

Prerequisites and Setup

An active Azure subscription (free trial available if needed).

Contributor or Owner role access in the subscription for creating resources.

Basic familiarity with Azure Portal, PowerShell/Azure CLI (optional for some labs).

Some labs require premium features (e.g., for advanced identity or monitoring).

Labs may incur costs; clean up resources after completion.

Download supporting files if referenced in individual labs

Recommendation: Complete labs in a dedicated resource group for easy cleanup. Lab

Modules and Instructions

Lab 01: Role Based Access Control

Lab scenario

You have been asked to create a proof of concept showing how Azure users and groups are created. Also, how role-based access control is used to assign roles to groups. Specifically, you need to:

Create a Senior Admins group containing the user account of Joseph Price as its member.

Create a Junior Admins group containing the user account of Isabel Garcia as its member.

Create a Service Desk group containing the user account of Dylan Williams as its member.

Assign the Virtual Machine Contributor role to the Service Desk group.

For all the resources in this lab, we are using the East US region. Verify with your instructor this is the region to use for class.

Lab objectives

In this lab, you will complete the following exercises:

Exercise 1: Create the Senior Admins group with the user account Joseph Price as its member (the Azure portal).

Exercise 2: Create the Junior Admins group with the user account Isabel Garcia as its member (PowerShell).

Exercise 3: Create the Service Desk group with the user Dylan Williams as its member (Azure CLI).

Exercise 4: Assign the Virtual Machine Contributor role to the Service Desk group. Role-Based Access Control architecture diagram

Exercise 1

Instructions

Exercise 1: Create the Senior Admins group with the user account Joseph Price as its member.

Estimated timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Use the Azure portal to create a user account for Joseph Price.

Task 2: Use the Azure portal to create a Senior Admins group and add the user account of Joseph Price to the group.

Task 1: Use the Azure portal to create a user account for Joseph Price

In this task, you will create a user account for Joseph Price.

Start a browser session and sign-in to the Azure portal <https://portal.azure.com/>.

Note: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab and the Global Administrator role in the Microsoft Entra tenant associated with that subscription.

In the Search resources, services, and docs text box at the top of the Azure portal page, type Microsoft Entra ID and press the Enter key.

On the Overview blade of the Microsoft Entra ID tenant, in the Manage section, select Users, and then select + New user.

On the New User blade, ensure that the Create user option is selected, and specify the following settings:

SettingValue

User nameJoseph

NameJoseph Price

Click on the copy icon next to the User name to copy the full user.

Ensure that the Auto-generate password is selected, select the Show password checkbox to identify the automatically generated password. You would need to provide this password, along with the user name to Joseph.

Click Create.

Refresh the **UsersAll users** blade to verify the new user was created in your Microsoft Entra tenant.

Task2: Use the Azure portal to create a Senior Admins group and add the user account of Joseph Price to the group.

In this task, you will create the Senior Admins group, add the user account of Joseph Price to the group, and configure it as the group owner.

In the Azure portal, navigate back to the blade displaying your Microsoft Entra ID tenant.

In the Manage section, click Groups, and then select + New group.

On the New Group blade, specify the following settings (leave others with their default values):

SettingValue

Group typeSecurity

Group nameSenior Admins

Membership typeAssigned

Click the No owners selected link, on the Add owners blade, select Joseph Price, and click Select.

Click the No members selected link, on the Add members blade, select Joseph Price, and click Select.

Back on the New Group blade, click Create.

Result: You used the Azure Portal to create a user and a group, and assigned the user to the group.

Exercise 2

Exercise 2: Create a Junior Admins group containing the user account of Isabel Garcia as its member.

Estimated timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Use PowerShell to create a user account for Isabel Garcia.

Task 2: Use PowerShell to create the Junior Admins group and add the user account of Isabel Garcia to the group.

Task 1: Use PowerShell to create a user account for Isabel Garcia.

In this task, you will create a user account for Isabel Garcia by using PowerShell.

Open the Cloud Shell by clicking the Cloud Shell icon in the top-right corner of the Azure portal.

If prompted, set up Cloud Shell by creating a storage account. This is required only the first time you launch Cloud Shell.

In the Cloud Shell pane, ensure PowerShell is selected from the drop-down menu in the upperleft corner.

Note: To paste copied text into the Cloud Shell, right-click within the pane window and select Paste. Alternatively, you can use the Shift+Insert key combination.

In the PowerShell session within the Cloud Shell pane, run the following to create a password profile object:

Code

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

In the PowerShell session within the Cloud Shell pane, run the following to set the value of the password within the profile object:

Code

```
$passwordProfile.Password = "Pa55w.rd1234"
```

In the PowerShell session within the Cloud Shell pane, run the following to connect to Microsoft Entra ID:

Code

```
$passwordProfile.Password = "Pa55w.rd1234"
```

In the PowerShell session within the Cloud Shell pane,

Code

```
Connect-AzureAD
```

Code

```
$domainName = ((Get-AzureAdTenantDetail).VerifiedDomains)[0].Name
```

In the PowerShell session within the Cloud Shell pane, run the following to create a user account for Isabel Garcia:

Code

```
New-AzureADUser -DisplayName 'Isabel Garcia' -PasswordProfile $passwordProfile  
UserPrincipalName "Isabel@$domainName" -AccountEnabled $true -MailNickname 'Isabel'
```

In the PowerShell session within the Cloud Shell pane, run the following to list Microsoft Entra ID users (the accounts of Joseph and Isabel should appear on the listed):

Code

```
Get-AzureADUser -All $true | Where-Object {$_.UserPrincipalName -like "*43846135@LOD*"}  
  
Task2: Use PowerShell to create the Junior Admins group and add the user account of Isabel Garcia to the group.
```

In this task, you will create the Junior Admins group and add the user account of Isabel Garcia to the group by using PowerShell.

In the same PowerShell session within the Cloud Shell pane, run the following to create a new security group named Junior Admins:

Code

```
New-AzureADGroup -DisplayName 'Junior Admins43846135' -MailEnabled $false -  
SecurityEnabled $true -MailNickname JuniorAdmins
```

In the PowerShell session within the Cloud Shell pane, run the following to list groups in your Microsoft Entra tenant (the list should include the Senior Admins and Junior Admins groups)

Code

```
Get-AzureADGroup
```

In the PowerShell session within the Cloud Shell pane, run the following to obtain a reference to the user account of Isabel Garcia:

Code

```
$user = Get-AzureADUser -Filter "UserPrincipalName eq  
'Isabel43846135@LODSPRODMCA.onmicrosoft.com'"
```

In the PowerShell session within the Cloud Shell pane, run the following to add the user account of Isabel to the Junior Admins43846135 group:

Code

```
Add-AzADGroupMember -MemberUserPrincipalName $user.userPrincipalName -  
TargetGroupDisplayName "Junior Admins43846135"
```

In the PowerShell session within the Cloud Shell pane, run the following to verify that the Junior Admins43846135 group contains the user account of Isabel:

Code

```
Get-AzADGroupMember -GroupDisplayName "Junior Admins43846135"
```

Result: You used PowerShell to create a user and a group account, and added the user account to the group account.

Exercise 3

Exercise 3: Create a Service Desk group containing the user account of Dylan Williams as its member.

Estimated timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Use Azure CLI to create a user account for Dylan Williams.

Task 2: Use Azure CLI to create the Service Desk group and add the user account of Dylan to the group.

Task 1: Use Azure CLI to create a user account for Dylan Williams.

In this task, you will create a user account for Dylan Williams.

In the drop-down menu in the upper-left corner of the Cloud Shell pane, select Bash, and, when prompted, click Confirm.

In the Bash session within the Cloud Shell pane, run the following to identify the name of your Microsoft Entra tenant:

Code

```
DOMAINNAME=$(az ad signed-in-user show --query 'userPrincipalName' | cut -d '@' -f 2 | sed 's/\\"//')
```

In the Bash session within the Cloud Shell pane, run the following to create a user, Dylan Williams. Use yourdomain.

Code

```
az ad user create --display-name "Dylan Williams" --password "Pa55w.rd1234" --user-principalname Dylan@$DOMAINNAME
```

In the Bash session within the Cloud Shell pane, run the following to list Microsoft Entra ID user accounts (the list should include user accounts of Joseph, Isabel, and Dylan)

Code

```
az ad user list --output table
```

Task 2: Use Azure CLI to create the Service Desk group and add the user account of Dylan to the group.

In this task, you will create the Service Desk group and assign Dylan to the group.

In the same Bash session within the Cloud Shell pane, run the following to create a new security group named Service Desk.

Code `az ad group create --display-name "Service Desk" --mail-nickname`

`"ServiceDesk"`

In the Bash session within the Cloud Shell pane, run the following to list the Microsoft Entra ID groups (the list should include Service Desk, Senior Admins, and Junior Admins groups):

Code

`az ad group list -o table`

In the Bash session within the Cloud Shell pane, run the following to obtain a reference to the user account of Dylan Williams:

Code

`USER=$(az ad user list --filter "displayname eq 'Dylan Williams'")`

In the Bash session within the Cloud Shell pane, run the following to obtain the objectId property of the user account of Dylan Williams:

Code

`OBJECTID=$(echo $USER | jq '.[].id' | tr -d "")`

In the Bash session within the Cloud Shell pane, run the following to add the user account of Dylan to the Service Desk group:

```
Code az ad group member add --group "Service Desk" --member-id  
$OBJECTID
```

In the Bash session within the Cloud Shell pane, run the following to list members of the Service Desk group and verify that it includes the user account of Dylan:

```
Code az ad group member list --group "Service  
Desk"
```

Close the Cloud Shell pane.

Result: Using Azure CLI you created a user and a group accounts, and added the user account to the group.

Exercise 4

Exercise 4: Assign the Virtual Machine Contributor role to the Service Desk group.

Estimated timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Create a resource group.

Task 2: Assign the Service Desk Virtual Machine Contributor permissions to the resource group.

Task 1: Create a resource group

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Resource groups and press the Enter key.

On the Resource groups blade, click + Create and specify the following settings:

SettingValue

Subscription name the name of your Azure subscription

Resource group name AZ500Lab01

Location East US

Click Review + create and then Create.

Note: Wait for the resource group to deploy. Use the Notification icon (top right) to track progress of the deployment status.

Back on the Resource groups blade, refresh the page and verify your new resource group appears in the list of resource groups.

Task 2: Assign the Service Desk Virtual Machine Contributor permissions.

On the Resource groups blade, click the AZ500LAB01 resource group entry.

On the AZ500Lab01 blade, click Access control (IAM) in the middle pane.

On the AZ500Lab01 | Access control (IAM) blade, click + Add and then, in the drop-down menu, click Add role assignment.

On the Add role assignment blade, complete each of the following settings before clicking Next:

Note: After completing all the steps, click Next.

SettingValue

Role in the search tabVirtual Machine Contributor

Assign access to (Under Members Pane)User, group, or service principal

Select (+Select Members)Service Desk

Click Review + assign twice to create the role assignment.

