Contents

1 AZ-104: Microsoft Azure Administrator

There is a Lab Recordings and Demos repo with links to videos of labs used in Microsoft Official Curriculum. The intent is to provide Microsoft Certified Trainers an easy way to access a non-audio version recording of hands-on labs used in the portfolio.

- **Link to labs (HTML format)**
- **Are you a MCT?** - Have a look at our GitHub User Guide for MCTs

1.1 What are we doing?

- To support this course, we will need to make frequent updates to the course content to keep it current with the Azure services used in the course. We are publishing the lab instructions and lab files on GitHub to allow for open contributions between the course authors and MCTs to keep the content current with changes in the Azure platform.
- We hope that this brings a sense of collaboration to the labs like we've never had before - when Azure changes and you find it first during a live delivery, go ahead and make an enhancement right in the lab source. Help your fellow MCTs.

1.2 How should I use these files relative to the released MOC files?

- The instructor handbook and PowerPoints are still going to be your primary source for teaching the course content.
- These files on GitHub are designed to be used in conjunction with the student handbook, but are in GitHub as a central repository so MCTs and course authors can have a shared source for the latest lab files.
- It will be recommended that for every delivery, trainers check GitHub for any changes that may have been made to support the latest Azure services, and get the latest files for their delivery.

1.3 How do I contribute?

- Any MCT can submit a pull request to the code or content in the GitHub repro, Microsoft and the course author will triage and include content and lab code changes as needed.
- You can submit bugs, changes, improvement and ideas. Find a new Azure feature before we have? Submit a new demo!

1.4 What about changes to the student handbook?

- This repository is only for Issues with the course labs. Comments on the course content should be posted on the MCT Courseware Forum.

1.5 Notes

1.5.1 Classroom Materials

1.6 It is strongly recommended that MCTs and Partners access these materials and in turn, provide them separately to students. Pointing students directly to GitHub to access Lab steps as part of an ongoing class will require them to access yet another UI as part of the course, contributing to a confusing experience for the student. An explanation to the student regarding why they are receiving separate Lab instructions can highlight the nature of an always-changing cloud-based interface and platform. Microsoft Learning support for accessing files on GitHub and support for navigation of the GitHub site is limited to MCTs teaching this course only.

1.7 title: Online Hosted Instructions permalink: index.html layout: home

2 Content Directory

Required labs files can be DOWNLOADED HERE

Hyperlinks to each of the lab exercises are listed below.

2.1 Labs

{% assign labs = site.pages | where_exp:"page", "page.url contains '/Instructions/Labs'" %} | Module | Lab | | --- | --- | {% for activity in labs %}| {{ activity.lab.module }} | [{{ activity.lab.title }}{% if activity.lab.type %} - {{ activity.lab.type }}{% endif %}](/home/ll/Azure_clone/Azure_new/AZ-104-MicrosoftAzureAdministrator/{{ site.github.url }}{{ activity.url }}) | {% endfor %}

2.2 lab: title: '01 - Manage Azure Active Directory Identities' module: 'Module 01 - Identity'

3 Lab 01 - Manage Azure Active Directory Identities

4 Student lab manual

4.1 Lab scenario

In order to allow Contoso users to authenticate by using Azure AD, you have been tasked with provisioning users and group accounts. Membership of the groups should be updated automatically based on the user job titles. You also need to create a test Azure AD tenant with a test user account and grant that account limited permissions to resources in the Contoso Azure subscription.

4.2 Objectives

In this lab, you will:

- Task 1: Create and configure Azure AD users
- Task 2: Create Azure AD groups with assigned and dynamic membership
- Task 3: Create an Azure Active Directory (AD) tenant
- Task 4: Manage Azure AD guest users

4.3 Estimated timing: 30 minutes

4.4 Instructions

4.4.1 Exercise 1

4.4.1.1 Task 1: Create and configure Azure AD users

In this task, you will create and configure Azure AD users.

Note: If you have previously used the Trial license for Azure AD Premium on this Azure AD Tenant you will need a new Azure AD Tenant or perform the Task 2 after Task 3 in that new Azure AD tenant.

1. Sign in to the Azure portal.
2. In the Azure portal, search for and select **Azure Active Directory**.
3. On the Azure Active Directory blade, scroll down to the **Manage** section, click **User settings**, and review available configuration options.
4. On the Azure Active Directory blade, in the **Manage** section, click **Users**, and then click your user account to display its **Profile** settings.
5. Click **edit**, in the **Settings** section, set **Usage location** to **United States** and click **save** to apply the change.

   Note: This is necessary in order to assign an Azure AD Premium P2 license to your user account later in this lab.

6. Navigate back to the **Users - All users** blade, and then click **+ New user**.
7. Create a new user with the following settings (leave others with their defaults):

Setting	Value
User name	**az104-01a-aaduser1**
Name	**az104-01a-aaduser1**
Let me create the password	enabled
Initial password	**Pa55w.rd124**
Usage location	**United States**
Job title	**Cloud Administrator**
Department	**IT**

   Note: **Copy to clipboard** the full **User Principal Name** (user name plus domain). You will need it later in this task.

8. In the list of users, click the newly created user account to display its blade.
9. Review the options available in the **Manage** section and note that you can identify the Azure AD roles assigned to the user account as well as the user account's permissions to Azure resources.
10. In the **Manage** section, click **Assigned roles**, then click **+ Add assignment** button and assign the **User administrator** role to **az104-01a-aaduser1**.

    Note: You also have the option of assigning Azure AD roles when provisioning a new user.

11. Open an **InPrivate** browser window and sign in to the Azure portal using the newly created user account. When prompted to update the password, change the password for the user.

    Note: Rather than typing the user name (including the domain name), you can paste the content of Clipboard.

12. In the **InPrivate** browser window, in the Azure portal, search for and select **Azure Active Directory**.

    Note: While this user account can access the Azure Active Directory tenant, it does not have any access to Azure resources. This is expected, since such access would need to be granted explicitly by using Azure Role-Based Access Control.

13. In the **InPrivate** browser window, on the Azure AD blade, scroll down to the **Manage** section, click **User settings**, and note that you do not have permissions to modify any configuration options.
14. In the **InPrivate** browser window, on the Azure AD blade, in the **Manage** section, click **Users**, and then click **+ New user**.
15. Create a new user with the following settings (leave others with their defaults):

Setting	Value
User name	**az104-01a-aaduser2**
Name	**az104-01a-aaduser2**

Setting	Value
Let me create the password	enabled
Initial password	**Pa55w.rd124**
Usage location	**United States**
Job title	**System Administrator**
Department	**IT**

16. Sign out as the az104-01a-aaduser1 user from the Azure portal and close the InPrivate browser window.

4.4.1.2 Task 2: Create Azure AD groups with assigned and dynamic membership

In this task, you will create Azure Active Directory groups with assigned and dynamic membership.

1. Back in the Azure portal where you are signed in with your **user account**, navigate back to the **Overview** blade of the Azure AD tenant and, in the **Manage** section, click **Licenses**.

   Note: Azure AD Premium P1 or P2 licenses are required in order to implement dynamic groups.

2. In the **Manage** section, click **All products**.
3. Click **+ Try/Buy** and activate the free trial of Azure AD Premium P2.
4. Refresh the browser window to verify that the activation was successful.
5. From the **Licenses - All products** blade, select the **Azure Active Directory Premium P2** entry, and assign all license options of Azure AD Premium P2 to your user account and the two newly created user accounts.
6. In the Azure portal, navigate back to the Azure AD tenant blade and click **Groups**.
7. Use the **+ New group** button to create a new group with the following settings:

Setting	Value
Group type	**Security**
Group name	**IT Cloud Administrators**
Group description	**Contoso IT cloud administrators**
Membership type	**Dynamic User**

   Note: If the **Membership type** drop-down list is grayed out, wait a few minutes and refresh the browser page.

8. Click **Add dynamic query**.
9. On the **Configure Rules** tab of the **Dynamic membership rules** blade, create a new rule with the following settings:

Setting	Value
Property	**jobTitle**
Operator	**Equals**
Value	**Cloud Administrator**

10. Save the rule and, back on the **New Group** blade, click **Create**.
11. Back on the **Groups - All groups** blade of the Azure AD tenant, click the **+ New group** button and create a new group with the following settings:

Setting	Value
Group type	**Security**
Group name	**IT System Administrators**
Group description	**Contoso IT system administrators**
Membership type	**Dynamic User**

12. Click **Add dynamic query**.
13. On the **Configure Rules** tab of the **Dynamic membership rules** blade, create a new rule with the following settings:

Setting	Value
Property	**jobTitle**
Operator	**Equals**
Value	**System Administrator**

14. Save the rule and, back on the **New Group** blade, click **Create**.
15. Back on the **Groups - All groups** blade of the Azure AD tenant, click the + **New group** button, and create a new group with the following settings:

Setting	Value
Group type	**Security**
Group name	**IT Lab Administrators**
Group description	**Contoso IT Lab administrators**
Membership type	**Assigned**

16. Click **No members selected**.
17. From the **Add members** blade, search and select the **IT Cloud Administrators** and **IT System Administrators** groups and, back on the **New Group** blade, click **Create**.
18. Back on the **Groups - All groups** blade, click the entry representing the **IT Cloud Administrators** group and, on then display its **Members** blade. Verify that the **az104-01a-aaduser1** appears in the list of group members.

    Note: You might experience delays with updates of the dynamic membership groups. To expedite the update, navigate to the group blade, display its **Dynamic membership rules** blade, **Edit** the rule listed in the **Rule syntax** textbox by adding a whitespace at the end, and **Save** the change.

19. Navigate back to the **Groups - All groups** blade, click the entry representing the **IT System Administrators** group and, on then display its **Members** blade. Verify that the **az104-01a-aaduser2** appears in the list of group members.

4.4.1.3 Task 3: Create an Azure Active Directory (AD) tenant

In this task, you will create a new Azure AD tenant.

1. In the Azure portal, search for and select **Azure Active Directory**.
2. Click + **Create a tenant** and specify the following setting:

Setting	Value
Directory type	**Azure Active Directory**

3. Click **Next : Configuration**

Setting	Value
Organization name	**Contoso Lab**
Initial domain name	any valid DNS name consisting of lower case letters and digits and starting with a letter
Country/Region	**United States**

Note: The **Initial domain name** should not be a legitimate name that potentially matches your organization or another. The green check mark in the **Initial domain name** text box will indicate that the domain name you typed in is valid and unique.

4. Click **Review + create** and then click **Create**.
5. Display the blade of the newly created Azure AD tenant by using the **Click here to navigate to your new tenant: Contoso Lab** link or the **Directory + Subscription** button (directly to the right of the Cloud Shell button) in the Azure portal toolbar.

4.4.1.4 Task 4: Manage Azure AD guest users.

In this task, you will create Azure AD guest users and grant them access to resources in an Azure subscription.

1. In the Azure portal displaying the Contoso Lab Azure AD tenant, in the **Manage** section, click **Users**, and then click **+ New user**.
2. Create a new user with the following settings (leave others with their defaults):

Setting	Value
User name	**az104-01b-aaduser1**
Name	**az104-01b-aaduser1**
Let me create the password	enabled
Initial password	**Pa55w.rd124**
Job title	**System Administrator**
Department	**IT**

3. Click on the newly created profile.

   Note: **Copy to clipboard** the full **User Principal Name** (user name plus domain). You will need it later in this task.

4. Switch back to your default Azure AD tenant by using the **Directory + Subscription** button (directly to the right of the Cloud Shell button) in the Azure portal toolbar.
5. Navigate back to the **Users - All users** blade, and then click **+ New guest user**.
6. Create a new guest user with the following settings (leave others with their defaults):

Setting	Value
Name	**az104-01b-aaduser1**
Email address	the User Principal Name you copied earlier in this task
Usage location	**United States**
Job title	**Lab Administrator**
Department	**IT**

7. Click **Invite**.
8. Back on the **Users - All users** blade, click the entry representing the newly created guest user account.
9. On the **az104-01b-aaduser1 - Profile** blade, click **Groups**.
10. Click **+ Add membership** and add the guest user account to the **IT Lab Administrators** group.

