleting a resource group deletes all the resources contained within it. That resource group might contain resources that resources in other resource groups depend on.



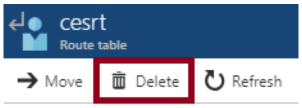
Using PowerShell to delete resource groups

To remove a resource group use, **Remove-AzureRMResourceGroup**. In this example, we are removing the ContosoRG01 resource group from the subscription. The cmdlet prompts you for confirmation and returns no output.

```
Remove-AzureRmResourceGroup -Name "ContosoRG01"
```

Resources

You can also delete individual resources within a resource group.



Using PowerShell to delete a resource

To remove an individual resource use, **Remove-AzureRmResource**. You will need the ResourceId. In this example, we are removing a website.

```
    Remove-AzureRmResource -ResourceId "/subscriptio
ns/00000000-0000-0000-0000-000000000000/resourceGroups/ResourceGroup11/
providers/Microsoft.Web/sites/ContosoSite"
```

Video: Removing a Resource Group in Azure

Removing a Resource Group in Azure

The Azure portal interface has changed slightly since this video was recorded. However, it is still a good overview of what is available through Resource Manager.

Additional Practice - Lock Resources

Resource Manager locks apply only to operations that happen in the management plane. To create or delete management locks, you must have access to Microsoft.Authorization/* or Microsoft.Authorization/locks/* actions. Of the built-in roles, only Owner and User Access Administrator are granted those actions.

Take a few minutes to access the **Lock resources to prevent unexpected changes**[4] page. In this **practice**[5], you will learn to add a lock for a resource using the Azure portal.

Next, read through the **example template**[6] that creates an app service plan, a web site, and a lock on the web site. **Try using PowerShell to**[7]:

- Deploy the example template
- Let information about all locks on resources and resource groups
- Delete a lock

✓ You can also perform the same exercise with the **Azure CLI**[8] if you prefer.

For more information, you can see:

Lock resources to prevent unexpected changes - **https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources**

[4] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources
[5] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources
[6] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources
[7] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources
[8] https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources

Module 5 Review Questions

Module 5 Review Questions

ARM Templates

You administer Azure resources for an organization. You are using Azure Resource Manager templates to deploy and manage Azure services. You need to define the Azure Resource Manager type for each of the following: Web App, SQL Database, and VM Compute instance. What type is each of the resources, and why are they divided in these types?

Click for suggested answer ↓

A Web App is a resource. SQL Database is a Resource. A VM Compute instance is a Resource Group. There are three primary concepts in Resource Manager:

- **Resource**: A resource is simply a single service instance in Azure. Most services in Azure can be represented as a resource. For example, a Web App instance is a resource. An App Service Plan is also a resource. Even a SQL Database instance is a resource.
- **Resource Group**: A resource group is a logical grouping of resources. For example, a Resource Group where you would deploy a VM compute instance may be composed of a Network Interface Card (NIC), a Virtual Machine, a Virtual Network, and a Public IP Address.
- **Resource Group Template**: A resource group template is a JSON file that allows you to declaratively describe a set of resources. These resources can then be added to a new or existing resource group. For example, a template could contain the configuration necessary to create two API App instances, a Mobile App instance and a Document DB instance.

ARM Templates

You are using Azure Resource Manager (ARM) templates to deploy multiple resources. What are some benefits of using ARM templates?

Click for suggested answer ↓

A template can ensure idempotency, which from a RESTful service standpoint means that multiple identical requests produce the same results as a single request. This results in no side effects on the server, and the result of the request may differ, because the resource state has changed between requests. If you deploy an identical template to multiple resource groups, they would functionally be the same.

A template can simplify orchestration as you only need to deploy the template to deploy all your resources. Normally this would take multiple operations.

A template allows you to configure multiple resources simultaneously and use variables/parameters/functions to create dependencies between resources. For example, you can require virtual machine creation before the Web App is created because you need the public IP address for one of the Web App's settings. Another example is to require that Storage account creation before virtual machine creation so that you can place the VHDs in that storage account.

A template is a JSON file. You can manage and configure the file by using a source control provider and any continuous integration process.

Templates can parameterize input and output values, so you can reuse them across many different scenarios. Templates can also be nested so you can reuse smaller templates as part of a larger orchestration.

Resource Groups

You need to implement Resource Manager Resource Groups. What rules exists for Resource Groups?

Click for suggested answer ↓

Resources can only exist in one resource group.

Resource Groups cannot be renamed.

Resource Groups can have resources of many different types (services).

Resource Groups can have resources from many different regions.