From the Access control (IAM) blade, select Role assignments.

On the AZ500Lab01 | Access control (IAM) blade, on the Check access tab, in the Search by name or email address text box, type Dylan Williams.

In the list of search results, select the user account of Dylan Williams and, on the Dylan Williams assignments - AZ500Lab01 blade, view the newly created assignment.

Close the Dylan Williams assignments - AZ500Lab01 blade.

Repeat the same last two steps to check access for Joseph Price.

Result: You have assigned and checked RBAC permissions.

Clean up resources

Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs.

In the Azure portal, open the Cloud Shell by clicking the first icon in the top right of the Azure Portal.

In the drop-down menu in the upper-left corner of the Cloud Shell pane, select PowerShell, and, when prompted, click Confirm.

In the PowerShell session within the Cloud Shell pane, run the following to remove the resource group you created in this lab:

Code

```
Remove-AzResourceGroup -Name "AZ500LAB01" -Force -AsJob
```

Close the Cloud Shell pane.

Lab 02: Network Security Groups and Application Security Groups

Lab scenario

You have been asked to implement your organization's virtual networking infrastructure and test to ensure it is working correctly. In particular:

The organization has two groups of servers: Web Servers and Management Servers.

Each group of servers should be in its own Application Security Group.

You should be able to RDP into the Management Servers, but not the Web Servers.

The Web Servers should display the IIS web page when accessed from the internet.

Network security group rules should be used to control network access.

For all the resources in this lab, we are using the East US region. Verify with your instructor this is the region to use for class.

Lab objectives

In this lab, you will complete the following exercises:

Exercise 1: Create the virtual networking infrastructure

Exercise 2: Deploy virtual machines and test the network filters

Network and Application Security Groups diagram

Exercise 1

Exercise 1: Create the virtual networking infrastructure

Estimated timing: 20 minutes

For all the resources in this lab, we are using the East (US) region. Verify with your instructor this is the region to use for your class.

In this exercise, you will complete the following tasks:

Task 1: Create a virtual network with one subnet.

Task 2: Create two application security groups.

Task 3: Create a network security group and associate it with the virtual network subnet.

Task 4: Create inbound NSG security rules to all traffic to web servers and RDP to the management servers.

Task 1: Create a virtual network

In this task, you will create a virtual network to use with the network and application security groups.

Sign-in to the Azure portal <https://portal.azure.com/>.

Note: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type +++Virtual networks+++ and press the Enter key.

On the Virtual networks blade, click + Create.

On the Basics tab of the Create virtual network blade, specify the following settings (leave others with their default values) and click Next: IP Addresses:

SettingValue

SubscriptionName of the Azure subscription you are using in this lab

Resource groupUse the provided Resource Group named AZ500LAB07

Name+++myVirtualNetwork+++

RegionEast US

On the IP addresses tab of the Create virtual network blade, set the IPv4 address space to 10.0.0.0/16, and, if needed, in the Subnet name column, click default, on the Edit subnet blade, specify the following settings and click Save:

SettingValue

Subnet namedefault

Subnet address range10.0.0.0/24

Back on the IP addresses tab of the Create virtual network screen, click Review + create.

On the Review + create tab of the Create virtual network screen, click Create.

Task 2: Create application security groups

In this task, you will create an application security group.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type +++Application security groups+++ and press the Enter key.

On the Application security groups blade, click + Create.

On the Basics tab of the Create an application security group blade, specify the following settings:

SettingValue

Resource groupAZ500LAB07

Name+++myAsgWebServers+++

RegionEast US

Note: This group will be for the web servers.

Click Review + create and then click Create.

Navigate back to the Application security groups blade and click + Create.

On the Basics tab of the Create an application security group blade, specify the following settings:

SettingValue

Resource groupAZ500LAB07

Name+++myAsgMgmtServers+++

RegionEast US

Note: This group will be for the management servers.

Click Review + create and then click Create.

Task 3: Create a network security group and associate the NSG to the subnet

In this task, you will create a network security group.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type +++Network security groups+++ and press the Enter key.

On the Network security groups blade, click + Create.

On the Basics tab of the Create network security group blade, specify the following settings:

SettingValue

SubscriptionName of the Azure subscription you are using in this lab

Resource groupAZ500LAB07

Name+++myNsg+++

RegionEast US

Click Review + create and then click Create.

In the Azure portal, navigate back to the Network security groups blade and select the myNsg entry. Or select Go to resource if available.

On the myNsg blade, in the Settings section, click Subnets and then select + Associate.

On the Associate subnet blade, specify the following settings and select OK:

SettingValue

Virtual networkmyVirtualNetwork

Subnetdefault

Task 4: Create inbound NSG security rules to all traffic to web servers and RDP to the servers.

On the myNsg blade, in the Settings section, click Inbound security rules.

Review the default inbound security rules and then click + Add.

On the Add inbound security rule blade, specify the following settings to allow TCP ports 80 and 443 to the myAsgWebServers application security group (leave all other values with their default values):

SettingValue

SourceAny

Source port ranges*

Destinationin the drop-down list, select Application security group and then click myAsgWebServers

ServiceCustom

Destination port ranges80,443

ProtocolTCP

ActionAllow

Priority100

NameAllow-Web-All

Select the Add button on the Add inbound security rule page, to create the new inbound rule.

On the myNsg blade, in the Settings section, click Inbound security rules, and then click + Add.

On the Add inbound security rule blade, specify the following settings to allow the RDP port (TCP 3389) to the myAsgMgmtServers application security group (leave all other values with their default values):

SettingValue

SourceAny

Source port ranges*

Destinationin the drop-down list, select Application security group and then click myAsgMgmtServers

ServiceCustom

Destination port ranges3389

ProtocolTCP

ActionAllow

Priority110

NameAllow-RDP-All

Select Add on the Add inbound security rule page, to create the new inbound rule.

Result: You have deployed a virtual network, network security with inbound security rules, and two application security groups.

Exercise 2

Exercise 2: Deploy virtual machines and test network filters

Estimated timing: 25 minutes

In this exercise, you will complete the following tasks:

Task 1: Create a virtual machine to use as a web server.

Task 2: Create a virtual machine to use as a management server.

Task 3: Associate each virtual machines network interface to it's application security group.

Task 4: Test the network traffic filtering.

Task 1: Create a virtual machine to use as a web server

In this task, you will create a virtual machine to use as a web server.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type +++Virtual machines+++ and press the Enter key.

On the Virtual machines blade, click + Create and, in the dropdown list, click + Azure virtual machine.

On the Basics tab of the Create a virtual machine blade, specify the following settings (leave others with their default values):

SettingValue

Subscriptionthe name of the Azure subscription you will be using in this lab

Resource groupAZ500LAB07

Virtual machine namemyVmWeb

Region(US)East US

Availability optionsNo infrastructure redundancy required

Security typeStandard

ImageWindows Server 2022 Datacenter: Azure Edition- x64 Gen2

SizeStandard D2s v3

UsernameStudent

PasswordPlease create your own password and record it for future reference in subsequent labs

Confirm passwordRetype your password

Public inbound portsNone

Would you like to use an existing Windows Server LicenseNo

Note: For public inbound ports, we will rely on the precreated NSG.

Click Next: Disks > and, on the Disks tab of the Create a virtual machine blade, set the OS disk type to Standard HDD and click Next: Networking >.

On the Networking tab of the Create a virtual machine blade, select the previously created network myVirtualNetwork.

Under NIC network security group select None.

Click Next: Management >, then click Next: Monitoring >. On the Monitoring tab of the Create a virtual machine blade, verify the following setting:

SettingValue

Boot diagnosticsEnabled with managed storage account (recommended)

Click Review + create, on the Review + create blade, ensure that validation was successful and click Create.

Task 2: Create a virtual machine to use as a management server.

In this task, you will create a virtual machine to use as a management server.

In the Azure portal, navigate back to the Virtual machines blade, click + Create, and, in the dropdown list, click + Azure virtual machine.

On the Basics tab of the Create a virtual machine blade, specify the following settings (leave others with their default values):

Setting Value

Subscription the name of the Azure subscription you will be using in this lab

Resource group AZ500LAB07

Virtual machine name myVMgmt

Region (US) East US

Availability options No infrastructure redundancy required

Security type Standard

Image Windows Server 2022 Datacenter: Azure Edition - x64 Gen2

Size Standard D2s v3

Username Student

Password Please use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.

Public inbound ports None

Already have a Windows Server license Yes

Note: For public inbound ports, we will rely on the precreated NSG.

Click Next: Disks > and, on the Disks tab of the Create a virtual machine blade, set the OS disk type to Standard HDD and click Next: Networking >.

On the Networking tab of the Create a virtual machine blade, select the previously created network myVirtualNetwork.

Under NIC network security group select None.

Click Next: Management >, then click Next: Monitoring >. On the Monitoring tab of the Create a virtual machine blade, verify the following setting:

SettingValue

Boot diagnosticsEnabled with managed storage account (recommended)

Click Review + create, on the Review + create blade, ensure that validation was successful and click Create.

Note: Wait for both virtual machines to be provisioned before continuing.

Task 3: Associate each virtual machine's network interface to its application security group.

In this task, you will associate each virtual machines network interface with the corresponding application security group. The myVMWeb virtual machine interface will be associated to the myAsgWebServers ASG. The myVMMgmt virtual machine interface will be associated to the myAsgMgmtServers ASG.

In the Azure portal, navigate back to the Virtual machines blade and verify that both virtual machines are listed with the Running status.

In the list of virtual machines, click the myVMWeb entry.

On the myVMWeb blade, in the Networking section, click Network settings and then, on the myVMWeb | Networking settings blade, click the Application security groups tab.

Click + Add application security groups, in the Application security group list, select myAsgWebServers, and then click Save.

Navigate back to the Virtual machines blade and in the list of virtual machines, click the myVMMgmt entry.

On the myVMMgmt blade, in the Networking section, click Networking settings and then, on the myVMMgmt | Networking settings blade, click the Application security groups tab.

Click + Add application security groups, in the Application security group list, select myAsgMgmtServers, and then click Add.

Task 4: Test the network traffic filtering

In this task, you will test the network traffic filters. You should be able to RDP into the myVMMgmt virtual machine. You should be able to connect from the internet to the myVMWeb virtual machine and view the default IIS web page.

Navigate back to the myVMMgmt virtual machine blade.

On the myVMMgmt Overview blade, click Connect and, in the drop down menu, click Connect.

Download the RDP file and use it to connect to the myVMMgmt Azure VM via Remote Desktop. When prompted to authenticate, provide the following credentials:

SettingValue

User nameStudent

PasswordPlease use your personal password created in Lab 02 > Exercise 1 > Task 1 > Step 9.

Note: Verify that the Remote Desktop connection was successful. At this point you have confirmed you can connect via Remote Desktop to myVMMgmt.

In the Azure portal, navigate to the myVMWeb virtual machine blade.