4.4.1.5 Clean up resources

Note: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not incur unexpected costs. While, in this case, there are no additional charges associated with Azure Active Directory tenants and their objects, you might want to consider removing the user accounts, the group accounts, and the Azure Active Directory tenant you created in this lab.

1. In the **Azure Portal** search for **Azure Active Directory** in the search bar. Within **Azure Active Directory** under **Manage** select **Licenses**. Once at **Licenses** under **Manage** select **All Products** and then select **Azure Active Directory Premium P2** item in the list. Proceed by then selecting **Licensed**

Users. Select the user accounts **az104-01a-aaduser1** and **az104-01a-aaduser2** to which you assigned licenses in this lab, click **Remove license**, and, when prompted to confirm, click **OK**.

2. In the Azure portal, navigate to the **Users - All users** blade, click the entry representing the **az104-01b-aaduser1** guest user account, on the **az104-01b-aaduser1 - Profile** blade click **Delete**, and, when prompted to confirm, click **OK**.
3. Repeat the same sequence of steps to delete the remaining user accounts you created in this lab.
4. Navigate to the **Groups - All groups** blade, select the groups you created in this lab, click **Delete**, and, when prompted to confirm, click **OK**.
5. In the Azure portal, display the blade of the Contoso Lab Azure AD tenant by using the **Directory + Subscription** button (directly to the right of the Cloud Shell button) in the Azure portal toolbar.
6. Navigate to the **Users - All users** blade, click the entry representing the **az104-01b-aaduser1** user account, on the **az104-01b-aaduser1 - Profile** blade click **Delete**, and, when prompted to confirm, click **OK**.
7. Navigate to the **Contoso Lab - Overview** blade of the Contoso Lab Azure AD tenant, click **Delete tenant**, on the **Delete tenant 'Contoso Lab'** blade, click the **Get permission to delete Azure resources** link, on the **Properties** blade of Azure Active Directory, set **Access management for Azure resources** to **Yes** and click **Save**.
8. Sign out from the Azure portal and sign in back.
9. Navigate back to the **Delete tenant 'Contoso Lab'** blade and click **Delete**.

   Note: You will have to wait for the trial license expiration before you can delete the tenant. This does not incur any additional cost.

4.4.1.6 Review

In this lab, you have:

- Created and configured Azure AD users
- Created Azure AD groups with assigned and dynamic membership
- Created an Azure Active Directory (AD) tenant
- Managed Azure AD guest users

4.5 lab: title: '02a - Manage Subscriptions and RBAC' module: 'Module 02 - Governance and Compliance'

5 Lab 02a - Manage Subscriptions and RBAC

6 Student lab manual

6.1 Lab requirements:

This lab requires permissions to create Azure Active Directory (Azure AD) users, create custom Azure Role Based Access Control (RBAC) roles, and assign these roles to Azure AD users. Not all lab hosters may provide this capability. Ask your instructor for the availability of this lab.

6.2 Lab scenario

In order to improve management of Azure resources in Contoso, you have been tasked with implementing the following functionality:

- creating a management group that would include all of Contoso's Azure subscriptions
- granting permissions to submit support requests for all subscriptions in the management group to a designated Azure Active Directory user. That user's permissions should be limited only to:
  - creating support request tickets
  - viewing resource groups

6.3 Objectives

In this lab, you will:

- Task 1: Implement Management Groups
- Task 2: Create custom RBAC roles
- Task 3: Assign RBAC roles

6.4 Estimated timing: 30 minutes

6.5 Instructions

6.5.1 Exercise 1

6.5.1.1 Task 1: Implement Management Groups

In this task, you will create and configure management groups.

1. Sign in to the Azure portal.

2. Search for and select **Management groups** to navigate to the **Management groups** blade.

3. Review the messages at the top of the **Management groups** blade. If you are seeing the message stating **You are registered as a directory admin but do not have the necessary permissions to access the root management group**, perfom the following sequence of steps:

    1. In the Azure portal, search for and select **Azure Active Directory**.

    2. On the blade displaying properties of your Azure Active Directory tenant, in the vertical menu on the left side, in the **Management** section, select **Properties**.

    3. On the **Properties** blade of your your Azure Active Directory tenant, in the **Access management for Azure resources** section, select **Yes** and then select **Save**.

    4. Navigate back to the **Management groups** blade, and select **Refresh**.

4. On the **Management groups** blade, click **+ Add**.

    > **Note**: If you have not previously created Management Groups, select **Start using management groups**

5. Create a management group with the following settings:

Setting	Value
Management group ID	**az104-02-mg1**
Management group display name	**az104-02-mg1**

6. In the list of management groups, click the entry representing the newly created management group.

7. On the **az104-02-mg1** blade, click **Subscriptions**.

8. On the **az104-02-mg1 | Subscriptions** blade, click **+ Add**, on the **Add subscription** blade, in the **Subscription** drop-down list, seletc the subscription you are using in this lab and click **Save**.

    > **Note**: On the **az104-02-mg1 | Subscriptions** blade, copy the ID of your Azure subscription into Clipboard. You will need it in the next task.

6.5.1.2 Task 2: Create custom RBAC roles

In this task, you will create a definition of a custom RBAC role.

1. From the lab computer, open the file **\Allfiles\Labs\02\az104-02a-customRoleDefinition.json** in Notepad and review its content:

    ```
    {
       "Name": "Support Request Contributor (Custom)",
       "IsCustom": true,
       "Description": "Allows to create support requests",
       "Actions": [
    ```

```
            "Microsoft.Resources/subscriptions/resourceGroups/read",
            "Microsoft.Support/*"
        ],
        "NotActions": [
        ],
        "AssignableScopes": [
            "/providers/Microsoft.Management/managementGroups/az104-02-mg1",
            "/subscriptions/SUBSCRIPTION_ID"
        ]
    }
```

2. Replace the `SUBSCRIPTION_ID` placeholder in the JSON file with the subscription ID you copied into Clipboard and save the change.
3. In the Azure portal, open **Cloud Shell** pane by clicking on the toolbar icon directly to the right of the search textbox.
4. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

5. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu click **Upload**, and upload the file **\Allfiles\Labs\02\az104-02a-customRoleDefinition.json** into the Cloud Shell home directory.
6. From the Cloud Shell pane, run the following to create the custom role definition:

   ```
   New-AzRoleDefinition -InputFile $HOME/az104-02a-customRoleDefinition.json
   ```

7. Close the Cloud Shell pane.

6.5.1.3 Task 3: Assign RBAC roles

In this task, you will create an Azure Active Directory user, assign the RBAC role you created in the previous task to that user, and verify that the user can perform the task specified in the RBAC role definition.

1. In the Azure portal, search for and select **Azure Active Directory**, on the Azure Active Directory blade, click **Users**, and then click **+ New user**.
2. Create a new user with the following settings (leave others with their defaults):

Setting	Value
User name	**az104-02-aaduser1**
Name	**az104-02-aaduser1**
Let me create the password	enabled
Initial password	**Pa55w.rd124**

   > **Note**: **Copy to clipboard** the full **User name**. You will need it later in this lab.

3. In the Azure portal, navigate back to the **az104-02-mg1** management group and display its **details**.
4. Click **Access control (IAM)**, click **+ Add** followed by **Role assignment**, and assign the **Support Request Contributor (Custom)** role to the newly created user account.
5. Open an **InPrivate** browser window and sign in to the Azure portal using the newly created user account. When prompted to update the password, change the password for the user.

   > **Note**: Rather than typing the user name, you can paste the content of Clipboard.

6. In the **InPrivate** browser window, in the Azure portal, search and select **Resource groups** to verify that the az104-02-aaduser1 user can see all resource groups.
7. In the **InPrivate** browser window, in the Azure portal, search and select **All resources** to verify that the az104-02-aaduser1 user cannot see any resources.
8. In the **InPrivate** browser window, in the Azure portal, search and select **Help + support** and then click **+ New support request**.

9. In the **InPrivate** browser window, on the **Basic** tab of the **Help + support - New support request** blade, select the **Service and subscription limits (quotas)** issue type and note that the subscription you are using in this lab is listed in the **Subscription** drop-down list.

    > **Note**: The presence of the subscription you are using in this lab in the **Subscription** drop-down list indicates that the account you are using has the permissions required to create the subscription-specific support request.

    > **Note**: If you do not see the **Service and subscription limits (quotas)** option, sign out from the Azure portal and sign in back.

10. Do not continue with creating the support request. Instead, sign out as the az104-02-aaduser1 user from the Azure portal and close the InPrivate browser window.

6.5.1.4 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use.

> **Note**: Removing unused resources ensures you will not see unexpected charges, although, resources created in this lab do not incur extra cost.

1. In the Azure portal, search for and select **Azure Active Directory**, on the Azure Active Directory blade, click **Users**.
2. On the **Users - All users** blade, click **az104-02-aaduser1**.
3. On the **az104-02-aaduser1 - Profile** blade, copy the value of **Object ID** attribute.
4. In the Azure portal, start a **PowerShell** session within the **Cloud Shell**.
5. From the Cloud Shell pane, run the following to remove the assignment of the custom role definition (replace the `[object_ID]` placeholder with the value of the **object ID** attribute of the **az104-02-aaduser1** Azure Active Directory user account you copied earlier in this task):

    ```
    $scope = (Get-AzRoleAssignment -RoleDefinitionName 'Support Request Contributor (Custom)').Scope

    Remove-AzRoleAssignment -ObjectId '[object_ID]' -RoleDefinitionName 'Support Request Contributor (
    ```

6. From the Cloud Shell pane, run the following to remove the custom role definition:

    ```
    Remove-AzRoleDefinition -Name 'Support Request Contributor (Custom)' -Force
    ```

7. In the Azure portal, navigate back to the **Users - All users** blade of the **Azure Active Directory**, and delete the **az104-02-aaduser1** user account.
8. In the Azure portal, navigate back to the **Management groups** blade.
9. On the **Management groups** blade, in the **Child subscriptions** column, in the row representing the name of the management group to which you want to move the Azure subscription you used in this lab, select the link represeting its current number of subscriptions.

    > **Note**: It is likely that the target management group is the **Tenant Root management group**, unless you created a custom management group hierarchy before running this lab.

10. On the **Subscriptions** blade of the target management group, select **+ Add**.
11. On the **Add subscription** blade, in the **Subscriptions** drop-down list, select the name of the Azure subscription you used in this lab and click **Save**.
12. Navigate back to the **Management groups** blade, right click the **ellipsis** icon to the right of the **az104-02-mg1** management group and click **Delete**.

6.5.1.5 Review

In this lab, you have:

- Implemented Management Groups
- Created custom RBAC roles
- Assigned RBAC roles

6.6 lab: title: '02b - Manage Governance via Azure Policy' module: 'Module 02 - Governance and Compliance'

7 Lab 02b - Manage Governance via Azure Policy

8 Student lab manual

8.1 Lab scenario

In order to improve management of Azure resources in Contoso, you have been tasked with implementing the following functionality:

- tagging resource groups that include only infrastructure resources (such as Cloud Shell storage accounts)
- ensuring that only properly tagged infrastructure resources can be added to infrastructure resource groups
- remediating any non-compliant resources

8.2 Objectives

In this lab, we will:

- Task 1: Create and assign tags via the Azure portal
- Task 2: Enforce tagging via an Azure policy
- Task 3: Apply tagging via an Azure policy

8.3 Estimated timing: 30 minutes

8.4 Instructions

8.4.1 Exercise 1

8.4.1.1 Task 1: Assign tags via the Azure portal

In this task, you will create and assign a tag to an Azure resource group via the Azure portal.