Module 6 Azure Tips, Tricks, and Tools

The Azure Portal

Video: Azure Portal Updates

Azure Portal Updates

As a System Administrator you will using the Azure portal all the time. In this lesson, we will work with the Portal and learn some tips and tricks. Even a seasoned professional will find something useful in this lesson.

Video: Keyboard Shortcuts

Keyboard Shortcuts

Who doesn’t love keyboard shortcuts. Here are a few for the portal:

```
Ctrl+/Search blade menu items
G+/ Search resources (global)
G+D Go to the Dashboard
G+A Go to All resources
```

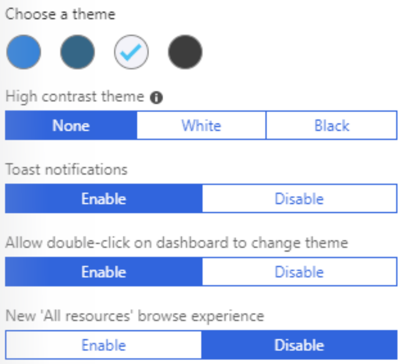
 This video and the next ones are courtesy of Michael Crump in the Azure Product team. We are highlighting just a few of the many tips and tricks videos available on his blog. Be sure to check out the complete list.

Azure Tips and Tricks - The Complete List - **https://www.michaelcrump.net/azure-tips-and-tricks-complete-list/**

Video: Portal Themes

Portal Themes

You can customize your portal theme. Which do you prefer?



Choose a theme

● ● [✓ ○] ●

High contrast theme ⓘ

[✓ None] White Black

Toast notifications

[✓ Enable] Disable

Allow double-click on dashboard to change theme

[✓ Enable] Disable

New 'All resources' browse experience

Enable [✓ Disable]

Video: Customize the Dashboard

Customize the Dashboard

After watching the video take a few minutes to customize your Dashboard.

Video: Dashboard Favorites

Dashboard Favorites

You can easily add and reorder favorites to customize your Azure service list. Another tip is that you can Shift+Space to toggle favorites after typing a name. For example, if you type cosmos and press Shift+Space then you can toggle the favorite flag.

✓ As you have time, check out other Tips and Tricks videos.

Azure Tools and Environment

Video: Cloud Shell

Cloud Shell

Azure Cloud Shell is an interactive, browser-accessible shell for managing Azure resources. It provides the flexibility of choosing the shell experience that best suits the way you work. Linux users can opt for a Bash experience, while Windows users can opt for PowerShell.

For more information, you can see:

Overview of Azure Cloud Shell - **https://docs.microsoft.com/en-us/azure/cloud-shell/overview?view=azurermps-6.5.0**

Video: Azure CLI

Azure CLI

The Azure CLI 2.0 is Microsoft's cross-platform command line experience for managing Azure resources. You can use it in your browser with Azure Cloud Shell, or install it on macOS, Linux, or Windows and run it from the command line.

For more information, you can see:

Get started with Azure CLI 2.0 - **https://docs.microsoft.com/en-us/cli/azure/get-started-with-azure-cli?view=azure-cli-latest**

Video: Azure PowerShell

Azure PowerShell

Azure PowerShell provides a set of cmdlets that use the Azure Resource Manager model for managing your Azure resources. You can use it in your browser with Azure Cloud Shell, or you can install it on your local machine and use it in any PowerShell session.

For more information, you can see:

Azure PowerShell - **https://docs.microsoft.com/en-us/powershell/azure/overview?view=azurermps-6.5.0**

Video: Azure PowerShell Cross-Platform

Azure PowerShell Cross-Platform

In this video, Aaron and Scott check out the latest updates in Azure PowerShell, from simplified scenarios to `–AsJob` backgrounding support. This new functionality is now available on Mac, Linux, and Windows with PowerShell Core 6.

Video: Resource Explorer

Resource Explorer

Be sure to try the Resource Explorer website - **https://resources.azure.com/subscriptions**

Module 7 Lab-Manage Azure Subscriptions and Resources

Lab

Lab

Scenario

Adatum Corporation wants to use Azure Role Based Access Control and Azure Policy to control provisioning and management of their Azure resources. It also wants to be able to automate and track provisioning and management tasks.

Exercise 1

Configure delegation of provisioning and management of Azure resources by using built-in Role-Based Access Control (RBAC) roles and built-in Azure policies.

Exercise 2

Verify delegation by provisioning Azure resources as a delegated admin and auditing provisioning events.

Estimated Time: 60 minutes

✓ If you are in a classroom, ask your instructor for the lab guide. If you are in a self-paced online course, check the Course Handouts page.