On the myVMWeb blade, in the Operations section, click Run command and then click RunPowerShellScript.

On the Run Command Script pane, run the following to install the Web server role on myVmWeb:

Code

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

Note: Wait for the installation to complete. This might take a couple of minutes. At that point, you can verify that myVMWeb can be accessed via HTTP/HTTPS.

In the Azure portal, navigate back to the myVMWeb blade.

On the myVMWeb blade, identify the Public IP address of the myVmWeb Azure VM.

Open another browser tab and navigate to IP address you identified in the previous step.

Note: The browser page should display the default IIS welcome page because port 80 is allowed inbound from the internet based on the setting of the myAsgWebServers application security group. The network interface of the myVMWeb Azure VM is associated with that application security group.

Result: You have validated that the NSG and ASG configuration is working and traffic is being correctly managed.

Clean up resources

Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs.

Open the Cloud Shell by clicking the first icon in the top right of the Azure Portal. If prompted, select PowerShell and Create storage.

Ensure PowerShell is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

In the PowerShell session within the Cloud Shell pane, run the following to remove the resource group you created in this lab:

Code

```
Remove-AzResourceGroup -Name "AZ500LAB07" -Force -AsJob
```

Close the Cloud Shell pane.

Lab 03: Azure Firewall

Lab scenario

You have been asked to install Azure Firewall. This will help your organization control inbound and outbound network access which is an important part of an overall network security plan. Specifically, you would like to create and test the following infrastructure components:

A virtual network with a workload subnet and a jump host subnet.

A virtual machine in each subnet.

A custom route that ensures all outbound workload traffic from the workload subnet must use the firewall.

Firewall Application rules that only allow outbound traffic to www.bing.com.

Firewall Network rules that allow external DNS server lookups.

For all the resources in this lab, we are using the East US region. Verify with your instructor this is the region to use for class.

Exercise 1

Exercise 1: Deploy and test an Azure Firewall

Azure Firewall diagram

Instructions

Lab files:

\Allfiles\Labs\08\template.json

Exercise 1: Deploy and test an Azure Firewall

Estimated timing: 40 minutes

For all the resources in this lab, we are using the East (US) region. Verify with your instructor this is the region to use for your class.

In this exercise, you will complete the following tasks:

Task 1: Use a template to deploy the lab environment.

Task 2: Deploy an Azure firewall.

Task 3: Create a default route.

Task 4: Configure an application rule.

Task 5: Configure a network rule.

Task 6: Configure DNS servers.

Task 7: Test the firewall.

Task 1: Use a template to deploy the lab environment.

In this task, you will review and deploy the lab environment.

In this task, you will create a virtual machine by using an ARM template. This virtual machine will be used in the last exercise for this lab.

Sign-in to the Azure portal <https://portal.azure.com/>.

Note: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Deploy a custom template and press the Enter key.

On the Custom deployment blade, click the Build your own template in the editor option.

On the Edit template blade, click Load file, locate the \Allfiles\Labs\08\template.json file and click Open.

Note: Review the content of the template and note that it deploys an Azure VM hosting Windows Server 2016 Datacenter.

On the Edit template blade, click Save.

On the Custom deployment blade, ensure that the following settings are configured (leave any others with their default values):

SettingValue

Subscriptionthe name of the Azure subscription you will be using in this lab

Resource groupclick Create new and type the name AZ500LAB08 Location(US)

East US

adminPasswordA secure password of your own choosing for the virtual machines. Remember the password. You will need it later to connect to the VMs.

Note: To identify Azure regions where you can provision Azure VMs, refer to
<https://azure.microsoft.com/en-us/regions/offers/>

Click Review + create, and then click Create.

Note: Wait for the deployment to complete. This should take about 2 minutes.

Task 2: Deploy the Azure firewall

In this task you will deploy the Azure firewall into the virtual network.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Firewalls and press the Enter key.

On the Firewalls blade, click + Create.

On the Basics tab of the Create a firewall blade, specify the following settings:

SettingValue

Resource groupAZ500LAB08

NameTest-FW01

Region(US) East US

Firewall SKUStandard

Firewall managementUse Firewall rules (classic) to manage this firewall

Choose a virtual networkclick the Use existing option and, in the drop-down list, select Test-FWVN

Firewall Management NICTo disable this feature, deselect the Enable Firewall Management NIC option.

Public IP addressclick Add new and type the name TEST-FW-PIP and click OK

Click Review + create and then click Create.

Note: Wait for the deployment to complete. This should take about 5 minutes.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Resource groups and press the Enter key.

On the Resource groups blade, in the list of resource group, click the AZ500LAB08 entry.

Note: On the AZ500LAB08 resource group blade, review the list of resources. You can sort by Type.

In the list of resources, click the entry representing the Test-FW01 firewall.

On the Test-FW01 blade, identify the Private IP address that was assigned to the firewall.

Note: You will need this information in the next task.

Task 3: Create a default route

In this task, you will create a default route for the Workload-SN subnet. This route will configure outbound traffic through the firewall.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Route tables and press the Enter key.

On the Route tables blade, click + Create.

On the Create route table blade, specify the following settings:

SettingValue

Resource groupAZ500LAB08

RegionEast US

NameFirewall-route

Click Review + create, then click Create, and wait for the provisioning to complete.

On the Route tables blade, click Refresh, and, in the list of route tables, click the Firewall-route entry.

On the Firewall-route blade, in the Settings section, click Subnets and then, on the Firewallroute | Subnets blade, click + Associate.

On the Associate subnet blade, specify the following settings:

SettingValue

Virtual networkTest-FW-VN

SubnetWorkload-SN

Note: Ensure the Workload-SN subnet is selected for this route, otherwise the firewall won't work correctly.

Click OK to associate the firewall to the virtual network subnet.

Back on the Firewall-route blade, in the Settings section, click Routes and then click + Add.

On the Add route blade, specify the following settings:

SettingValue

Route nameFW-DG

Destination TypeIP Address

Destination IP addresses/CIDR ranges 0.0.0.0/0

Next hop type Virtual appliance

Next hop address the private IP address of the firewall that you identified in the previous task

Note: Azure Firewall is actually a managed service, but virtual appliance works in this situation.

Click Add to add the route.

Task 4: Configure an application rule

In this task you will create an application rule that allows outbound access to www.bing.com.

In the Azure portal, navigate back to the Test-FW01 firewall.

On the Test-FW01 blade, in the Settings section, click Rules (classic).

On the Test-FW01 | Rules (classic) blade, click the Application rule collection tab, and then click + Add application rule collection.

On the Add application rule collection blade, specify the following settings (leave others with their default values):

Setting Value

Name App-Coll01

Priority 200

Action Allow

On the Add application rule collection blade, create a new entry in the Target FQDNs section with the following settings (leave others with their default values):

SettingValue nameAllowGH

Source typeIP Address

Source10.0.2.0/24

Protocol porthttp:80, https:443

Target FQDNswww.bing.com

Click Add to add the Target FQDNs-based application rule.

Note: Azure Firewall includes a built-in rule collection for infrastructure FQDNs that are allowed by default. These FQDNs are specific for the platform and can 鈥橁 be used for other purposes.

Task 5: Configure a network rule

In this task, you will create a network rule that allows outbound access to two IP addresses on port 53 (DNS).

In the Azure portal, navigate back to the Test-FW01 | Rules (classic) blade.

On the Test-FW01 | Rules (classic) blade, click the Network rule collection tab and then click + Add network rule collection.

On the Add network rule collection blade, specify the following settings (leave others with their default values):

SettingValue

NameNet-Coll01

Priority200

ActionAllow

On the Add network rule collection blade, create a new entry in the IP Addresses section with the following settings (leave others with their default values):

SettingValue

NameAllowDNS

ProtocolUDP

Source typeIP address

Source Addresses10.0.2.0/24

Destination typeIP address

Destination Address209.244.0.3,209.244.0.4

Destination Ports53

Click Add to add the network rule.

Note: The destination addresses used in this case are known public DNS servers.

Task 6: Configure the virtual machine DNS servers

In this task, you will configure the primary and secondary DNS addresses for the virtual machine. This is not a firewall requirement.

In the Azure portal, navigate back to the AZ500LAB08 resource group.

On the AZ500LAB08 blade, in the list of resources, click the Srv-Work virtual machine.

On the Srv-Work blade, click Networking.

On the Srv-Work | Networking Settings blade, click the link next to the Network interface entry.

On the network interface blade, in the Settings section, click DNS servers, select the Custom option, add the two DNS servers referenced in the network rule: 209.244.0.3 and 209.244.0.4, and click Save to save the change.

Return to the Srv-Work virtual machine page.

Note: Wait for the update to complete.

Note: Updating the DNS servers for a network interface will automatically restart the virtual machine to which that interface is attached, and if applicable, any other virtual machines in the same availability set.

Task 7: Test the firewall

In this task, you will test the firewall to confirm that it works as expected.

In the Azure portal, navigate back to the AZ500LAB08 resource group.

On the AZ500LAB08 blade, in the list of resources, click the Srv-Jump virtual machine.

On the Srv-Jump blade, click Connect and, in the drop down menu, click Connect.

Download the RDP file and use it to connect to the Srv-Jump Azure VM via Remote Desktop. When prompted to authenticate, provide the following credentials:

SettingValue

User namelocaladmin

PasswordThe secure password you chose during deployment of the custom template in task 1 step 6.

Note: The following steps are performed in the Remote Desktop session to the Srv-Jump Azure VM.

Note: You will connect to the Srv-Work virtual machine. This is being done so we can test the ability to access the bing.com website.

Within the Remote Desktop session to Srv-Jump, right-click Start, in the right-click menu, click Run, and, from the Run dialog box, run the following to connect to Srv-Work.

Code mstsc

/v:Srv-Work

When prompted to authenticate, provide the following credentials:

SettingValue

User namelocaladmin

PasswordThe secure password you chose during deployment of the custom template in task 1 step 6.

Note: Wait for the Remote Desktop session to be established and the Server Manager interface to load.

Within the Remote Desktop session to Srv-Work, in Server Manager, click Local Server and then click IE Enhanced Security Configuration.

In the Internet Explorer Enhanced Security Configuration dialog box, set both options to Off and click OK.

Within the Remote Desktop session to Srv-Work, start Internet Explorer and browse to <https://www.bing.com>.

Note: The website should successfully display. The firewall allows you access.

Browse to <http://www.microsoft.com/>

Note: Within the browser page, you should receive a message with text resembling the following: HTTP request from 10.0.2.4:xxxxx to <microsoft.com:80>. Action: Deny. No rule matched. Proceeding with default action. This is expected, since the firewall blocks access to this website.

Terminate both Remote Desktop sessions.

Result: You have successfully configured and tested the Azure Firewall.

Clean up resources

Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs.

In the Azure portal, open the Cloud Shell by clicking the first icon in the top right of the Azure Portal. If prompted, click PowerShell and Create storage.