1. In the Azure portal, start a **PowerShell** session within the **Cloud Shell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

2. From the Cloud Shell pane, run the following to identify the name of the storage account used by Cloud Shell:

   ```
   df
   ```

3. In the output of the command, note the first part of the fully qualified path designating the Cloud Shell home drive mount (marked here as `xxxxxxxxxxxxxx`:

   ```
   //xxxxxxxxxxxxxx.file.core.windows.net/cloudshell   (..)  /usr/csuser/clouddrive
   ```

4. In the Azure portal, search and select **Storage accounts** and, in the list of the storage accounts, click the entry representing the storage account you identified in the previous step.

5. On the storage account blade, click the link representing the name of the resource group containing the storage account.

   Note: note what resource group the storage account is in, you'll need it later in the lab.

6. On the resource group blade, click **Tags**.

7. Create a tag with the following settings and save your change:

Setting	Value
Name	**Role**
Value	**Infra**

8. Navigate back to the storage account blade. Review the **Overview** information and note that the new tag was not automatically assigned to the storage account.

8.4.1.2 Task 2: Enforce tagging via an Azure policy

In this task, you will assign the built-in *Require a tag and its value on resources* policy to the resource group and evaluate the outcome.

1. In the Azure portal, search for and select **Policy**.
2. In the **Authoring** section, click **Definitions**. Take a moment to browse through the list of built-in policy definitions that are available for you to use. List all built-in policies that involve the use of tags by selecting the **Tags** entry (and de-selecting all other entries) in the **Category** drop-down list.
3. Click the entry representing the **Require a tag and its value on resources** built-in policy and review its definition.
4. On the **Require a tag and its value on resources** built-in policy definition blade, click **Assign**.
5. Specify the **Scope** by clicking the ellipsis button and selecting the following values:

Setting	Value
Subscription	the name of the Azure subscription you are using in this lab
Resource Group	the name of the resource group containing the Cloud Shell account you identified in the previous task

> **Note**: A scope determines the resources or resource groups where the policy assignment takes effect. You could assign policies on the management group, subscription, or resource group level. You also have the option of specifying exclusions, such as individual subscriptions, resource groups, or resources (depending on the assignment scope).

6. Configure the **Basics** properties of the assignment by specifying the following settings (leave others with their defaults):

Setting	Value
Assignment name	**Require Role tag with Infra value**
Description	**Require Role tag with Infra value for all resources in the Cloud Shell resource group**
Policy enforcement	Enabled

> **Note**: The **Assignment name** is automatically populated with the policy name you selected, but you can change it. You can also add an optional **Description**. **Assigned by** is automatically populated based on the user name creating the assignment.

7. Click **Next** and set **Parameters** to the following values:

Setting	Value
Tag Name	**Role**
Tag Value	**Infra**

8. Click **Next** and review the **Remediation** tab. Leave the **Create a Managed Identity** checkbox unchecked.

> **Note**: This setting can be used when the policy or initiative includes the **deployIfNotExists** or **Modify** effect.

9. Click **Review + Create** and then click **Create**.

> **Note**: Now you will verify that the new policy assignment is in effect by attempting to create another Azure Storage account in the resource group without explicitly adding the required tag.

> **Note**: It might take between 5 and 15 minutes for the policy to take effect.

10. Navigate back to the blade of the resource group hosting the storage account used for the Cloud Shell home drive, which you identified in the previous task.

11. On the resource group blade, click **+ Add**.
12. On the **New** blade, search for and select **Storage account**, and click **Create**.
13. On the **Basics** tab of the **Create storage account** blade, specify the following settings (leave others with their defaults), click **Review + create** and then click **Create**:

Setting	Value
Storage account name	any globally unique combination of between 3 and 24 lower case letters and digits, starting with a

14. Once you create the deployment, you should see the **Deployment failed** message in the **Notifications** list of the portal. From the **Notifications** list, navigate to the deployment overview and click the **Deployment failed. Click here for details** message to identify the reason for the failure.

    > **Note**: Verify whether the error message states that the resource deployment was disallowed by the policy.

    > **Note**: By clicking the **Raw Error** tab, you can find more details about the error, including the name of the role definition **Require Role tag with Infra value**. The deployment failed because the storage account you attempted to create did not have a tag named **Role** with its value set to **Infra**.

8.4.1.3 Task 3: Apply tagging via an Azure policy

In this task, we will use a different policy definition to remediate any non-compliant resources.

1. In the Azure portal, search for and select **Policy**.
2. In the **Authoring** section, click **Assignments**.
3. In the list of assignments, right click the ellipsis icon in the row representing the **Require Role tag with Infra value** policy assignment and use the **Delete assignment** menu item to delete the assignment.
4. Click **Assign policy** and specify the **Scope** by clicking the ellipsis button and selecting the following values:

Setting	Value
Subscription	the name of the Azure subscription you are using in this lab
Resource Group	the name of the resource group containing the Cloud Shell account you identified in the first task

5. To specify the **Policy definition**, click the ellipsis button and then search for and select **Inherit a tag from the resource group if missing**.
6. Configure the remaining **Basics** properties of the assignment by specifying the following settings (leave others with their defaults):

Setting	Value
Assignment name	**Inherit the Role tag and its Infra value from the Cloud Shell resource group if missing**
Description	**Inherit the Role tag and its Infra value from the Cloud Shell resource group if missing**
Policy enforcement	Enabled

7. Click **Next** and set **Parameters** to the following values:

Setting	Value
Tag Name	**Role**

8. Click **Next** and, on the **Remediation** tab, configure the following settings (leave others with their defaults):

Setting	Value
Create a remediation task	enabled
Policy to remediate	**Inherit a tag from the resource group if missing**

> **Note**: This policy definition includes the **Modify** effect.

9. Click **Review + Create** and then click **Create**.

   > **Note**: To verify that the new policy assignment is in effect, you will create another Azure Storage account in the same resource group without explicitly adding the required tag.

   > **Note**: It might take between 5 and 15 minutes for the policy to take effect.

10. Navigate back to the blade of the resource group hosting the storage account used for the Cloud Shell home drive, which you identified in the first task.
11. On the resource group blade, click **+ Add**.
12. On the **New** blade, search for and select **Storage account**, and click **Create**.
13. On the **Basics** tab of the **Create storage account** blade, specify the following settings (leave others with their defaults) and click **Review + create**:

Setting	Value
Storage account name	any globally unique combination of between 3 and 24 lower case letters and digits, starting with a

14. Verify that this time the validation passed and click **Create**.
15. Once the new storage account is provisioned, click **Go to resource** button and, on the **Overview** blade of the newly created storage account, note that the tag **Role** with the value **Infra** has been automatically assigned to the resource.

8.4.1.4 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use.

> **Note**: Removing unused resources ensures you will not see unexpected charges, although keep in mind that Azure policies do not incur extra cost.

1. In the portal, search for and select **Policy**.
2. In the **Authoring** section, click **Assignments**, click the ellipsis icon to the right of the assignment you created in the previous task and click **Delete assignment**.
3. In the portal, search for and select **Storage accounts**.
4. In the list of storage accounts, select the resource group corresponding to the storage account you created in the last task of this lab. Select **Tags** and click **Delete** (Trash can to the right) to the **Role:Infra** tag and press **Save**.
5. In the portal, again search for and select **Storage accounts** or use the menu at the top to select **Storage accounts**
6. In the list of storage accounts, select the storage account you created in the last task of this lab, click **Delete**, when prompted for the confirmation, in the **Confirm delete** type **yes** and click **Delete**.

8.4.1.5 Review

In this lab, you have:

- Created and assigned tags via the Azure portal
- Enforced tagging via an Azure policy
- Applied tagging via an Azure policy

8.5 lab: title: '03a - Manage Azure resources by Using the Azure Portal' module: 'Module 03 - Azure Administration'

9 Lab 03a - Manage Azure resources by Using the Azure Portal

10 Student lab manual

10.1 Lab scenario

You need to explore the basic Azure administration capabilities associated with provisioning resources and organizing them based on resource groups, including moving resources between resource groups. You also want to explore options for protecting disk resources from being accidentally deleted, while still allowing for modifying their performance characteristics and size.

10.2 Objectives

In this lab, we will:

- Task 1: Create resource groups and deploy resources to resource groups
- Task 2: Move resources between resource groups
- Task 3: Implement and test resource locks

10.3 Estimated timing: 20 minutes

10.4 Instructions

10.4.1 Exercise 1

10.4.1.1 Task 1: Create resource groups and deploy resources to resource groups

In this task, you will use the Azure portal to create resource groups and create a disk in the resource group.

1. Sign in to the Azure portal.

2. In the Azure portal, search for and select **Disks**, click **+ Add, + Create, or + New**, and specify the following settings:

Setting	Value
Subscription	the name of the Azure subscription where you created the resource group
Resource Group	the name of a new resource group **az104-03a-rg1**
Disk name	**az104-03a-disk1**
Region	the name of the Azure region where you created the resource group
Availability zone	**None**
Source type	**None**

> **Note**: When creating a resource, you have the option of creating a new resource group or using an existing one.

3. Change the disk type and size to **Standard HDD** and **32 GiB**, respectively.

4. Click **Review + Create** and then click **Create**.

> **Note**: Wait until the disk is created. This should take less than a minute.

10.4.1.2 Task 2: Move resources between resource groups

In this task, we will move the disk resource you created in the previous task to a new resource group.

1. Search for and select **Resource groups**.

2. On the **Resource groups** blade, click the entry representing the **az104-03a-rg1** resource group you created in the previous task.

3. From the **Overview** blade of the resource group, in the list of resource group resources, select the entry representing the newly created disk, click **Move** in the toolbar, and, in the drop-down list, select **Move to another resource group**.

> **Note**: This method allows you to move multiple resources at the same time.

4. On the **Move resources** blade, click **Create a new group**.
5. Below the **Resource group** text box, click **Create a new group** then type **az104-03a-rg2** in the text box, select the checkbox **I understand that tools and scripts associated with moved resources will not work until I update them to use new resource IDs**, and click **OK**.

   > **Note**: Do not wait for the move to complete but instead proceed to the next task. The move might take about 10 minutes. You can determine that the operation was completed by monitoring activity log entries of the source or target resource group. Revisit this step once you complete the next task.

10.4.1.3 Task 3: Implement resource locks

In this task, you will apply a resource lock to an Azure resource group containing a disk resource.

1. In the Azure portal, search for and select **Disks**, click **+ Add, + Create, or + New**, and specify the following settings:

Setting	Value
Subscription	the name of the subscription you are using in this lab
Resource Group	click **create new** resource group and name it **az104-03a-rg3**
Disk name	**az104-03a-disk2**
Region	the name of the Azure region where you created the other resource groups in this lab
Availability zone	**None**
Source type	**None**

2. Set the disk type and size to **Standard HDD** and **32 GiB**, respectively.
3. Click **Review + Create** and then click **Create**.
4. Click Go to resouce.
5. On the **az104-03a-rg3** resource group blade, click **Locks** then **+ Add** and specify the following settings:

Setting	Value
Lock name	**az104-03a-delete-lock**
Lock type	**Delete**

6. Click **OK**
7. On the **az104-03a-rg3** resource group blade, click **Overview**, in the list of resource group resources, select the entry representing the disk you created earlier in this task, and click **Delete** in the toolbar.
8. When prompted **Do you want to delete all the selected resources?**, in the **Confirm delete** text box, type **yes** and click **Delete**.
9. You should see an error message, notifying about the failed delete operation.

   > **Note**: As the error message states, this is expected due to the delete lock applied on the resource group level.

10. Navigate back to the list of resources of the **az104-03a-rg3** resource group and click the entry representing the **az104-03a-disk2** resource.
11. On the **az104-03a-disk2** blade, in the **Settings** section, click **Size + performance**, set the disk type and size to **Premium SSD** and **64 GiB**, respectively, and click **Resize** to apply the change. Verify that the change was successful.