Ensure PowerShell is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

In the PowerShell session within the Cloud Shell pane, run the following to remove the resource group you created in this lab:

Code

```
Remove-AzResourceGroup -Name "AZ500LAB08" -Force -AsJob
```

Close the Cloud Shell pane.

Lab 04: Configuring and Securing ACR and AKS

Lab scenario

You have been asked to deploy a proof of concept with Azure Container Registry and Azure Kubernetes Service. Specifically, the proof of concept should demonstrate:

Using Dockerfile to build an image.

Using Azure Container Registry to store images.

Configuring an Azure Kubernetes Service.

Securing and accessing container applications both internally and externally.

For all the resources in this lab, we are using the East US region. Verify with your instructor this is the region to use for class.

Exercise 1: Configuring and Securing ACR and AKS

\Allfiles\Labs\09\nginxexternal.yaml

\Allfiles\Labs\09\nginxinternal.yaml

Exercise 1: Configuring and Securing ACR and AKS

Estimated timing: 45 minutes

For all the resources in this lab, we are using the East (US) region. Verify with your instructor this is the region to use for your class.

In this exercise, you will complete the following tasks:

Task 1: Create an Azure Container Registry

Task 2: Create a Dockerfile, build a container and push it to Azure Container Registry

Task 3: Create an Azure Kubernetes Service cluster

Task 4: Grant the AKS cluster permissions to access the ACR

Task 5: Deploy an external service to AKS

Task 6: Verify the you can access an external AKS-hosted service

Task 7: Deploy an internal service to AKS

Task 8: Verify the you can access an internal AKS-hosted service

Task 1: Create an Azure Container Registry

In this task, you will create a resource group for the lab and an Azure Container Registry.

Sign-in to the Azure portal <https://portal.azure.com/>.

Note: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab and the Global Administrator role in the Microsoft Entra tenant associated with that subscription.

In the Azure portal, open the Cloud Shell by clicking the first icon in the top right of the Azure Portal. If prompted, click Bash and Create storage.

Ensure Bash is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

In the Bash session within the Cloud Shell pane, run the following to create a new resource group for this lab:

```
Shell az group create --name AZ500LAB09 --location  
eastus
```

In the Bash session within the Cloud Shell pane, run the following to verify the resource group was created:

```
Code az group list --query "[?name=='AZ500LAB09']" -o  
table In the Bash session run the following commands  
to register the Container Registry in the lab  
environment.
```

```
Shell az provider register --namespace Microsoft.Kubernetes az  
provider register --namespace Microsoft.KubernetesConfiguration az
```

```
provider register --namespace Microsoft.OperationsManagement az  
provider register --namespace Microsoft.OperationalInsights az  
provider register --namespace Microsoft.ContainerService az  
provider register --namespace Microsoft.ContainerRegistry
```

In the Bash session within the Cloud Shell pane, run the following to create a new Azure Container Registry (ACR) instance (The name of the ACR must be globally unique):

```
Shell az acr create --resource-group AZ500LAB09 --name az500$RANDOM$RANDOM --sku
```

Basic

In the Bash session within the Cloud Shell pane, run the following to confirm that the new ACR was created:

```
Shell az acr list --resource-group
```

```
AZ500LAB09
```

Note: Record the name of the ACR. You will need it in the next task.

Task 2: Create a Dockerfile, build a container and push it to Azure Container Registry

In this task, you will create a Dockerfile, build an image from the Dockerfile, and deploy the image to the ACR.

In the Bash session within the Cloud Shell pane, run the following to create a Dockerfile to create an Nginx-based image:

```
Shell echo FROM nginx >
```

```
Dockerfile
```

In the Bash session within the Cloud Shell pane, run the following to build an image from the Dockerfile and push the image to the new ACR.

Note: The trailing period at the end of the command line is required. It designates the current directory as the location of Dockerfile.

Shell

```
ACRNAME=$(az acr list --resource-group AZ500LAB09 --query '['.{Name:name}'' --output tsv)
```

```
az acr build --resource-group AZ500LAB09 --image sample/nginx:v1 --registry $ACRNAME --file Dockerfile .
```

Note: Wait for the command to successfully complete. This might take about 2 minutes.

Close the Cloud Shell pane.

In the Azure portal, navigate to the AZ500Lab09 resource group and, in the list of resources, click the entry representing the Azure Container Registry instance you provisioned in the previous task.

On the Container registry blade, in the Services section, click Repositories.

Verify that the list of repositories includes the new container image named sample/nginx.

Click the sample/nginx entry and verify presence of the v1 tag that identifies the image version.

Click the v1 entry to view the image manifest.

Note: The manifest includes the sha256 digest, manifest creation date, and platform entries.

Task 3: Create an Azure Kubernetes Service cluster

In this task, you will create an Azure Kubernetes service and review the deployed resources.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type +++Kubernetes services+++ and press the Enter key.

On the Kubernetes services blade, click + Create and, in the drop-down menu, click + Create a Kubernetes cluster

On the Basics tab of the Create Kubernetes cluster blade, select Cluster preset configuration, select Dev/Test (\$). Now specify the following settings (leave others with their default values):

Setting Value

Subscription Use the name of the Azure subscription you are using in this lab

Resource group AZ500LAB09

Cluster preset configuration Dev/Test

Kubernetes cluster name +++MyKubernetesCluster+++

Region(US) East US

Fleet Manager None

Availability zones None

AKS pricing tier Free

Enable long term support Unchecked

Kubernetes versionUse default

Automatic upgrade*Keep the default *

Node security channel typeKeep the default

Authentication and AuthorizationLocal accounts with Kubernetes RBAC

Click Next and, on the Node Pools tab of the Create Kubernetes cluster blade, specify the following settings (leave others with their default values):

SettingValue

Enable node auto-provisioningUncheck box

Enable virtual nodesUncheck box

Other valuesKeep the defaults

Click Next, to get to Networking.

NOTE - you will get a pop-up that says a Recommendation is available for your VM Size. Accept the recommendation to move forward.

On the Networking tab of the Create Kubernetes cluster blade, specify the following settings (leave others with their default values):

SettingValue

Enable private clusterUnchecked

Set authorized IP rangesUnchecked

Network configurationAzure CNI Overlay

DNS name prefixLeave the default value

Network policyNone

Note: AKS can be configured as a private cluster. This assigns a private IP to the API server to ensure network traffic between your API server and your node pools remains on the private network only. For more information, visit [Create a private Azure Kubernetes Service cluster](#) page.

Click Next and, on the Integrations tab of the Create Kubernetes cluster page, leave All values at default.

Note: In production scenarios, you would want to enable monitoring. Monitoring is disabled in this case since it is not covered in the lab.

Click Review + Create and then click Create.

Note: Wait for the deployment to complete. This might take about 10 minutes.

Once the deployment completes, in the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Resource groups and press the Enter key.

On the Resource groups blade, in the listing of resource groups, note a new resource group named MC_AZ500LAB09_MyKubernetesCluster_eastus that holds components of the AKS Nodes. Review resources in this resource group.

Navigate back to the Resource groups blade and click the AZ500LAB09 entry.

Note: In the list of resources, note the AKS Cluster and the corresponding virtual network.

In the Azure portal, open a Bash session in the Cloud Shell.

Note: Ensure Bash is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

In the Bash session within the Cloud Shell pane, run the following to connect to the Kubernetes cluster:

```
Shell az aks get-credentials --resource-group AZ500LAB09 --name
```

```
MyKubernetesCluster
```

In the Bash session within the Cloud Shell pane, run the following to list nodes of the Kubenetus cluster:

```
Shell kubectl get
```

```
nodes
```

Note: Verify that the Status of the cluster node is listed as Ready.

Task 4: Grant the AKS cluster permissions to access the ACR and manage its virtual network

In this task, you will grant the AKS cluster permission to access the ACR and manage its virtual network.

In the Bash session within the Cloud Shell pane, run the following to configure the AKS cluster to use the Azure Container Registry instance you created earlier in this lab.

```
Shell
```

```
ACRNAME=$(az acr list --resource-group AZ500LAB09 --query '[].{Name:name}' --output tsv)
```

```
az aks update -n MyKubernetesCluster -g AZ500LAB09 --attach-acr $ACRNAME
```

Note: This command grants the `crpull` role assignment to the ACR.

Note: It may take a few minutes for this command to complete.

In the Bash session within the Cloud Shell pane, run the following to grant the AKS cluster the Contributor role to its virtual network.

Shell

```
RG_AKS=AZ500LAB09
```

```
RG_VNET=MC_AZ500LAB09_MyKubernetesCluster_eastus
```

```
AKS_VNET_NAME=aks-vnet-30198516
```

```
AKS_CLUSTER_NAME=MyKubernetesCluster
```

```
AKS_VNET_ID=$(az network vnet show --name $AKS_VNET_NAME --resource-group $RG_VNET --query id -o tsv)
```

```
AKS_MANAGED_ID=$(az aks show --name $AKS_CLUSTER_NAME --resource-group $RG_AKS --query identity.principalId -o tsv)
```

```
az role assignment create --assignee $AKS_MANAGED_ID --role "Contributor" --scope $AKS_VNET_ID
```

Task 5: Deploy an external service to AKS

In this task, you will download the Manifest files, edit the YAML file, and apply your changes to the cluster.

In the Bash session within the Cloud Shell pane, click the Upload/Download files icon, in the drop-down menu, click Upload, in the Open dialog box, navigate to the location where you downloaded the lab files, select \Allfiles\Labs\09\nginxexternal.yaml click Open. Next, select \Allfiles\Labs\09\nginxinternal.yaml, and click Open.

In the Bash session within the Cloud Shell pane, run the following to identify the name of the Azure Container Registry instance:

Shell

```
echo $ACRNAME
```

Note: Record the Azure Container Registry instance name. You will need it later in this task.

In the Bash session within the Cloud Shell pane, run the following to open the nginxexternal.yaml file, so you can edit its content.

Shell code

```
./nginxexternal.yaml
```

Note: This is the external
yaml file.

In the editor pane, scroll down to line 24 and replace the <ACRUniquename> placeholder with the ACR name.

In the editor pane, in the upper right corner, click the ellipses icon, click Save and then click Close editor.

In the Bash session within the Cloud Shell pane, run the following to apply the change to the cluster:

```
Shell kubectl apply -f
```

```
nginxexternal.yaml
```

In the Bash session within the Cloud Shell pane, review the output of the command you run in the previous task to verify that the deployment and the corresponding service have been created.

```
Code deployment.apps/nginxexternal
```

```
created service/nginxexternal created
```

Task 6: Verify the you can access an external AKS-hosted service

In this task, verify the container can be accessed externally using the public IP address.

In the Bash session within the Cloud Shell pane, run the following to retrieve information about the nginxexternal service including name, type, IP addresses, and ports.

```
Shell kubectl get service
```

```
nginxexternal
```

In the Bash session within the Cloud Shell pane, review the output and record the value in the External-IP column. You will need it in the next step.

Open a new browser tab and browse to the IP address you identified in the previous step.

Ensure the Welcome to nginx! page displays.

Task 7: Deploy an internal service to AKS

In this task, you will deploy the internal facing service on the AKS.

In the Bash session within the Cloud Shell pane, run the following to open the nginxinternal.yaml file, so you can edit its content.

Shell code

```
./nginxinternal.yaml
```

Note: This is the internal yaml file.

In the editor pane, scroll down to the line containing the reference to the container image and replace the <ACRUniquename> placeholder with the ACR name.

In the editor pane, in the upper right corner, click the ellipses icon, click Save and then click Close editor.

In the Bash session within the Cloud Shell pane, run the following to apply the change to the cluster:

Shell kubectl apply -f

```
nginxinternal.yaml
```

In the Bash session within the Cloud Shell pane, review the output to verify your deployment and the service have been created:

```
Code deployment.apps/nginxinternal
```

```
created service/nginxinternal created
```

In the Bash session within the Cloud Shell pane, run the following to retrieve information about the nginxinternal service including name, type, IP addresses, and ports.

```
Shell kubectl get service
```

```
nginxinternal
```

In the Bash session within the Cloud Shell pane, review the output. The External-IP is, in this case, a private IP address. If it is in a Pending state then run the previous command again.

Note: Record this IP address. You will need it in the next task.

Note: To access the internal service endpoint, you will connect interactively to one of the pods running in the cluster.

Note: Alternatively you could use the CLUSTER-IP address.

Task 8: Verify the you can access an internal AKS-hosted service

In this task, you will use one of the pods running on the AKS cluster to access the internal service.

In the Bash session within the Cloud Shell pane, run the following to list the pods in the default namespace on the AKS cluster:

```
Shell kubectl get
```

```
pods
```

In the listing of the pods, copy the first entry in the NAME column.

Note: This is the pod you will use in the subsequent steps.

In the Bash session within the Cloud Shell pane, run the following to connect interactively to the first pod (replace the <pod_name> placeholder with the name you copied in the previous step):

```
Shell kubectl exec -it <pod_name> --
```

```
/bin/bash
```

In the Bash session within the Cloud Shell pane, run the following to verify that the nginx web site is available via the private IP address of the service (replace the <internal_IP> placeholder with the IP address you recorded in the previous task):

```
Shell curl
```

```
http://<internal_IP>
```

Close the Cloud Shell pane.

Result: You have configured and secured ACR and AKS.

Clean up resources

Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs.

In the Azure portal, open the Cloud Shell by clicking the first icon in the top right of the Azure Portal.

In the upper-left drop-down menu of the Cloud Shell pane, select PowerShell and, when prompted, click Confirm.

In the PowerShell session within the Cloud Shell pane, run the following to remove the resource groups you created in this lab:

Code

```
Remove-AzResourceGroup -Name "AZ500LAB09" -Force -AsJob
```

Close the Cloud Shell pane.

Lab 05: Service Endpoints and Securing Storage

Lab scenario

You have been asked to create a proof of concept to demonstrate securing Azure file shares. Specifically, you want to:

Create a storage endpoint so traffic destined to Azure Storage always stays within the Azure backbone network.

Configure the storage endpoint so only resources from a specific subnet can access the storage.

Confirm that resources outside of the specific subnet cannot access the storage.

For all the resources in this lab, we are using the East US region. Verify with your instructor this is the region to use for class.

Instructions

Exercise 1: Service endpoints and security storage

Estimated timing: 45 minutes

In this exercise, you will complete the following tasks:

Task 1: Create a virtual network

Task 2: Add a subnet to the virtual network and configure a storage endpoint

Task 3: Configure a network security group to restrict access to the subnet

Task 4: Configure a network security group to allow rdp on the public subnet

Task 5: Create a storage account with a file share

Task 6: Deploy virtual machines into the designated subnets

Task 7: Test the storage connection from the private subnet to confirm that access is allowed

Task 8: Test the storage connection from the public subnet to confirm that access is denied

Task 1: Create a virtual network

In this task, you will create a virtual network.

Sign-in to the Azure portal <https://portal.azure.com/>.

Note: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Virtual networks and press the Enter key.

On the Virtual Networks blade, click + Create.

On the Basics tab of the Create virtual network blade, specify the following settings (leave others with their default values) and click Next: IP Addresses:

SettingValue

Subscriptionthe name of the Azure subscription you are using in this lab

Resource groupclick Create new and type the name AZ500LAB12

Name myVirtualNetwork

Region(US) East US

On the IP addresses tab of the Create virtual network blade, set the IPv4 address space to 10.0.0.0/16, in the Subnet name column, click default and, on the Edit subnet blade, specify the following settings and click Save:

SettingValue

Subnet namePublic

Subnet address range10.0.0.0/24

Back on the IP addresses tab of the Create virtual network blade, click Review + create.

On the Review + create tab of the Create virtual network blade, click Create.

Task 2: Add a subnet to the virtual network and configure a storage endpoint

In this task, you will create another subnet and enable a service endpoint on that subnet. Service endpoints are enabled per service, per subnet.

In the Azure portal, navigate back to the Virtual Networks blade.

On the Virtual networks blade, click the myVirtualNetwork entry.

On the myVirtualNetwork blade, in the Settings section, click Subnets.

On the myVirtualNetwork | Subnets blade, click + Subnet.

On the Add subnet blade, specify the following settings (leave others with their default values):

SettingValue

Subnet namePrivate

Subnet address range10.0.1.0/24

Service endpointsLeave the default of None

On the Add subnet blade, click Save to add the new subnet.

Note: The virtual network now has two subnets: Public and Private.

Task 3: Configure a network security group to restrict access to the subnet

In this task, you will create a network security group with two outbound security rules (Storage and internet) and one inbound security rule (RDP). You will also associate the network security group with the Private subnet. This will restrict outbound traffic from Azure VMs connected to that subnet.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Network security groups and press the Enter key.

On the Network security groups blade, click + Create.

On the Basics tab of the Create network security group blade, specify the following settings:

SettingValue

Subscriptionthe name of the Azure subscription you are using in this lab

Resource groupAZ500LAB12

Name myNsgPrivate

Region East US

Click Review + create and then click Create.

Note: In the next steps, you will create an outbound security rule that allows communication to the Azure Storage service.

In the Azure portal, navigate back to the Network security groups blade and click the myNsgPrivate entry.

On the myNsgPrivate blade, in the Settings section, click Outbound security rules.

On the myNsgPrivate | Outbound security rules blade, click + Add.

On the Add outbound security rule blade, specify the following settings to explicitly allow outbound traffic to Azure Storage (leave all other values with their default settings):

SettingValue

Source Service Tag

Source service tag VirtualNetwork

Source port ranges*

DestinationService Tag

Destination service tagStorage

Destination port ranges*

ProtocolAny

ActionAllow

Priority1000

NameAllow-Storage-All

On the Add outbound security rule blade, click Add to create the new outbound rule.

On the myNsgPrivate blade, in the Settings section, click Outbound security rules, and then click + Add.

On the Add outbound security rule blade, specify the following settings to explicitly deny outbound traffic to Internet (leave all other values with their default settings):

SettingValue

SourceService Tag

Source service tagVirtualNetwork

Source port ranges*

DestinationService Tag

Destination service tagInternet

Destination port ranges*

ProtocolAny

ActionDeny

Priority1100

NameDeny-Internet-All

Note: This rule overrides a default rule in all network security groups that allows outbound internet communication.

Note: In the next steps, you will create an inbound security rule that allows Remote Desktop Protocol (RDP) traffic to the subnet. The rule overrides a default security rule that denies all inbound traffic from the internet. Remote Desktop connections are allowed to the subnet so that connectivity can be tested in a later step.

On the myNsgPrivate blade, in the Settings section, click Inbound security rules and then click + Add.

On the Add inbound security rule blade, specify the following settings (leave all other values with their default values):

SettingValue

SourceAny

Source port ranges*

DestinationService Tag

Destination service tagVirtualNetwork

Destination port ranges3389

ProtocolTCP

ActionAllow

Priority1200

NameAllow-RDP-All

On the Add inbound security rule blade, click Add to create the new inbound rule.

Note: Now you will associate the network security group with the Private subnet.

On the Subnets blade, select + Associate and specify the following settings in the Associate subnet section and then click OK:

SettingValue

Virtual networkmyVirtualNetwork

SubnetPrivate

Task 4: Configure a network security group to allow rdp on the public subnet

In this task, you will create a network security group with one inbound security rule (RDP). You will also associate the network security group with the Public subnet. This will allow RDP access to the Public VM.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Network security groups and press the Enter key.

On the Network security groups blade, click + Create.

On the Basics tab of the Create network security group blade, specify the following settings:

SettingValue

Subscriptionthe name of the Azure subscription you are using in this lab

Resource groupAZ500LAB12

Name myNsgPublic

Region East US

Click Review + create and then click Create.

Note: In the next steps, you will create an outbound security rule that allows communication to the Azure Storage service.

In the Azure portal, navigate back to the Network security groups blade and click the myNsgPublic entry.

On the myNsgPublic blade, in the Settings section, click Inbound security rules and then click + Add.

On the Add inbound security rule blade, specify the following settings (leave all other values with their default values):

SettingValue

SourceAny

Source port ranges*

DestinationService Tag

Destination service tagVirtualNetwork

Destination port ranges3389

ProtocolTCP

ActionAllow

Priority1200

NameAllow-RDP-All

On the Add inbound security rule blade, click Add to create the new inbound rule.

Note: Now you will associate the network security group with the Public subnet.

On the Subnets blade, select + Associate and specify the following settings in the Associate subnet section and then click OK:

SettingValue

Virtual networkmyVirtualNetwork

SubnetPublic

Task 5: Create a storage account with a file share

In this task, you will create a storage account with a file share and obtain the storage account key.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Storage accounts and press the Enter key.

On the Storage accounts blade, click + Create.

On the Basics tab of the Create storage account blade, specify the following settings (leave others with their default values):

SettingValue

Subscriptionthe name of the Azure subscription you are using in this lab

Resource groupAZ500LAB12

Storage account nameany globally unique name between 3 and 24 in length consisting of letters and digits

Location(US) EastUS

PerformanceStandard (general-purpose v2 account)

RedundencyLocally redundant storage (LRS)

On the Basics tab of the Create storage account blade, click Review + Create, wait for the validation process to complete, and click Create.

Note: Wait for the Storage account to be created. This should take about 2 minutes.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Resource groups and press the Enter key.

On the Resource groups blade, in the list of resource group, click the AZ500LAB12 entry.

On the AZ500LAB12 resource group blade, in the list of resources, click the entry representing the newly created storage account.

On the storage account Overview blade, click File Shares under the Data storage tab, and then click + File Share.

On the New file share blade, untick the Enable backup option in the backup tab.

On the New file share blade, specify the following settings:

SettingValue

Name my-file-share

On the New file share blade, click Create.

Note: Now, retrieve and record the PowerShell script that creates a drive mapping to the Azure file share.