    > **Note**: This is expected, since the resource group-level lock applies to delete operations only.

10.4.1.4 Clean up resources

> **Note**: Do not delete resources you deployed in this lab. You will be using them in the next lab of this module. Remove only the resource lock you created in this lab.

1. Navigate to the **az104-03a-rg3** resource group blade, display its **Locks** blade, and remove the lock **az104-03a-delete-lock** by clicking the **Delete** link on the right-hand side of the **Delete** lock entry.

10.4.1.5 Review

In this lab, you have:

- Created resource groups and deployed resources to resource groups
- Moved resources between resource groups
- Implemented and tested resource locks

10.5 lab: title: '03b - Manage Azure resources by Using ARM Templates' module: 'Module 03 - Azure Administration'

11 Lab 03b - Manage Azure resources by Using ARM Templates

12 Student lab manual

12.1 Lab scenario

Now that you explored the basic Azure administration capabilities associated with provisioning resources and organizing them based on resource groups by using the Azure portal, you need to carry out the equivalent task by using Azure Resource Manager templates.

12.2 Objectives

In this lab, you will:

- Task 1: Review an ARM template for deployment of an Azure managed disk
- Task 2: Create an Azure managed disk by using an ARM template
- Task 3: Review the ARM template-based deployment of the managed disk

12.3 Estimated timing: 20 minutes

12.4 Instructions

12.4.1 Exercise 1

12.4.1.1 Task 1: Review an ARM template for deployment of an Azure managed disk

In this task, you will create an Azure disk resource by using an Azure Resource Manager template.

1. Sign in to the Azure portal.

2. In the Azure portal, search for and select **Resource groups**.

3. In the list of resource groups, click **az104-03a-rg1**.

4. On the **az104-03a-rg1** resource group blade, in the **Settings** section, click **Deployments**.

5. On the **az104-03a-rg1 - Deployments** blade, click the first entry in the list of deployments.

6. On the **Microsoft.ManagedDisk-*XXXXXXXXX* | Overview** blade, click **Template**.

   > **Note**: Review the content of the template and note that you have the option to **Download** it to the local computer, **Add to library**, or **Deploy** it again.

7. Click **Download** and save the compressed file containing the template and parameters files to the **Downloads** folder on your lab computer.

8. On the **Microsoft.ManagedDisk-*XXXXXXXXX* | Template** blade, click **Inputs**.

9. Note the value of the **location** parameter. You will need it in the next task.

10. Extract the content of the downloaded file into the **Downloads** folder on your lab computer.

Note: These files are also available as **\Allfiles\Labs\03\az104-03b-md-template.json** and **\Allfiles\Labs\03\az104-03b-md-parameters.json**

11. Close all **File Explorer** windows.

12.4.1.2 Task 2: Create an Azure managed disk by using an ARM template

1. In the Azure portal, search for and select **Deploy a custom template**.
2. Click **Template deployment (deploy using custom templates)** found under the **Marketplace** group.
3. On the **Custom deployment** blade, click **Build your own template in the editor**.
4. On the **Edit template** blade, click **Load file** and upload the **template.json** file you downloaded in the previous task.
5. Within the editor pane, remove the following lines:

```
"sourceResourceId": {
    "type": "String"
},
"sourceUri": {
    "type": "String"
},
"sourceImageVersionId": {
    "type": "String"
},
"osType": {
    "type": "String"
},
```

```
"hyperVGeneration": {
    "defaultValue": "V1",
    "type": "String"
},
```

```
"osType": "[parameters('osType')]",
```

Note: These parameters are removed since they are not applicable to the current deployment. In particular, sourceResourceId, sourceUri, osType, and hyperVGeneration parameters are applicable to creating an Azure disk from an existing VHD file.

6. **Save** the changes.
7. Back on the **Custom deployment** blade, click **Edit parameters**.
8. On the **Edit parameters** blade, click **Load file** and upload the **parameters.json** file you downloaded in the previous task, and **Save** the changes.
9. Back on the **Custom deployment** blade, specify the following settings:

Setting	Value
Subscription	*the name of the Azure subscription you are using in this lab*
Resource Group	the name of a **new** resource group **az104-03b-rg1**
Region	the name of any Azure region available in the subscription you are using in this lab
Disk Name	**az104-03b-disk1**
Location	the value of the location parameter you noted in the previous task
Sku	**Standard_LRS**
Disk Size Gb	**32**
Create Option	**empty**
Disk Encryption Set Type	**EncryptionAtRestWithPlatformKey**
Network Access Policy	**AllowAll**

10. Select **Review + Create** and then select **Create**.
11. Verify that the deployment completed successfully.

12.4.1.3 Task 3: Review the ARM template-based deployment of the managed disk

1. In the Azure portal, search for and select **Resource groups**.
2. In the list of resource groups, click **az104-03b-rg1**.
3. On the **az104-03b-rg1** resource group blade, in the **Settings** section, click **Deployments**.
4. From the **az104-03b-rg1 - Deployments** blade, click the first entry in the list of deployments and review the content of the **Input** and **Template** blades.

12.4.1.4 Clean up resources

> **Note**: Do not delete resources you deployed in this lab. You will reference them in the next lab of this module.

12.4.1.5 Review

In this lab, you have:

- Reviewed an ARM template for deployment of an Azure managed disk
- Created an Azure managed disk by using an ARM template
- Reviewed the ARM template-based deployment of the managed disk

---

12.5 lab: title: '03c - Manage Azure resources by Using Azure PowerShell' module: 'Module 03 - Azure Administration'

13 Lab 03c - Manage Azure resources by Using Azure PowerShell

14 Student lab manual

14.1 Lab scenario

Now that you explored the basic Azure administration capabilities associated with provisioning resources and organizing them based on resource groups by using the Azure portal and Azure Resource Manager templates, you need to carry out the equivalent task by using Azure PowerShell. To avoid installing Azure PowerShell modules, you will leverage PowerShell environment available in Azure Cloud Shell.

14.2 Objectives

In this lab, you will:

- Task 1: Start a PowerShell session in Azure Cloud Shell
- Task 2: Create a resource group and an Azure managed disk by using Azure PowerShell
- Task 3: Configure the managed disk by using Azure PowerShell

14.3 Estimated timing: 20 minutes

14.4 Instructions

14.4.1 Exercise 1

14.4.1.1 Task 1: Start a PowerShell session in Azure Cloud Shell

In this task, you will open a PowerShell session in Cloud Shell.

1. In the portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
2. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

3. If prompted, click **Create storage**, and wait until the Azure Cloud Shell pane is displayed.
4. Ensure **PowerShell** appears in the drop-down menu in the upper-left corner of the Cloud Shell pane.

14.4.1.2 Task 2: Create a resource group and an Azure managed disk by using Azure PowerShell

In this task, you will create a resource group and an Azure managed disk by using Azure PowerShell session within Cloud Shell

1. To create a resource group in the same Azure region as the **az104-03b-rg1** resource group you created in the previous lab, from the PowerShell session within Cloud Shell, run the following:

   ```
   $location = (Get-AzResourceGroup -Name az104-03b-rg1).Location

   $rgName = 'az104-03c-rg1'

   New-AzResourceGroup -Name $rgName -Location $location
   ```

2. To retrieve properties of the newly created resource group, run the following:

   ```
   Get-AzResourceGroup -Name $rgName
   ```

3. To create a new managed disk with the same characteristics as those you created in the previous labs of this module, run the following:

   ```
   $diskConfig = New-AzDiskConfig `
    -Location $location `
    -CreateOption Empty `
    -DiskSizeGB 32 `
    -Sku Standard_LRS

   $diskName = 'az104-03c-disk1'

   New-AzDisk `
    -ResourceGroupName $rgName `
    -DiskName $diskName `
    -Disk $diskConfig
   ```

4. To retrieve properties of the newly created disk, run the following:

   ```
   Get-AzDisk -ResourceGroupName $rgName -Name $diskName
   ```

14.4.1.3 Task 3: Configure the managed disk by using Azure PowerShell

In this task, you will managing configuration of the Azure managed disk by using Azure PowerShell session within Cloud Shell.

1. To increase the size of the Azure managed disk to **64 GB**, from the PowerShell session within Cloud Shell, run the following:

   ```
   New-AzDiskUpdateConfig -DiskSizeGB 64 | Update-AzDisk -ResourceGroupName $rgName -DiskName $diskNa
   ```

2. To verify that the change took effect, run the following:

   ```
   Get-AzDisk -ResourceGroupName $rgName -Name $diskName
   ```

3. To verify the current SKU as **Standard_LRS**, run the following:

   ```
   (Get-AzDisk -ResourceGroupName $rgName -Name $diskName).Sku
   ```

4. To change the disk performance SKU to **Premium_LRS**, from the PowerShell session within Cloud Shell, run the following:

   ```
   New-AzDiskUpdateConfig -Sku Premium_LRS | Update-AzDisk -ResourceGroupName $rgName -DiskName $disk
   ```

5. To verify that the change took effect, run the following:

   ```
   (Get-AzDisk -ResourceGroupName $rgName -Name $diskName).Sku
   ```

14.4.1.4 Clean up resources

> **Note**: Do not delete resources you deployed in this lab. You will reference them in the next lab of this module.

14.4.1.5 Review

In this lab, you have:

- Started a PowerShell session in Azure Cloud Shell
- Created a resource group and an Azure managed disk by using Azure PowerShell
- Configured the managed disk by using Azure PowerShell

14.5 lab: title: '03d - Manage Azure resources by Using Azure CLI' module: 'Module 03 - Azure Administration'

15 Lab 03d - Manage Azure resources by Using Azure CLI

16 Student lab manual

16.1 Lab scenario

Now that you explored the basic Azure administration capabilities associated with provisioning resources and organizing them based on resource groups by using the Azure portal, Azure Resource Manager templates, and Azure PowerShell, you need to carry out the equivalent task by using Azure CLI. To avoid installing Azure CLI, you will leverage Bash environment available in Azure Cloud Shell.

16.2 Objectives

In this lab, you will:

- Task 1: Start a Bash session in Azure Cloud Shell
- Task 2: Create a resource group and an Azure managed disk by using Azure CLI
- Task 3: Configure the managed disk by using Azure CLI

16.3 Estimated timing: 20 minutes

16.4 Instructions

16.4.1 Exercise 1

16.4.1.1 Task 1: Start a Bash session in Azure Cloud Shell

In this task, you will open a Bash session in Cloud Shell.

1. From the portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.

2. If prompted to select either **Bash** or **PowerShell**, select **Bash**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

3. If prompted, click **Create storage**, and wait until the Azure Cloud Shell pane is displayed.

4. Ensure **Bash** appears in the drop-down menu in the upper-left corner of the Cloud Shell pane.

16.4.1.2 Task 2: Create a resource group and an Azure managed disk by using Azure CLI

In this task, you will create a resource group and an Azure managed disk by using Azure CLI session within Cloud Shell.

1. To create a resource group in the same Azure region as the **az104-03c-rg1** resource group you created in the previous lab, from the Bash session within Cloud Shell, run the following:

   ```
   LOCATION=$(az group show --name 'az104-03c-rg1' --query location --out tsv)

   RGNAME='az104-03d-rg1'

   az group create --name $RGNAME --location $LOCATION
   ```

2. To retrieve properties of the newly created resource group, run the following:

   ```
   az group show --name $RGNAME
   ```

3. To create a new managed disk with the same characteristics as those you created in the previous labs of this module, from the Bash session within Cloud Shell, run the following:

   ```
   DISKNAME='az104-03d-disk1'

   az disk create \
   --resource-group $RGNAME \
   --name $DISKNAME \
   --sku 'Standard_LRS' \
   --size-gb 32
   ```

   > **Note**: When using multi-line syntax, ensure that each line ends with back-slash (\) with no trailing spaces and that there are no leading spaces at the beginning of each line.