On the storage account blade, in the list of file shares, click my-file-share.

On the my-file-share blade, click Connect.

On the Connect blade, on the Windows tab, copy the PowerShell script that creates a Z drive mapping to the file share.

Note: Record this script. You will need this in a later in this lab in order to map the file share from the Azure virtual machine on the Private subnet.

Navigate back to the storage account blade, then in the Security + networking section, click Networking.

Under Public network access select Manage and as Default action select Enable from selected networks.

Under Resource settings: Virtual networks, IP addresses, and exceptions blade, select view and click the + Add existing virtual network link.

On the Add networks blade, specify the following settings:

Setting Value

Subscription the name of the Azure subscription you are using in this lab

Virtual networks myVirtualNetwork

SubnetsPrivate

On the Add networks blade, click Add.

Back on the storage account blade, click Save.

Note: At this point in the lab you have configured a virtual network, a network security group, and a storage account with a file share.

Task 6: Deploy virtual machines into the designated subnets

In this task, you will create two virtual machines one in the Private subnet and one in the Public subnet.

Note: The first virtual machine will be connected to the Private subnet.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Virtual machines and press the Enter key.

On the Virtual machines blade, click + Create and, in the dropdown list, click + Azure Virtual machine

On the Basics tab of the Create a virtual machine blade, specify the following settings (leave others with their default values):

SettingValue

Subscriptionthe name of the Azure subscription you will be using in this lab

Resource groupAZ500LAB12

Virtual machine namemyVmPrivate

Region(US)East US

ImageWindows Server 2022 Datacenter: Azure Edition - Gen 2

UsernameStudent

PasswordPlease use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.

Public inbound portsNone

Already have a Windows Server licenseNot selected

Note: For public inbound ports, we will rely on the precreated NSG.

Click Next: Disks > and, on the Disks tab of the Create a virtual machine blade, set the OS disk type to Standard HDD and click Next: Networking >.

Click Next: Networking >, on the Networking tab of the Create a virtual machine blade, specify the following settings (leave others with their default values):

SettingValue

Virtual networkmyVirtualNetwork

SubnetPrivate (10.0.1.0/24)

Public IP(new)myVmPrivate-ip

NIC network security groupNone

Click Next: Management >, on the Management tab of the Create a virtual machine blade, accept the default settings and click Review + create.

On the Review + create blade, ensure that validation was successful and click Create.

Note: The second virtual machine will be connected to the Public subnet.

On the Virtual machines blade, click + Add and, in the dropdown list, click + Azure Virtual machine.

On the Basics tab of the Create a virtual machine blade, specify the following settings (leave others with their default values):

SettingValue

Subscriptionthe name of the Azure subscription you will be using in this lab

Resource groupAZ500LAB12

Virtual machine namemyVmPublic

Region(US)East US

ImageWindows Server 2022 Datacenter: Azure Edition - Gen 2

UsernameStudent

PasswordPlease use your personal password created in Lab 02 > Exercise 1 > Task 1 > Step 9.

Public inbound portsNone

Already have a Windows Server licenseNot selected

Note: For public inbound ports, we will rely on the precreated NSG.

Click Next: Disks > and, on the Disks tab of the Create a virtual machine blade, set the OS disk type to Standard HDD and click Next: Networking >.

Click Next: Networking >, on the Networking tab of the Create a virtual machine blade, specify the following settings (leave others with their default values):

SettingValue

Virtual networkmyVirtualNetwork

SubnetPublic (10.0.0.0/24)

Public IP(new)myVmPublic-ip

NIC network security groupNone

Click Next: Management >, on the Management tab of the Create a virtual machine blade, accept the default settings and click Review + create.

On the Review + create blade, ensure that validation was successful and click Create.

Note: You can continue to the next task once the deployment of the myVMPublic Azure VM is completed.

Task 7: Test the storage connection from the private subnet to confirm that access is allowed

In this task, you will connect to the myVMPublic virtual machine via Remote Desktop and map a drive to the file share.

Navigate back to the Virtual machines blade.

On the Virtual machines blade, click the myVMPublic entry.

On the myVMPublic blade, click Connect and, in the drop down menu, click Connect.

Download the RDP file and use it to connect to the myVMPublic Azure VM via Remote Desktop. When prompted to authenticate, provide the following credentials:

SettingValue

UsernameStudent

PasswordPlease use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.

Note: Wait for the Remote Desktop session to open and Server Manager to load.

Note: You will now map drive Z to an Azure File share within the Remote Desktop session to a Windows Server 2022 computer

Within the Remote Desktop session to myVMPrivate, click Start and then click Windows PowerShell ISE.

Within the Windows PowerShell ISE window, open the Script pane, then paste and run the PowerShell script that you recorded earlier in this lab. The script has the following format:

Code

```
$connectTestResult = Test-NetConnection -ComputerName  
<storage_account_name>.file.core.windows.net -Port 445  
if ($connectTestResult.TcpTestSucceeded) {  
  
    # Save the password so the drive will persist on reboot  
  
    cmd.exe /C "cmdkey /add:<storage_account_name>.file.core.windows.net"  
    /user:"localhost\<storage_account_name>" /pass:<storage_account_key>"  
  
    # Mount the drive  
  
    New-PSDrive -Name Z -PSProvider FileSystem -Root  
    "\\\<storage_account_name>.file.core.windows.net\my-file-share" -Persist  
  
} else {  
  
    Write-Error -Message "Unable to reach the Azure storage account via port 445. Check to make  
sure your organization or ISP is not blocking port 445, or use Azure P2S VPN, Azure S2S VPN, or  
Express Route to tunnel SMB traffic over a different port."  
  
}
```

Note: The <storage_account_name> placeholder represents the name of the storage account hosting the file share and <storage_account_key> one its primary key

Start File Explorer and verify that the Z: drive mapping has been successfully created.

Next, from the console pane of the Windows PowerShell ISE console, run the following to verify that the virtual machine has no outbound connectivity to the internet:

Code

```
Test-NetConnection -ComputerName www.bing.com -Port 80
```

Note: The test will fail because the network security group associated with the Private subnet does not allow outbound access to the internet.

Terminate the Remote Desktop session to the myVMPrivate Azure VM.

Note: At this point, you have confirmed that the virtual machine in the Private subnet can access the storage account.

Task 8: Test the storage connection from the public subnet to confirm that access is denied

Navigate back to the Virtual machines blade.

On the Virtual machines blade, click the myVMPublic entry.

On the myVMPublic blade, click Connect and, in the drop down menu, click Connect.

Click Connect via RDP and use it to connect to the myVMPublic Azure VM via Remote Desktop. When prompted to authenticate, provide the following credentials:

SettingValue

UsernameStudent

PasswordPlease use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.

Note: Wait for the Remote Desktop session to open and Server Manager to load.

Note: You will now map drive Z to an Azure File share within the Remote Desktop session to a Windows Server 2022 computer

Within the Remote Desktop session to myVMPublic, click Start and then click Windows PowerShell ISE.

Within the Windows PowerShell ISE window, open the Script pane, then paste and run the same PowerShell script that you ran within the Remote Desktop session to the myVMPublic Azure VM.

Note: This time, you will receive the New-PSDrive : Access is denied error.

Note: Access is denied because the myVmPublic virtual machine is deployed in the Public subnet. The Public subnet does not have a service endpoint enabled for the Azure Storage. The storage account only allows network access from the Private subnet.

Next, from the console pane of the Windows PowerShell ISE console, run the following to verify that the virtual machine has outbound connectivity to the internet:

Code

```
Test-NetConnection -ComputerName www.bing.com -Port 80
```

Note: The test will succeed because there is no outbound security rule to deny internet on the Public subnet.

Terminate the Remote Desktop session to the myVMPublic Azure VM.

Note: At this point, you have confirmed that the virtual machine in the Public subnet cannot access the storage account, but has access to the internet.

Clean up resources

Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs.

Lab 06: Create a Log Analytics Workspace, Azure Storage Account, and Data Collection Rule (DCR)

Student lab manual

Lab scenario

As an Azure Security Engineer for a financial technology company, you are tasked with enhancing monitoring and security visibility across all Azure virtual machines (VMs) used for processing financial transactions and managing sensitive customer data. The security team requires detailed logs and performance metrics from these VMs to detect potential threats and optimize system performance. The Chief Information Security Officer (CISO) has asked you to implement a solution that collects security events, system logs, and performance counters. You have been assigned to configure the Azure Monitor Agent (AMA) along with Data Collection Rules (DCRs) to centralize log collection and performance monitoring.

For all the resources in this lab, we are using the East US region. Verify with your instructor this is the region to use for class.

Lab objectives

In this lab, you will complete the following exercises:

Exercise 1: Deploy an Azure virtual machine

Exercise 2: Create a Log Analytics workspace

Exercise 3: Create an Azure storage account

Exercise 4: Create a data collection rule

Exercise 1: Deploy an Azure virtual machine

Exercise 1: Deploy an Azure virtual machine

Exercise timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Deploy an Azure virtual machine

Sign-in to the Azure portal <https://portal.azure.com/>.

Note: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab.

Open the Cloud Shell by clicking the first icon in the top right of the Azure Portal. If prompted, select PowerShell.

Ensure PowerShell is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

In the Getting started window, leave the default setting as is: Select a subscription to get started. You can optionally mount a storage account to persist files between sessions. No storage account required.

From the Subscription drop-down menu, select your lodsbscription.

Leave Use an existing private virtual network unchecked, then click Apply.

In the PowerShell session within the Cloud Shell pane, run the following to create a resource group that will be used in this lab:

Code

```
New-AzResourceGroup -Name AZ500LAB131415 -Location 'EastUS'
```

Note: This resource group will be used for labs 8, 9, and 10.

In the PowerShell session within the Cloud Shell pane, run the following to enable encryption at host (EAH)

Code

```
Register-AzProviderFeature -FeatureName "EncryptionAtHost" -ProviderNamespace Microsoft.Compute
```

In the PowerShell session within the Cloud Shell pane, run the following to create a new Azure virtual machine.

Code

```
New-AzVm -ResourceGroupName "AZ500LAB131415" -Name "myVM" -Location 'EastUS' -VirtualNetworkName "myVnet" -SubnetName "mySubnet" -SecurityGroupName "myNetworkSecurityGroup" -PublicIpAddressName "myPublicIpAddress" -PublicIpSku Standard -OpenPorts 80,3389 -Size Standard_D2_v4
```

When prompted for credentials:

SettingValue

Userlocaladmin

PasswordPlease use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.

Note: Wait for the deployment to complete.

In the PowerShell session within the Cloud Shell pane, run the following to confirm that the virtual machine named myVM was created and its ProvisioningState is Succeeded.

Code

```
Get-AzVM -Name 'myVM' -ResourceGroupName 'AZ500LAB131415' | Format-Table Close  
the Cloud Shell pane.
```

Exercise 2: Create a Log Analytics workspace

Exercise 2: Create an Log Analytics workspace

Exercise timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Create a Log Analytics workspace

In this task, you will create a Log Analytics workspace.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Log Analytics workspaces and press the Enter key.

On the Log Analytics workspaces blade, click + Create.

On the Basics tab of the Create Log Analytics workspace blade, specify the following settings (leave others with their default values):

SettingValue

Subscriptionthe name of the Azure subscription you are using in this lab

Resource groupAZ500LAB131415

NameIgwnite

RegionEast US

Select Review + create.

On the Review + create tab of the Create Log Analytics workspace blade, select Create.

Exercise 3: Create an Azure storage account

Exercise 3: Create an Azure storage account

Estimated timing: 10 minutes

In this exercise, you will complete the following tasks:

Task 1: Create an Azure storage account

In this task, you will create a storage account.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Storage accounts and press the Enter key.

On the Storage accounts blade in the Azure portal, click the + Create button to create a new storage account.

On the Basics tab of the Create storage account blade, specify the following settings (leave others with their default values):

Setting	Value
Subscription	the name of the Azure subscription you are using in this lab
Resource group	AZ500LAB131415
Instance details	
Storage account name	strgactignite
Region	(US) EastUS
Primary service	Azure Blob Storage or Azure Data Lake Storage Gen 2
Performance	Standard (general-purpose v2 account)
Redundancy	Locally redundant storage (LRS)

On the Basics tab of the Create storage account blade, click Review + create. After the validation process completes, click Create.

Note: Wait for the Storage account to be created. This should take about 2 minutes.

Exercise 4: Create a data collection rule

Exercise 4: Create a Data Collection Rule

Estimated timing: 15 minutes

In this exercise, you will complete the following tasks:

Task 1: Create a Data Collection Rule.

In this task, you will create a data collection rule.

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Monitor and press the Enter key.

On the Monitor Settings blade, click Data Collection Rules.

Click the + Create button to create a new data collection rule.

On the Basics tab of the Create Data Collection Rule blade, specify the following settings:

Setting	Value
Rule details	Rule Name DCR1 Subscription the name of the Azure subscription you are using in this lab Resource Group AZ500LAB131415 Region East US Platform Type Windows Data Collection Endpoint Leave Blank

Click on the button labeled Next: Resources > to proceed.

On the Resources page, select + Add resources.

In the Select a scope template, check the Subscription box in the Scope.

At the bottom of the Select a scope template, click Apply.

At the bottom of the Resources page, select Next: Collect and deliver >.

Click + Add data source, then on the Add data source page, change the Data source type dropdown menu to display Performance Counters. Leave the following default settings:

SettingValue

Performance counterSample rate (seconds)

CPU60

Memory60

Disk60

Network60

Click on the button labeled Next: Destination > to proceed.

Click + Add destination, change the Destination type drop-down menu to display Azure Monitor Logs. In the Subscription window, ensure that your Subscription is displayed, then change the Account or namespace drop-down menu to reflect your previously created Log Analytics Workspace.

Click on Add data source at the bottom of the page.

Click Review + create.

Click Create.

Results: You deployed an Azure virtual machine, Log Analytics workspace, Azure storage account, and a data collection rule to collect events and performance counters from virtual machines with Azure Monitor Agent.

Note: Do not remove the resources from this lab, as they are needed for the Microsoft Defender for Cloud lab, the 'Enable just-in-time access on VMs' lab, and the Microsoft Sentinel lab

Lab 07: Configuring Microsoft Defender for Cloud Enhanced Security Features for Servers Enable advanced protection for servers in Defender for Cloud.

Lab scenario

As an Azure Security Engineer for a global e-commerce company, you are responsible for securing the company's cloud infrastructure. The organization relies heavily on Azure virtual machines (VMs) and on-premises servers to run critical applications, manage customer data, and process transactions. The Chief Information Security Officer (CISO) has identified the need for enhanced security measures to protect these resources against cyber threats, vulnerabilities, and misconfigurations. You have been tasked with enabling Microsoft Defender for Servers in Microsoft Defender for Cloud to provide advanced threat protection and security monitoring for both Azure VMs and hybrid servers.

Lab objectives

Configure Microsoft Defender for Cloud Enhanced Security Features for Servers

Review the enhanced security features for Microsoft Defender for Servers Plan 2

Exercise instructions

Configure Microsoft Defender for Cloud Enhanced Security Features for Servers

In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Microsoft Defender for Cloud and press the Enter key.

On the Microsoft Defender for Cloud, Management blade, go to the Environment settings. Expand the environment settings folders until the subscription section is displayed, then click the subscription to view details.

In the Settings blade, under Defender plans, expand Cloud Workload Protection (CWP).

From the Cloud Workload Protection (CWP) Plan list, select Servers. On the right side of the page, change the Status from Off to On, then click Save.

To review the details of Microsoft Defender for Servers Plan 2, select Change plan >.

Note: Enabling the Cloud Workload Protection (CWP) Servers plan from Off to On enables Microsoft Defender for Servers Plan 2.

Results: You have enabled Microsoft Defender for Servers Plan 2 on your subscription.

Lab 08: Enable Just-in-Time Access on VMs Implement JIT VM access to reduce attack surface.

Lab scenario

As an Azure Security Engineer at a financial services company, you're responsible for securing Azure resources, including virtual machines (VMs) that host critical applications. The security team has identified that continuous open access to VMs increases the risk of brute-force attacks and unauthorized access. To mitigate this, the Chief Information Security Officer (CISO) has requested that you enable Just-in-Time (JIT) VM access on a specific Azure VM used for processing financial transactions.

Lab objectives

In this lab, you will complete the following exercises:

Exercise 1: Enable JIT on your VMs from the Azure portal.

Exercise 2: Request access to a VM that has JIT enabled from the Azure portal.

Exercise 1: Enable JIT on your VMs from the Azure portal

Note: You can enable JIT on a VM from the Azure virtual machines pages of the Azure portal.

In the search box at the top of the portal, enter virtual machines. Select Virtual machines in the search results.

Select myVM.

Select Configuration from the Settings section of myVM.

Under Just-in-time VM access, select Enable just-in-time.

Under Just-in-time VM access, click on the link that reads Open Microsoft Defender for Cloud.

By default, just-in-time access for the VM uses these settings:

Windows machines

RDP port: 3389

Maximum allowed access: Three hours

Allowed source IP addresses: Any

Linux machines

SSH port: 22

Maximum allowed access: Three hours

Allowed source IP addresses: Any

By default, just-in-time access for the VM uses these settings:

From the Configured tab, right-click on the VM to which you want to add a port, and select edit.

Under JIT VM access configuration, you can either edit the existing settings of an already protected port or add a new custom port.

When you've finished editing the ports, select Save.

Exercise 2: Request access to a VM that has JIT enabled from the Azure portal

Exercise 2: Request access to a JIT-enabled VM from the Azure virtual machine's connect page.

Note: When a VM has a JIT enabled, you have to request access to connect to it. You can request access in any of the supported ways, regardless of how you enabled JIT.

In the Azure portal, open the virtual machines pages.

Select the VM to which you want to connect, and open the Connect page.

Azure checks to see if JIT is enabled on that VM.

If JIT isn't enabled for the VM, you're prompted to enable it.

If JIT is enabled, select Request access to pass an access request with the requesting IP, time range, and ports that were configured for that VM.

Results: You have explored various methods on how to enable JIT on your VMs and how to request access to VMs that have JIT enabled in Microsoft Defender for Cloud.

Lab 09: Microsoft Sentinel Deploy and configure Microsoft Sentinel for SIEM and SOAR capabilities.

Lab scenario

Note: **Azure Sentinel** is renamed to **Microsoft Sentinel**

You have been asked to create a proof of concept of Microsoft Sentinel-based threat detection and response. Specifically, you want to:

- Start collecting data from Azure Activity and Microsoft Defender for Cloud. - Add built in and custom alerts
- Review how Playbooks can be used to automate a response to an incident.

> For all the resources in this lab, we are using the **East US** region. Verify with your instructor this is the region to use for class.

Lab objectives

In this lab, you will complete the following exercise:

- Exercise 1: Implement Microsoft Sentinel

Microsoft Sentinel diagram

![Diagram of the process flow of tasks for this lab.](./media/microsoft-sentinel-diagram.png)

Instructions

Lab files:

- **\\Allfiles\\Labs\\15\\changeincidentseverity.json**

Exercise 1: Implement Microsoft Sentinel

Estimated timing: 30 minutes

In this exercise, you will complete the following tasks:

- Task 1: On-board Microsoft Sentinel
- Task 2: Connect Azure Activity to Sentinel
- Task 3: Create a rule that uses the Azure Activity data connector.
- Task 4: Create a playbook
- Task 5: Create a custom alert and configure the playbook as an automated response.
- Task 6: Invoke an incident and review the associated actions.

Task 1: On-board Microsoft Sentinel

In this task, you will on-board Microsoft Sentinel and connect the Log Analytics workspace.

1. Sign-in to the Azure portal **`https://portal.azure.com/`**.

>**Note**: Sign in to the Azure portal using an account that has the Owner or Contributor role in the Azure subscription you are using for this lab.

2. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Microsoft Sentinel** and press the **Enter** key.

Note: If this is your first attempt to action Microsoft Sentinel in the Azure dashboard complete the following step(s): In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Microsoft Sentinel** and press the **Enter** key. Select **Microsoft Sentinel** from the **Services** view.

3. On the **Microsoft Sentinel** blade, click **+ Create**.

4. On the **Add Microsoft Sentinel to a workspace** blade, select the Log Analytics workspace you created in the Azure Monitor lab and click **Add**.

>**Note**: Microsoft Sentinel has very specific requirements for workspaces. For example, workspaces created by Microsoft Defender for Cloud can not be used. Read more at [Quickstart: On-board Microsoft Sentinel](<https://docs.microsoft.com/en-us/microsoft/sentinel/quickstartonboard>)

Task 2: Configure Microsoft Sentinel to use the Azure Activity data connector.

In this task, you will configure Sentinel to use the Azure Activity data connector.

1. In the Azure portal, on the **Microsoft Sentinel \| Overview** blade, in the **Content management** section, click **Content hub**.
2. On the **Microsoft Sentinel \| Content hub** blade, review the list of available content.