4. To retrieve properties of the newly created disk, run the following:

   ```
   az disk show --resource-group $RGNAME --name $DISKNAME
   ```

16.4.1.3 Task 3: Configure the managed disk by using Azure CLI

In this task, you will managing configuration of the Azure managed disk by using Azure CLI session within Cloud Shell.

1. To increase the size of the Azure managed disk to **64 GB**, from the Bash session within Cloud Shell, run the following:

   ```
   az disk update --resource-group $RGNAME --name $DISKNAME --size-gb 64
   ```

2. To verify that the change took effect, run the following:

   ```
   az disk show --resource-group $RGNAME --name $DISKNAME --query diskSizeGb
   ```

3. To change the disk performance SKU to **Premium_LRS**, from the Bash session within Cloud Shell, run the following:

   ```
   az disk update --resource-group $RGNAME --name $DISKNAME --sku 'Premium_LRS'
   ```

4. To verify that the change took effect, run the following:

   ```
   az disk show --resource-group $RGNAME --name $DISKNAME --query sku
   ```

16.4.1.4 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **Bash** shell session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   az group list --query "[?starts_with(name,'az104-03')].name" --output tsv
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   az group list --query "[?starts_with(name,'az104-03')].[name]" --output tsv | xargs -L1 bash -c 'a
   ```

   > **Note**: The command executes asynchronously (as determined by the --nowait parameter), so while you will be able to run another Azure CLI command immediately afterwards within the same Bash session, it will take a few minutes before the resource groups are actually removed.

16.4.1.5 Review

In this lab, you have:

- Started a Bash session in Azure Cloud Shell
- Created a resource group and an Azure managed disk by using Azure CLI
- Configured the managed disk by using Azure CLI

16.5 lab: title: '04 - Implement Virtual Networking' module: 'Module 04 - Virtual Networking'

17 Lab 04 - Implement Virtual Networking

18 Student lab manual

18.1 Lab scenario

You need to explore Azure virtual networking capabilities. To start, you plan to create a virtual network in Azure that will host a couple of Azure virtual machines. Since you intend to implement network-based segmentation, you will deploy them into different subnets of the virtual network. You also want to make sure that their private and public IP addresses will not change over time. To comply with Contoso security requirements, you need to protect public endpoints of Azure virtual machines accessible from Internet. Finally, you need to implement DNS name resolution for Azure virtual machines both within the virtual network and from Internet.

18.2 Objectives

In this lab, you will:

- Task 1: Create and configure a virtual network
- Task 2: Deploy virtual machines into the virtual network
- Task 3: Configure private and public IP addresses of Azure VMs
- Task 4: Configure network security groups
- Task 5: Configure Azure DNS for internal name resolution
- Task 6: Configure Azure DNS for external name resolution

18.3 Estimated timing: 40 minutes

18.4 Instructions

18.4.1 Exercise 1

18.4.1.1 Task 1: Create and configure a virtual network

In this task, you will create a virtual network with multiple subnets by using the Azure portal

1. Sign in to the Azure portal.
2. In the Azure portal, search for and select **Virtual networks**, and, on the **Virtual networks** blade, click **+ Add**.
3. Create a virtual network with 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	the name of a **new** resource group **az104-04-rg1**
Name	**az104-04-vnet1**
Region	the name of any Azure region available in the subscription you will use in this lab

4. Click **Next : IP Addresses** and enter the following values

Setting	Value
IPv4 address space	**10.40.0.0/20**

5. Click **+ Add subnet** enter the following values then click **Add**

Setting	Value
Subnet name	**subnet0**
Subnet address range	**10.40.0.0/24**

6. Accept the defaults and click **Review and Create**. Let validation occur, and hit **Create** again to submit your deployment.

   > **Note:** Wait for the virtual network to be provisioned. This should take less than a minute.

7. Click on **Go to resource**
8. On the **az104-04-vnet1** virtual network blade, click **Subnets** and then click **+ Subnet**.
9. Create a subnet with the following settings (leave others with their default values):

Setting	Value
Name	**subnet1**
Address range (CIDR block)	**10.40.1.0/24**
Network security group	**None**
Route table	**None**

10. Click **Save**

18.4.1.2 Task 2: Deploy virtual machines into the virtual network

In this task, you will deploy Azure virtual machines into different subnets of the virtual network by using an ARM template

1. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
2. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

3. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\04\az104-04-vms-loop-template.json** and **\Allfiles\Labs\04\az104-04-vms-loop-parameters.json** into the Cloud Shell home directory.

   > **Note**: You might need to upload each file separately.

4. From the Cloud Shell pane, run the following to deploy two virtual machines by using the template and parameter files you uploaded:

   ```
   $rgName = 'az104-04-rg1'

   New-AzResourceGroupDeployment `
      -ResourceGroupName $rgName `
      -TemplateFile $HOME/az104-04-vms-loop-template.json `
      -TemplateParameterFile $HOME/az104-04-vms-loop-parameters.json
   ```

   > **Note**: This method of deploying ARM templates uses Azure PowerShell. You can perform the same task by running the equivalent Azure CLI command **az deployment create** (for more information, refer to Deploy resources with Resource Manager templates and Azure CLI.

   > **Note**: Wait for the deployment to complete before proceeding to the next task. This should take about 2 minutes.

5. Close the Cloud Shell pane.

18.4.1.3 Task 3: Configure private and public IP addresses of Azure VMs

In this task, you will configure static assignment of public and private IP addresses assigned to network interfaces of Azure virtual machines.

> **Note**: Private and public IP addresses are actually assigned to the network interfaces, which, in turn are attached to Azure virtual machines, however, it is fairly common to refer to IP addresses assigned to Azure VMs instead.

1. In the Azure portal, search for and select **Resource groups**, and, on the **Resource groups** blade, click **az104-04-rg1**.
2. On the **az104-04-rg1** resource group blade, in the list of its resources, click **az104-04-vnet1**.
3. On the **az104-04-vnet1** virtual network blade, review the **Connected devices** section and verify that there are two network interfaces **az104-04-nic0** and **az104-04-nic1** attached to the virtual network.
4. Click **az104-04-nic0** and, on the **az104-04-nic0** blade, click **IP configurations**.

    > **Note**: Verify that **ipconfig1** is currently set up with a dynamic private IP address.

5. In the list IP configurations, click **ipconfig1**.
6. On the **ipconfig1** blade, in the **Public IP address settings** section, select **Associate**, click **+ Create new**, specify the following settings, and click **OK**:

Setting	Value
Name	**az104-04-pip0**
SKU	**Standard**

7. On the **ipconfig1** blade, set **Assignment** to **Static**, leave the default value of **IP address** set to **10.40.0.4**.
8. Back on the **ipconfig1** blade, save the changes.
9. Navigate back to the **az104-04-vnet1** blade
10. Click **az104-04-nic1** and, on the **az104-04-nic1** blade, click **IP configurations**.

    > **Note**: Verify that **ipconfig1** is currently set up with a dynamic private IP address.

11. In the list IP configurations, click **ipconfig1**.
12. On the **ipconfig1** blade, in the **Public IP address settings** section, select **Associate**, click **+ Create new**, specify the following settings, and click **OK**:

Setting	Value
Name	**az104-04-pip1**
SKU	**Standard**

13. On the **ipconfig1** blade, set **Assignment** to **Static**, leave the default value of **IP address** set to **10.40.1.4**.
14. Back on the **ipconfig1** blade, save the changes.
15. Navigate back to the **az104-04-rg1** resource group blade, in the list of its resources, click **az104-04-vm0**, and from the **az104-04-vm0** virtual machine blade, note the public IP address entry.
16. Navigate back to the **az104-04-rg1** resource group blade, in the list of its resources, click **az104-04-vm1**, and from the **az104-04-vm1** virtual machine blade, note the public IP address entry.

    > **Note**: You will need both IP addresses in the last task of this lab.

18.4.1.4 Task 4: Configure network security groups

In this task, you will configure network security groups in order to allow for restricted connectivity to Azure virtual machines.

1. In the Azure portal, navigate back to the **az104-04-rg1** resource group blade, and in the list of its resources, click **az104-04-vm0**.

2. On the **az104-04-vm0** overview blade, click **Connect**, click **RDP** in the drop-down menu, on the **Connect with RDP** blade, click **Download RDP File** using the Public IP address and follow the prompts to start the Remote Desktop session.

3. Note that the connection attempt fails.

    > **Note**: This is expected, because public IP addresses of the Standard SKU, by default, require that the network interfaces to which they are assigned are protected by a network security group. In order to allow Remote Desktop connections, you will create a network security group explicitly allowing inbound RDP traffic from Internet and assign it to network interfaces of both virtual machines.

4. In the Azure portal, search for and select **Network security groups**, and, on the **Network security groups** blade, click **+ Add**.

5. Create a network security group with 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	**az104-04-rg1**
Name	**az104-04-nsg01**
Region	the name of the Azure region where you deployed all other resources in this lab

6. Click **Review and Create**. Let validation occur, and hit **Create** to submit your deployment.

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

7. On the deployment blade, click **Go to resource** to open the **az104-04-nsg01** network security group blade.

8. On the **az104-04-nsg01** network security group blade, in the **Settings** section, click **Inbound security rules**.

9. Add an inbound rule with the following settings (leave others with their default values):

Setting	Value
Source	**Any**
Source port ranges	*
Destination	**Any**
Service	**RDP**
Action	**Allow**
Priority	**300**
Name	**AllowRDPInBound**

10. On the **az104-04-nsg01** network security group blade, in the **Settings** section, click **Network interfaces** and then click **+ Associate**.

11. Associate the **az104-04-nsg01** network security group with the **az104-04-nic0** and **az104-04-nic1** network interfaces.

    > **Note**: It may take up to 5 minutes for the rules from the newly created Network Security Group to be applied to the Network Interface Card.

12. Navigate back to the **az104-04-vm0** virtual machine blade.

    > **Note**: In the subsequent steps, you will verify that you can successfully connect to the target virtual machine and sign in by using the **Student** username and **Pa55w.rd1234** password.

13. On the **az104-04-vm0** blade, click **Connect**, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** using the Public IP address and follow the prompts to start the Remote Desktop session.

    > **Note**: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

> **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

14. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.

> **Note**: Leave the Remote Desktop session open. You will need it in the next task.

18.4.1.5 Task 5: Configure Azure DNS for internal name resolution

In this task, you will configure DNS name resolution within a virtual network by using Azure private DNS zones.

1. In the Azure portal, search for and select **Private DNS zones** and, on the **Private DNS zones** blade, click **+ Add**.
2. Create a private DNS zone with 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	**az104-04-rg1**
Name	**contoso.org**

3. Click Review and Create. Let validation occur, and hit Create again to submit your deployment.

> **Note**: Wait for the private DNS zone to be created. This should take about 2 minutes.

4. Click **Go to resource** to open the **contoso.org** DNS private zone blade.
5. On the **contoso.org** private DNS zone blade, in the **Settings** section, click **Virtual network links**
6. Click **+ Add** to create a virtual network link with the following settings (leave others with their default values):

Setting	Value
Link name	**az104-04-vnet1-link**
Subscription	the name of the Azure subscription you are using in this lab
Virtual network	**az104-04-vnet1**
Enable auto registration	enabled

7. Click **OK**.

> **Note:** Wait for the virtual network link to be created. This should take less than 1 minute.

8. On the **contoso.org** private DNS zone blade, in the sidebar, click **Overview**
9. Verify that the DNS records for **az104-04-vm0** and **az104-04-vm1** appear in the list of record sets as **Auto registered**.

> **Note:** You might need to wait a few minutes and refresh the page if the record sets are not listed.

10. Switch to the Remote Desktop session to **az104-04-vm0**, right-click the **Start** button and, in the right-click menu, click **Windows PowerShell (Admin)**.
11. In the Windows PowerShell console window, run the following to test internal name resolution of the **az104-04-vm1** DNS record set in the newly created private DNS zone:

```
nslookup az104-04-vm1.contoso.org
```

12. Verify that the output of the command includes the private IP address of **az104-04-vm1** (**10.40.1.4**).

18.4.1.6 Task 6: Configure Azure DNS for external name resolution

In this task, you will configure external DNS name resolution by using Azure public DNS zones.

1. In the web browser, open a new tab and navigate to https://www.godaddy.com/domains/domain-name-search.

2. Use the domain name search to identify a domain name which is not in use.
3. In the Azure portal, search for and select **DNS zones** and, on the **DNS zones** blade, click **+ Add**.
4. Create a DNS zone with 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	**az104-04-rg1**
Name	the DNS domain name you identified earlier in this task

5. Click Review and Create. Let validation occur, and hit Create again to submit your deployment.

    Note: Wait for the DNS zone to be created. This should take about 2 minutes.

6. Click **Go to resource** to open the blade of the newly created DNS zone.
7. On the DNS zone blade, click **+ Record set**.
8. Add a record set with the following settings (leave others with their default values):

Setting	Value
Name	**az104-04-vm0**
Type	**A**
Alias record set	**No**
TTL	**1**
TTL unit	**Hours**
IP address	the public IP address of **az104-04-vm0** which you identified in the third exercise of this lab

9. Click **OK**
10. On the DNS zone blade, click **+ Record set**.
11. Add a record set with the following settings (leave others with their default values):

Setting	Value
Name	**az104-04-vm1**
Type	**A**
Alias record set	**No**
TTL	**1**
TTL unit	**Hours**
IP address	the public IP address of **az104-04-vm1** which you identified in the third exercise of this lab

12. Click **OK**
13. On the DNS zone blade, note the name of the **Name server 1** entry.
14. In the Azure portal, open the **PowerShell** session in **Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
15. From the Cloud Shell pane, run the following to test external name resolution of the **az104-04-vm0** DNS record set in the the newly created DNS zone (replace the placeholder `[Name server 1]` with the name of **Name server 1** you noted earlier in this task and the `[domain name]` placeholder with the name of the DNS domain you created earlier in this task):

    ```
    nslookup az104-04-vm0.[domain name] [Name server 1]
    ```

16. Verify that the output of the command includes the public IP address of **az104-04-vm0**.
17. From the Cloud Shell pane, run the following to test external name resolution of the **az104-04-vm1** DNS record set in the the newly created DNS zone (replace the placeholder `[Name server 1]` with the name of **Name server 1** you noted earlier in this task and the `[domain name]` placeholder with the name of the DNS domain you created earlier in this task):

```
nslookup az104-04-vm1.[domain name] [Name server 1]
```

18. Verify that the output of the command includes the public IP address of **az104-04-vm1**.

18.4.1.7 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-04*'
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-04*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

18.4.1.8 Review

In this lab, you have:

- Created and configured a virtual network
- Deployed virtual machines into the virtual network
- Configured private and public IP addresses of Azure VMs
- Configured network security groups
- Configured Azure DNS for internal name resolution
- Configured Azure DNS for external name resolution

18.5 lab: title: '05 - Implement Intersite Connectivity' module: 'Module 05 - Intersite Connectivity'

19 Lab 05 - Implement Intersite Connectivity

20 Student lab manual

20.1 Lab scenario

Contoso has its datacenters in Boston, New York, and Seattle offices connected via a mesh wide-area network links, with full connectivity between them. You need to implement a lab environment that will reflect the topology of the Contoso's on-premises networks and verify its functionality.

20.2 Objectives

In this lab, you will:

- Task 1: Provision the lab environment
- Task 2: Configure local and global virtual network peering
- Task 3: Test intersite connectivity

20.3 Estimated timing: 30 minutes

20.3.1 Instructions

20.3.1.1 Task 1: Provision the lab environment

In this task, you will deploy three virtual machines, each into a separate virtual network, with two of them in the same Azure region and the third one in another Azure region.

1. Sign in to the Azure portal.
2. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\05\az104-05-vnetvm-loop-template.json** and **\Allfiles\Labs\05\az104-05-vnetvm-loop-parameters.json** into the Cloud Shell home directory.
5. From the Cloud Shell pane, run the following to create the resource group that will be hosting the lab environment. The first two virtual networks and a pair of virtual machines will be deployed in `[Azure_region_1]`. The third virtual network and the third virtual machine will be deployed in the same resource group but another `[Azure_region_2]`. (replace the `[Azure_region_1]` and `[Azure_region_2]` placeholder with the names of two different Azure regions where you intend to deploy these Azure virtual machines):

   ```
   $location1 = '[Azure_region_1]'

   $location2 = '[Azure_region_2]'

   $rgName = 'az104-05-rg1'

   New-AzResourceGroup -Name $rgName -Location $location1
   ```

   > **Note**: In order to identify Azure regions, from a PowerShell session in Cloud Shell, run **(Get-AzLocation).Location**

6. From the Cloud Shell pane, run the following to create the three virtual networks and deploy virtual machines into them by using the template and parameter files you uploaded:

   ```
   New-AzResourceGroupDeployment `
      -ResourceGroupName $rgName `
      -TemplateFile $HOME/az104-05-vnetvm-loop-template.json `
      -TemplateParameterFile $HOME/az104-05-vnetvm-loop-parameters.json `
      -location1 $location1 `
      -location2 $location2
   ```

   > **Note**: Wait for the deployment to complete before proceeding to the next step. This should take about 2 minutes.

7. Close the Cloud Shell pane.

20.3.1.2 Task 2: Configure local and global virtual network peering

In this task, you will configure local and global peering between the virtual networks you deployed in the previous tasks.

1. In the Azure portal, search for and select **Virtual networks**.
2. Review the virtual networks you created in the previous task and verify that the first two are located in the same Azure region and the third one in a different Azure region.

   > **Note**: The template you used for deployment of the three virtual networks ensures that the IP address ranges of the three virtual networks do not overlap.

3. In the list of virtual networks, click **az104-05-vnet0**.
4. On the **az104-05-vnet0** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
5. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	**az104-05-vnet0_to_az104-05-vnet1**
This virtual network: Traffic to remote virtual network	**Allow (default)**
This virtual network: Traffic forwarded from remote virtual network	**Block traffic that originates from outside this**
Virtual network gateway	**None**
Remote virtual network: Peering link name	**az104-05-vnet1_to_az104-05-vnet0**
Virtual network deployment model	**Resource manager**
I know my resource ID	unselected
Subscription	the name of the Azure subscription you are using in
Virtual network	**az104-05-vnet1**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this**
Virtual network gateway	**None**

> **Note**: This step establishes two local peerings - one from az104-05-vnet0 to az104-05-vnet1 and the other from az104-05-vnet1 to az104-05-vnet0.
>
> **Note**: In case you run into an issue with the Azure portal interface not displaying the virtual networks created in the previous task, you can configure peering by running the following PowerShell commands from Cloud Shell:

```
$rgName = 'az104-05-rg1'

$vnet0 = Get-AzVirtualNetwork -Name 'az104-05-vnet0' -ResourceGroupName $rgname

$vnet1 = Get-AzVirtualNetwork -Name 'az104-05-vnet1' -ResourceGroupName $rgname

Add-AzVirtualNetworkPeering -Name 'az104-05-vnet0_to_az104-05-vnet1' -VirtualNetwork $vnet0 -Remot

Add-AzVirtualNetworkPeering -Name 'az104-05-vnet1_to_az104-05-vnet0' -VirtualNetwork $vnet1 -Remot
```

6. On the **az104-05-vnet0** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
7. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	**az104-05-vnet0_to_az104-05-vnet2**
This virtual network: Traffic to remote virtual network	**Allow (default)**
This virtual network: Traffic forwarded from remote virtual network	**Block traffic that originates from outside this**
Virtual network gateway	**None**
Remote virtual network: Peering link name	**az104-05-vnet2_to_az104-05-vnet0**
Virtual network deployment model	**Resource manager**
I know my resource ID	unselected
Subscription	the name of the Azure subscription you are using in
Virtual network	**az104-05-vnet2**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this**
Virtual network gateway	**None**

> **Note**: This step establishes two global peerings - one from az104-05-vnet0 to az104-05-vnet2 and the other from az104-05-vnet2 to az104-05-vnet0.
>
> **Note**: In case you run into an issue with the Azure portal interface not displaying the virtual networks created in the previous task, you can configure peering by running the following PowerShell commands from Cloud Shell:

```
$rgName = 'az104-05-rg1'

$vnet0 = Get-AzVirtualNetwork -Name 'az104-05-vnet0' -ResourceGroupName $rgname

$vnet2 = Get-AzVirtualNetwork -Name 'az104-05-vnet2' -ResourceGroupName $rgname
```

```
Add-AzVirtualNetworkPeering -Name 'az104-05-vnet0_to_az104-05-vnet2' -VirtualNetwork $vnet0 -Remote

Add-AzVirtualNetworkPeering -Name 'az104-05-vnet2_to_az104-05-vnet0' -VirtualNetwork $vnet2 -Remote
```

8. Navigate back to the **Virtual networks** blade and, in the list of virtual networks, click **az104-05-vnet1**.
9. On the **az104-05-vnet1** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.
10. Add a peering with the following settings (leave others with their default values) and click **Add**:

Setting	Value
This virtual network: Peering link name	**az104-05-vnet1_to_az104-05-vnet2**
This virtual network: Traffic to remote virtual network	**Allow (default)**
This virtual network: Traffic forwarded from remote virtual network	**Block traffic that originates from outside this**
Virtual network gateway	**None**
Remote virtual network: Peering link name	**az104-05-vnet2_to_az104-05-vnet1**
Virtual network deployment model	**Resource manager**
I know my resource ID	unselected
Subscription	the name of the Azure subscription you are using in
Virtual network	**az104-05-vnet2**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this**
Virtual network gateway	**None**

> **Note**: This step establishes two global peerings - one from az104-05-vnet1 to az104-05-vnet2 and the other from az104-05-vnet2 to az104-05-vnet1.

> **Note**: In case you run into an issue with the Azure portal interface not displaying the virtual networks created in the previous task, you can configure peering by running the following PowerShell commands from Cloud Shell:

```
$rgName = 'az104-05-rg1'

$vnet1 = Get-AzVirtualNetwork -Name 'az104-05-vnet1' -ResourceGroupName $rgname

$vnet2 = Get-AzVirtualNetwork -Name 'az104-05-vnet2' -ResourceGroupName $rgname

Add-AzVirtualNetworkPeering -Name 'az104-05-vnet1_to_az104-05-vnet2' -VirtualNetwork $vnet1 -Remote