3. Type **Azure** into the search bar and select the entry representing **Azure Activity**. Review its description at the far right, and then click **Install**.
 4. Wait for the **Install Success** notification. In the left navigation panel, in the **Configuration** section, click **Data connectors**.
 5. On the **Microsoft Sentinel \| Data connectors** blade, click **Refresh** and review the list of available connectors. Select the entry representing the **Azure Activity** connector (hide the menu bar on the left using \<> if needed), review its description and status at the far right, and then click **Open connector page**.
 6. On the **Azure Activity** blade the **Instructions** tab should be selected, note the **Prerequisites** and scroll down to the **Configuration**. Take note of the information describing the connector update. Your subscription never used the legacy connection method so you can skip step 1 (the **Disconnect All** button will be grayed out) and proceed to step 2.
 7. In step 2 **Connect your subscriptions through diagnostic settings new pipeline**, review the "Launch the Azure Policy Assignment wizard and follow the steps" instructions then click **Launch the Azure Policy Assignment wizard\>**.
 8. On the **Configure Azure Activity logs to stream to specified Log Analytics workspace** (Assign Policy page) **Basics** tab, click the **Scope ellipsis (...)** button. In the **Scope** page choose your subscription from the drop-down subscription list and click the **Select** button at the bottom of the page.
- >**Note**: *Do not* choose a Resource Group
9. Click the **Next** button at the bottom of the **Basics** tab twice to proceed to the **Parameters** tab. On the **Parameters** tab click the **Primary Log Analytics workspace ellipsis (...)** button. In the **Primary Log Analytics workspace** page, make sure your

subscription is selected and use the **workspaces** drop-down to select the Log Analytics workspace you are using for Sentinel. When done click the **Select** button at the bottom of the page.

10. Click the **Next** button at the bottom of the **Parameters** tab to proceed to the **Remediation** tab. On the **Remediation** tab select the **Create a remediation task** checkbox. This will enable the "Configure Azure Activity logs to stream to specified Log Analytics workspace" in the **Policy to remediate** drop-down. In the **System assigned identity location** drop-down, select the region (East US for example) you selected earlier for your Log Analytics workspace.
11. Click the **Next** button at the bottom of the **Remediation** tab to proceed to the **Non-compliance message** tab. Enter a Non-compliance message if you wish (this is optional) and click the **Review + Create** button at the bottom of the **Non-compliance message** tab.
12. Click the **Create** button. You should observe three succeeded status messages: **Creating policy assignment succeeded, Role Assignments creation succeeded, and Remediation task creation succeeded**.

>**Note**: You can check the Notifications, bell icon to verify the three successful tasks.

13. Verify that the **Azure Activity** pane displays the **Data received** graph (you might have to refresh the browser page).

>**Note**: It may take over 15 minutes before the Status shows "Connected" and the graph displays Data received.

Task 3: Create a rule that uses the Azure Activity data connector.

In this task, you will review and create a rule that uses the Azure Activity data connector.

1. On the **Microsoft Sentinel \| Configuration** blade, click **Analytics**.

2. On the **Microsoft Sentinel \| Analytics** blade, click the **Rule templates** tab.

>**Note**: Review the types of rules you can create. Each rule is associated with a specific Data Source.

3. In the listing of rule templates, type **Suspicious** into the search bar form and click the **Suspicious number of resource creation or deployment** entry associated with the **Azure Activity** data source. And then, in the pane displaying the rule template properties, click **Create rule** (scroll to the right of the page if needed).

>**Note**: This rule has medium severity.

4. On the **General** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, accept the default settings and click **Next: Set rule logic >**.

5. On the **Set rule logic** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, accept the default settings and click **Next: Incident settings (Preview) >**.

6. On the **Incident settings** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, accept the default settings and click **Next: Automated response >**.

>**Note**: This is where you can add a playbook, implemented as a Logic App, to a rule to automate the remediation of an issue.

7. On the **Automated response** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, accept the default settings and click **Next: Review and create >**.
8. On the **Review and create** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, click **Save**.

>**Note**: You now have an active rule.

Task 4: Create a playbook

In this task, you will create a playbook. A security playbook is a collection of tasks that can be invoked by Microsoft Sentinel in response to an alert.

1. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Deploy a custom template** and press the **Enter** key.
2. On the **Custom deployment** blade, click the **Build your own template in the editor** option.
3. On the **Edit template** blade, click **Load file**, locate the **\\Allfiles\\Labs\\15\\changeincidentseverity.json** file and click **Open**.

>**Note**: You can find sample playbooks at
[<https://github.com/Azure/AzureSentinel/tree/master/Playbooks>](https://github.com/Azure/AzureSentinel/tree/master/Playbooks).

4. On the **Edit template** blade, click **Save**.

5. On the **Custom deployment** blade, ensure that the following settings are configured (leave any others with their default values):

Setting	Value
---------	-------

---	---
-----	-----

Subscription	the name of the Azure subscription you are using in this lab
--------------	--

Resource group	**AZ500LAB131415**
----------------	--------------------

Location	**(US) East US**
----------	------------------

Playbook Name	**Change-Incident-Severity**
---------------	------------------------------

User Name	your email address
-----------	--------------------

6. Click **Review + create** and then click **Create**.

>**Note**: Wait for the deployment to complete.

7. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Resource groups** and press the **Enter** key.
8. On the **Resource groups** blade, in the list of resource group, click the **AZ500LAB131415** entry.
9. On the **AZ500LAB131415** resource group blade, in the list of resources, click the entry representing the newly created **Change-Incident-Severity** logic app.
10. On the **Change-Incident-Severity** blade, click **Edit**.

>**Note**: On the **Logic Apps Designer** blade, each of the four s displays a warning. This means that each needs to reviewed and configured.

11. On the **Logic Apps Designer** blade, click the first **s** step.
12. Click **Add new**, ensure that the entry in the **Tenant** drop down list contains your Azure AD tenant name and click **Sign-in**.
13. When prompted, sign in with the user account that has the Owner or Contributor role in the Azure subscription you are using for this lab.
14. Click the second **s** step and, in the list of s, select the second entry, representing the you created in the previous step.
15. Repeat the previous step for the remaining two **s** steps.

>**Note**: Ensure there are no warnings displayed on any of the steps.

16. On the **Logic Apps Designer** blade, click **Save** to save your changes.

Task 5: Create a custom alert and configure a playbook as an automated response

1. In the Azure portal, navigate back to the **Microsoft Sentinel \| Overview** blade.

2. On the the **Microsoft Sentinel \| Overview** blade, in the **Configuration** section, click **Analytics**.
3. On the **Microsoft Sentinel \| Analytics** blade, click **+ Create** and, in the drop-down menu, click **Scheduled query rule**.
4. On the **General** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, specify the following settings (leave others with their default values):

Setting	Value
Name	Playbook Demo
Tactics	Initial Access

5. Click **Next: Set rule logic >**.

6. On the **Set rule logic** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, in the **Rule query** text box, paste the following rule query.

```

AzureActivity

```
| where ResourceProviderValue =~ "Microsoft.Security"
| where OperationNameValue =~
"Microsoft.Security/locations/jitNetworkAccessPolicies/delete"
```
```

>**Note**: This rule identifies removal of Just-in-time VM access policies.

>**Note** if you receive a parse error, intellisense may have added values to your query. Ensure the query matches otherwise paste the query into notepad and then from notepad to the rule query.

7. On the **Set rule logic** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, in the **Query scheduling** section, set the **Run query every** to **5 Minutes**.
8. On the **Set rule logic** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, accept the default values of the remaining settings and click **Next: Incident settings >**.
9. On the **Incident settings** tab of the **Analytics rule wizard - Create a new Scheduled rule** blade, accept the default settings and click **Next: Automated response >**.
10. On the **Automated response** tab of the **Analytic rule wizard - Create a new Scheduled rule** blade, under **Automation rules**, click **+ Add new**.
11. In the **Create new automation rule** window, enter **Run Change-Severity Playbook** for the **Automation rule name**; under the **Trigger** field, click the drop-down menu and select **When alert is created**.
12. In the **Create new automation rule** window, under **Actions**, read the note and then click **Manage playbook permissions**. On the **Manage permissions** window, select the checkbox next to the previously created **Resource group AZ500LAB1314151** and then click **Apply**.

13. In the **Create new automation rule** window, under **Actions**, click the second dropdown menu and select the **Change-Incident-Severity** logic app. On the **Create new automation rule** window, click **Apply**.

14. On the **Automated response** tab of the **Analytic rule wizard - Create a new Scheduled rule** blade, click **Next: Review and create >** and click **Save**

>**Note**: You now have a new active rule called **Playbook Demo**. If an event identified by the rule logic occurs, it will result in a medium severity alert, which will generate a corresponding incident.

Task 6: Invoke an incident and review the associated actions.

1. In the Azure portal, navigate to the **Microsoft Defender for Cloud \| Overview** blade.

>**Note**: Check your secure score. By now it should have updated.

2. On the **Microsoft Defender for Cloud \| Overview** blade, click **Workload protections** under **Cloud Security** in the left navigation.

3. On the **Microsoft Defender for Cloud \| Workload protections** blade, scroll down and click **Just-in-time VM access** tile under **Advanced protection**.

4. On the **Just-in-time VM access** blade, on the right hand side of the row referencing the **myVM** virtual machine, click the **ellipsis (...)** button, click **Remove** and then click **Yes**.

>**Note:** If the VM is not listed in the **Just-in-time VMs**, navigate to **Virtual Machine** blade and click the **Configuration**, Click the **Enable the Just-in-time VMs** option under the **Just-in-time Vm's access**. Repeat the above step to navigate back to the **Microsoft Defender for Cloud** and refresh the page, the VM will appear.

5. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Activity log** and press the **Enter** key.

6. Navigate to the **Activity log** blade, note an **Delete JIT Network Access Policies** entry.

>**Note**: This may take a few minutes to appear. Refresh the page if it does not appear.

7. In the Azure portal, navigate back to the **Microsoft Sentinel \| Overview** blade.

8. On the **Microsoft Sentinel \| Overview** blade, review the dashboard and verify that it displays an incident corresponding to the deletion of the Just-in-time VM access policy.

>**Note**: It can take up to 5 minutes for alerts to appear on the **Microsoft Sentinel \| Overview** blade. If you are not seeing an alert at that point, run the query rule referenced in the previous task to verify that the Just-in-time access policy deletion activity has been propagated to the Log Analytics workspace associated with your Microsoft Sentinel instance. If that is not the case, re-create the Just-in-time VM access policy and delete it again.

9. On the **Microsoft Sentinel \| Overview** blade, in the **Threat Management** section, click **Incidents**.

10. Verify that the blade displays an incident with either medium or high severity level.

>**Note**: It can take up to 5 minutes for the incident to appear on the **Microsoft Sentinel \| Incidents** blade.

>**Note**: Review the **Microsoft Sentinel \| Playbooks** blade. You will find there the count of successful and failed runs.

>**Note**: You have the option of assigning a different severity level and status to an incident.

> Results: You have created an Microsoft Sentinel workspace, connected it to Azure Activity logs, created a playbook and custom alerts that are triggered in response to the removal of Justin-time VM access policies, and verified that the configuration is valid.

Clean up resources

> Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs.

1. In the Azure portal, open the Cloud Shell by clicking the first icon in the top right of the Azure Portal. If prompted, click **PowerShell** and **Create storage**.
2. Ensure **PowerShell** is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.
3. In the PowerShell session within the Cloud Shell pane, run the following to remove the resource group you created in this lab:

```
```powershell
```

```
Remove-AzResourceGroup -Name "AZ500LAB131415" -Force -AsJob
```

```
```
```

4. Close the **Cloud Shell** pane.