Add-AzVirtualNetworkPeering -Name 'az104-05-vnet2_to_az104-05-vnet1' -VirtualNetwork $vnet2 -Remote
```

20.3.1.3 Task 3: Test intersite connectivity

In this task, you will test connectivity between virtual machines on the three virtual networks that you connected via local and global peering in the previous task.

1. In the Azure portal, search for and select **Virtual machines**.
2. In the list of virtual machines, click **az104-05-vm0**.
3. On the **az104-05-vm0** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

   > **Note**: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

   > **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

4. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
5. Within the Remote Desktop session to **az104-05-vm0**, right-click the **Start** button and, in the right-click menu, click **Windows PowerShell (Admin)**.

6. In the Windows PowerShell console window, run the following to test connectivity to **az104-05-vm1** (which has the private IP address of **10.51.0.4**) over TCP port 3389:

   ```
   Test-NetConnection -ComputerName 10.51.0.4 -Port 3389 -InformationLevel 'Detailed'
   ```

   > **Note**: The test uses TCP 3389 since this is this port is allowed by default by operating system firewall.

7. Examine the output of the command and verify that the connection was successful.
8. In the Windows PowerShell console window, run the following to test connectivity to **az104-05-vm2** (which has the private IP address of **10.52.0.4**):

   ```
   Test-NetConnection -ComputerName 10.52.0.4 -Port 3389 -InformationLevel 'Detailed'
   ```

9. Switch back to the Azure portal on your lab computer and navigate back to the **Virtual machines** blade.
10. In the list of virtual machines, click **az104-05-vm1**.
11. On the **az104-05-vm1** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

    > **Note**: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

    > **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

12. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
13. Within the Remote Desktop session to **az104-05-vm1**, right-click the **Start** button and, in the right-click menu, click **Windows PowerShell (Admin)**.
14. In the Windows PowerShell console window, run the following to test connectivity to **az104-05-vm2** (which has the private IP address of **10.52.0.4**) over TCP port 3389:

    ```
    Test-NetConnection -ComputerName 10.52.0.4 -Port 3389 -InformationLevel 'Detailed'
    ```

    > **Note**: The test uses TCP 3389 since this is this port is allowed by default by operating system firewall.

15. Examine the output of the command and verify that the connection was successful.

20.3.1.4 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-05*'
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-05*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

20.3.1.5 Review

In this lab, you have:

- Provisioned the lab environment
- Configured local and global virtual network peering
- Tested intersite connectivity

20.4 lab: title: '06 - Implement Traffic Management' module: 'Module 06 - Network Traffic Management'

21 Lab 06 - Implement Traffic Management

22 Student lab manual

22.1 Lab scenario

You were tasked with testing managing network traffic targeting Azure virtual machines in the hub and spoke network topology, which Contoso considers implementing in its Azure environment (instead of creating the mesh topology, which you tested in the previous lab). This testing needs to include implementing connectivity between spokes by relying on user defined routes that force traffic to flow via the hub, as well as traffic distribution across virtual machines by using layer 4 and layer 7 load balancers. For this purpose, you intend to use Azure Load Balancer (layer 4) and Azure Application Gateway (layer 7).

> **Note**: This lab, by default, requires total of 8 vCPUs available in the Standard_Dsv3 series in the region you choose for deployment, since it involves deployment of four Azure VMs of Standard_D2s_v3 SKU. If your students are using trial accounts, with the limit of 4 vCPUs, you can use a VM size that requires only one vCPU (such as Standard_B1s).

22.2 Objectives

In this lab, you will:

- Task 1: Provision the lab environment
- Task 2: Configure the hub and spoke network topology
- Task 3: Test transitivity of virtual network peering
- Task 4: Configure routing in the hub and spoke topology
- Task 5: Implement Azure Load Balancer
- Task 6: Implement Azure Application Gateway

22.3 Estimated timing: 60 minutes

22.4 Instructions

22.4.1 Exercise 1

22.4.1.1 Task 1: Provision the lab environment

In this task, you will deploy four virtual machines into the same Azure region. The first two will reside in a hub virtual network, while each of the remaining two will reside in a separate spoke virtual network.

1. Sign in to the Azure portal.

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

3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\06\az104-06-vms-loop-template.json** and **\Allfiles\Labs\06\az104-06-vms-loop-parameters.json** into the Cloud Shell home directory.

5. From the Cloud Shell pane, run the following to create the first resource group that will be hosting the lab environment (replace the `[Azure_region]` placeholder with the name of an Azure region where you intend to deploy Azure virtual machines)(you can use the "(Get-AzLocation).Location" cmdlet to get the region list):

   ```
   $location = '[Azure_region]'

   $rgName = 'az104-06-rg1'
   ```

```
New-AzResourceGroup -Name $rgName -Location $location
```

6. From the Cloud Shell pane, run the following to create the three virtual networks and four Azure VMs into them by using the template and parameter files you uploaded:

```
New-AzResourceGroupDeployment `
   -ResourceGroupName $rgName `
   -TemplateFile $HOME/az104-06-vms-loop-template.json `
   -TemplateParameterFile $HOME/az104-06-vms-loop-parameters.json
```

> **Note**: Wait for the deployment to complete before proceeding to the next step. This should take about 5 minutes.

7. From the Cloud Shell pane, run the following to install the Network Watcher extension on the Azure VMs deployed in the previous step:

```
$rgName = 'az104-06-rg1'
$location = (Get-AzResourceGroup -ResourceGroupName $rgName).location
$vmNames = (Get-AzVM -ResourceGroupName $rgName).Name

foreach ($vmName in $vmNames) {
  Set-AzVMExtension `
  -ResourceGroupName $rgName `
  -Location $location `
  -VMName $vmName `
  -Name 'networkWatcherAgent' `
  -Publisher 'Microsoft.Azure.NetworkWatcher' `
  -Type 'NetworkWatcherAgentWindows' `
  -TypeHandlerVersion '1.4'
}
```

> **Note**: Wait for the deployment to complete before proceeding to the next step. This should take about 5 minutes.

8. Close the Cloud Shell pane.

22.4.1.2 Task 2: Configure the hub and spoke network topology

In this task, you will configure local peering between the virtual networks you deployed in the previous tasks in order to create a hub and spoke network topology.

1. In the Azure portal, search for and select **Virtual networks**.
2. Review the virtual networks you created in the previous task.

> **Note**: The template you used for deployment of the three virtual networks ensures that the IP address ranges of the three virtual networks do not overlap.

3. In the list of virtual networks, select **az104-06-vnet2**.
4. On the **az104-06-vnet2** blade, select **Properties**.
5. On the **az104-06-vnet2 | Properties** blade, record the value of the **Resource ID** property.
6. Navigate back to the list of virtual networks and select **az104-06-vnet3**.
7. On the **az104-06-vnet3** blade, select **Properties**.
8. On the **az104-06-vnet3 | Properties** blade, record the value of the **Resource ID** property.

> **Note**: You will need the values of the ResourceID property for both virtual networks later in this task.

> **Note**: This is a workaround that addresses the issue with the Azure portal occasionally not displaying the newly provisioned virtual network when creating virtual network peerings.

9. In the list of virtual networks, click **az104-06-vnet01**.
10. On the **az104-06-vnet01** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.

11. Add a peering with the following settings (leave others with their default values) and click Add:

Setting	Value
This virtual network: Peering link name	**az104-06-vnet01__to__az104-06-vnet2**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None (default)**
Remote virtual network: Peering link name	**az104-06-vnet2__to__az104-06-vnet01**
Virtual network deployment model	**Resource manager**
I know my resource ID	enabled
Resource ID	the value of resourceID parameter of **az104-06-vnet2** you recorded earlie
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Allow (default)**
Virtual network gateway	**None (default)**

> **Note**: Wait for the operation to complete.
>
> **Note**: This step establishes two local peerings - one from az104-06-vnet01 to az104-06-vnet2 and the other from az104-06-vnet2 to az104-06-vnet01.
>
> **Note**: **Allow forwarded traffic** needs to be enabled in order to facilitate routing between spoke virtual networks, which you will implement later in this lab.

12. On the **az104-06-vnet01** virtual network blade, in the **Settings** section, click **Peerings** and then click **+ Add**.

13. Add a peering with the following settings (leave others with their default values) and click Add:

Setting	Value
This virtual network: Peering link name	**az104-06-vnet01__to__az104-06-vnet3**
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Block traffic that originates from outside this virtual network**
Virtual network gateway	**None (default)**
Remote virtual network: Peering link name	**az104-06-vnet3__to__az104-06-vnet01**
Virtual network deployment model	**Resource manager**
I know my resource ID	enabled
Resource ID	the value of resourceID parameter of **az104-06-vnet3** you recorded earlie
Traffic to remote virtual network	**Allow (default)**
Traffic forwarded from remote virtual network	**Allow (default)**
Virtual network gateway	**None (default)**

> **Note**: This step establishes two local peerings - one from az104-06-vnet01 to az104-06-vnet3 and the other from az104-06-vnet3 to az104-06-vnet01. This completes setting up the hub and spoke topology (with two spoke virtual networks).
>
> **Note**: **Allow forwarded traffic** needs to be enabled in order to facilitate routing between spoke virtual networks, which you will implement later in this lab.

22.4.1.3 Task 3: Test transitivity of virtual network peering

In this task, you will test transitivity of virtual network peering by using Network Watcher.

1. In the Azure portal, search for and select **Network Watcher**.
2. On the **Network Watcher** blade, expand the listing of Azure regions and verify that the service is enabled in the Azure into which you deployed resources in the first task of this lab.
3. On the **Network Watcher** blade, navigate to the **Connection troubleshoot**.
4. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with 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	**az104-06-rg1**
Source type	**Virtual machine**
Virtual machine	**az104-06-vm0**
Destination	**Specify manually**
URI, FQDN or IPv4	**10.62.0.4**
Protocol	**TCP**
Destination Port	**3389**

> **Note**: **10.62.0.4** represents the private IP address of **az104-06-vm2**

5. Click **Check** and wait until results of the connectivity check are returned. Verify that the status is **Reachable**. Review the network path and note that the connection was direct, with no intermediate hops in between the VMs.

   > **Note**: This is expected, since the hub virtual network is peered directly with the first spoke virtual network.

6. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with 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	**az104-06-rg1**
Source type	**Virtual machine**
Virtual machine	**az104-06-vm0**
Destination	**Specify manually**
URI, FQDN or IPv4	**10.63.0.4**
Protocol	**TCP**
Destination Port	**3389**

> **Note**: **10.63.0.4** represents the private IP address of **az104-06-vm3**

7. Click **Check** and wait until results of the connectivity check are returned. Verify that the status is **Reachable**. Review the network path and note that the connection was direct, with no intermediate hops in between the VMs.

   > **Note**: This is expected, since the hub virtual network is peered directly with the second spoke virtual network.

8. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with 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	**az104-06-rg1**
Source type	**Virtual machine**
Virtual machine	**az104-06-vm2**
Destination	**Specify manually**
URI, FQDN or IPv4	**10.63.0.4**
Protocol	**TCP**
Destination Port	**3389**

9. Click **Check** and wait until results of the connectivity check are returned. Note that the status is **Unreachable**.

   > **Note**: This is expected, since the two spoke virtual networks are not peered with each other (virtual network peering is not transitive).

22.4.1.4 Task 4: Configure routing in the hub and spoke topology

In this task, you will configure and test routing between the two spoke virtual networks by enabling IP forwarding on the network interface of the **az104-06-vm0** virtual machine, enabling routing within its operating system, and configuring user-defined routes on the spoke virtual network.

1. In the Azure portal, search and select **Virtual machines**.
2. On the **Virtual machines** blade, in the list of virtual machines, click **az104-06-vm0**.
3. On the **az104-06-vm0** virtual machine blade, in the **Settings** section, click **Networking**.
4. Click the **az104-06-nic0** link next to the **Network interface** label, and then, on the **az104-06-nic0** network interface blade, in the **Settings** section, click **IP configurations**.
5. Set **IP forwarding** to **Enabled** and save the change.

    > **Note**: This setting is required in order for **az104-06-vm0** to function as a router, which will route traffic between two spoke virtual networks.

    > **Note**: Now you need to configure operating system of the **az104-06-vm0** virtual machine to support routing.

6. In the Azure portal, navigate back to the **az104-06-vm0** Azure virtual machine blade and click **Overview**.
7. On the **az104-06-vm0** blade, in the **Operations** section, click **Run command**, and, in the list of commands, click **RunPowerShellScript**.
8. On the **Run Command Script** blade, type the following and click **Run** to install the Remote Access Windows Server role.

    ```
    Install-WindowsFeature RemoteAccess -IncludeManagementTools
    ```

    > **Note**: Wait for the confirmation that the command completed successfully.

9. On the **Run Command Script** blade, type the following and click **Run** to install the Routing role service.

    ```
    Install-WindowsFeature -Name Routing -IncludeManagementTools -IncludeAllSubFeature

    Install-WindowsFeature -Name "RSAT-RemoteAccess-Powershell"

    Install-RemoteAccess -VpnType RoutingOnly

    Get-NetAdapter | Set-NetIPInterface -Forwarding Enabled
    ```

    > **Note**: Wait for the confirmation that the command completed successfully.

    > **Note**: Now you need to create and configure user defined routes on the spoke virtual networks.

10. In the Azure portal, search and select **Route tables** and, on the **Route tables** blade, click **+ Add**.
11. Create a route table with 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	**az104-06-rg1**
Location	the name of the Azure region in which you created the virtual networks
Name	**az104-06-rt23**
Propagate gateway routes	**No**

12. Click **Review and Create**. Let validation occur, and click **Create** to submit your deployment.

    > **Note**: Wait for the route table to be created. This should take about 3 minutes.

13. Back on the **Route tables** blade, click **Refresh** and then click **az104-06-rt23**.
14. On the **az104-06-rt23** route table blade, in the **Settings** section, click **Routes**, and then click **+ Add**.
15. Add a new route with the following settings (leave others with their default values):

Setting	Value
Route name	**az104-06-route-vnet2-to-vnet3**
Address prefix	**10.63.0.0/20**
Next hop type	**Virtual appliance**
Next hop address	**10.60.0.4**

16. Click **OK**
17. Back on the **az104-06-rt23** route table blade, in the **Settings** section, click **Subnets**, and then click **+ Associate**.
18. Associate the route table **az104-06-rt23** with the following subnet:

Setting	Value
Virtual network	**az104-06-vnet2**
Subnet	**subnet0**

19. Click **OK**
20. Navigate back to **Route tables** blade and click **+ Add**.
21. Create a route table with 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	**az104-06-rg1**
Region	the name of the Azure region in which you created the virtual networks
Name	**az104-06-rt32**
Propagate gateway routes	**No**

22. Click Review and Create. Let validation occur, and hit Create to submit your deployment.

    Note: Wait for the route table to be created. This should take about 3 minutes.

23. Back on the **Route tables** blade, click **Refresh** and then click **az104-06-rt32**.
24. On the **az104-06-rt32** route table blade, in the **Settings** section, click **Routes**, and then click **+ Add**.
25. Add a new route with the following settings:

Setting	Value
Route name	**az104-06-route-vnet3-to-vnet2**
Address prefix	**10.62.0.0/20**
Next hop type	**Virtual appliance**
Next hop address	**10.60.0.4**

26. Click **OK**
27. Back on the **az104-06-rt32** route table blade, in the **Settings** section, click **Subnets**, and then click **+ Associate**.
28. Associate the route table **az104-06-rt32** with the following subnet:

Setting	Value
Virtual network	**az104-06-vnet3**
Subnet	**subnet0**

29. Click **OK**

30. In the Azure portal, navigate back to the **Network Watcher - Connection troubleshoot** blade.
31. On the **Network Watcher - Connection troubleshoot** blade, initiate a check with 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	**az104-06-rg1**
Source type	**Virtual machine**
Virtual machine	**az104-06-vm2**
Destination	**Specify manually**
URI, FQDN or IPv4	**10.63.0.4**
Protocol	**TCP**
Destination Port	**3389**

32. Click **Check** and wait until results of the connectivity check are returned. Verify that the status is **Reachable**. Review the network path and note that the traffic was routed via **10.60.0.4**, assigned to the **az104-06-nic0** network adapter. If status is **Unreachable**, you should restart az104-06-vm0.

> **Note**: This is expected, since the traffic between spoke virtual networks is now routed via the virtual machine located in the hub virtual network, which functions as a router.

> **Note**: You can use **Network Watcher** to view topology of the network.

22.4.1.5 Task 5: Implement Azure Load Balancer

In this task, you will implement an Azure Load Balancer in front of the two Azure virtual machines in the hub virtual network

1. In the Azure portal, search and select **Load balancers** and, on the **Load balancers** blade, click **+ Add**.
2. Create a load balancer with 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	the name of a new resource group **az104-06-rg4**
Name	**az104-06-lb4**
Region	name of the Azure region into which you deployed all other resources in this lab
Type	**Public**
SKU	**Standard**
Public IP address	**Create new**
Public IP address name	**az104-06-pip4**
Availability zone	**No Zone**
Add a public IPv6 address	**No**

3. Click Review and Create. Let validation occur, and hit Create to submit your deployment.

> **Note**: Wait for the Azure load balancer to be provisioned. This should take about 2 minutes.

4. On the deployment blade, click **Go to resource**.
5. On the **az104-06-lb4** load balancer blade, in the **Settings** section, click **Backend pools**, and click **+ Add**.
6. Add a backend pool with the following settings (leave others with their default values):

Setting	Value
Name	**az104-06-lb4-be1**
Virtual network	**az104-06-vnet01**
IP version	**IPv4**
Virtual machine	**az104-06-vm0**
Virtual machine IP address	**ipconfig1 (10.60.0.4)**

Setting	Value
Virtual machine	**az104-06-vm1**
Virtual machine IP address	**ipconfig1 (10.60.1.4)**

7. Click **Add**
8. Wait for the backend pool to be created, in the **Settings** section, click **Health probes**, and then click **+ Add**.
9. Add a health probe with the following settings:

Setting	Value
Name	**az104-06-lb4-hp1**
Protocol	**TCP**
Port	**80**
Interval	**5**
Unhealthy threshold	**2**

10. Click **Add**
11. Wait for the health probe to be created, in the **Settings** section, click **Load balancing rules**, and then click **+ Add**.
12. Add a load balancing rule with the following settings (leave others with their default values):

Setting	Value
Name	**az104-06-lb4-lbrule1**
IP Version	**IPv4**
Protocol	**TCP**
Port	**80**
Backend port	**80**
Backend pool	**az104-06-lb4-be1**
Health probe	**az104-06-lb4-hp1**
Session persistence	**None**
Idle timeout (minutes)	**4**
TCP reset	**Disabled**
Floating IP (direct server return)	**Disabled**

13. Click **Add**
14. Wait for the load balancing rule to be created, click **Overview**, and note the value of the **Public IP address**.
15. Start another browser window and navigate to the IP address you identified in the previous step.
16. Verify that the browser window displays the message **Hello World from az104-06-vm0** or **Hello World from az104-06-vm1**.
17. Open another browser window but this time by using InPrivate mode and verify whether the target vm changes (as indicated by the message).

> **Note**: You might need to refresh the browser window or open it again by using InPrivate mode.

22.4.1.6 Task 6: Implement Azure Application Gateway

In this task, you will implement an Azure Application Gateway in front of the two Azure virtual machines in the spoke virtual networks.

1. In the Azure portal, search and select **Virtual networks**.
2. On the **Virtual networks** blade, in the list of virtual networks, click **az104-06-vnet01**.

3. On the **az104-06-vnet01** virtual network blade, in the **Settings** section, click **Subnets**, and then click **+ Subnet**.
4. Add a subnet with the following settings (leave others with their default values):

Setting	Value
Name	**subnet-appgw**
Subnet address range	**10.60.3.224/27**

5. Click **Save**

   Note: This subnet will be used by the Azure Application Gateway instances, which you will deploy later in this task. The Application Gateway requires a dedicated subnet of /27 or larger size.

6. In the Azure portal, search and select **Application Gateways** and, on the **Application Gateways** blade, click **+ Add**.
7. On the **Basics** tab of the **Create an application gateway** 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	the name of a new resource group **az104-06-rg5**
Application gateway name	**az104-06-appgw5**
Region	name of the Azure region into which you deployed all other resources in this lab
Tier	**Standard V2**
Enable autoscaling	**No**
HTTP2	**Disabled**
Virtual network	**az104-06-vnet01**
Subnet	**subnet-appgw**

8. Click **Next: Frontends >** and, on the **Frontends** tab of the **Create an application gateway** blade, click **Add new**, and specify the following settings (leave others with their default values):

Setting	Value
Frontend IP address type	**Public**
Firewall public IP address	the name of a new public ip address **az104-06-pip5**

9. Click **Next: Backends >**, on the **Backends** tab of the **Create an application gateway** blade, click **Add a backend pool**, and, on the **Add a backend pool** blade, specify the following settings (leave others with their default values):

Setting	Value
Name	**az104-06-appgw5-be1**
Add backend pool without targets	**No**
Target type	**IP address or FQDN**
Target	**10.62.0.4**
Target type	**IP address or FQDN**
Target	**10.63.0.4**

   Note: The targets represent the private IP addresses of virtual machines in the spoke virtual networks **az104-06-vm2** and **az104-06-vm3**.

10. Click **Add**, click **Next: Configuration >** and, on the **Configuration** tab of the **Create an application gateway** blade, click **+ Add a routing rule**.
11. On the **Add a routing rule** blade, on the **Listener** tab, specify the following settings:

Setting	Value
Rule name	**az104-06-appgw5-rl1**
Listener name	**az104-06-appgw5-rl1l1**
Frontend IP	**Public**
Protocol	**HTTP**
Port	**80**
Listener type	**Basic**
Error page url	**No**

12. Switch to the **Backend targets** tab of the **Add a routing rule** blade and specify the following settings (leave others with their default values):

Setting	Value
Target type	**Backend pool**
Backend target	**az104-06-appgw5-be1**

13. Click **Add new** under to the **HTTP setting** text box, and, on the **Add an HTTP setting** blade, specify the following settings (leave others with their default values):

Setting	Value
HTTP setting	**az104-06-appgw5-http1**
Backend protocol	**HTTP**
Backend port	**80**
Cookie-based affinity	**Disable**
Connection draining	**Disable**
Request time-out (seconds)	**20**

14. Click **Add** on the **Add an HTTP setting** blade, and back on the **Add a routing rule** blade, click **Add**.

15. Click **Next: Tags >**, followed by **Next: Review + create >** and then click **Create**.

    Note: Wait for the Application Gateway instance to be created. This might take about 8 minutes.

16. In the Azure portal, search and select **Application Gateways** and, on the **Application Gateways** blade, click **az104-06-appgw5**.

17. On the **az104-06-appgw5** Application Gateway blade, note the value of the **Frontend public IP address**.

18. Start another browser window and navigate to the IP address you identified in the previous step.

19. Verify that the browser window displays the message **Hello World from az104-06-vm2** or **Hello World from az104-06-vm3**.

20. Open another browser window but this time by using InPrivate mode and verify whether the target vm changes (based on the message displayed on the web page).

    Note: You might need to refresh the browser window or open it again by using InPrivate mode.

    Note: Targeting virtual machines on multiple virtual networks is not a common configuration, but it is meant to illustrate the point that Application Gateway is capable of targeting virtual machines on multiple virtual networks (as well as endpoints in other Azure regions or even outside of Azure), unlike Azure Load Balancer, which load balances across virtual machines in the same virtual network.

22.4.1.7 Clean up resources

Note: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-06*'
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-06*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

22.4.1.8 Review

In this lab, you have:

- Provisioned the lab environment
- Configured the hub and spoke network topology
- Tested transitivity of virtual network peering
- Task 4: Configure routing in the hub and spoke topology
- Task 5: Implement Azure Load Balancer
- Task 6: Implement Azure Application Gateway

22.5 lab: title: '07 - Manage Azure storage' module: 'Module 07 - Azure Storage'

23 Lab 07 - Manage Azure Storage

24 Student lab manual

24.1 Lab scenario

You need to evaluate the use of Azure storage for storing files residing currently in on-premises data stores. While majority of these files are not accessed frequently, there are some exceptions. You would like to minimize cost of storage by placing less frequently accessed files in lower-priced storage tiers. You also plan to explore different protection mechanisms that Azure Storage offers, including network access, authentication, authorization, and replication. Finally, you want to determine to what extent Azure Files service might be suitable for hosting your on-premises file shares.

24.2 Objectives

In this lab, you will:

- Task 1: Provision the lab environment
- Task 2: Create and configure Azure Storage accounts
- Task 3: Manage blob storage
- Task 4: Manage authentication and authorization for Azure Storage
- Task 5: Create and configure an Azure Files shares
- Task 6: Manage network access for Azure Storage

24.3 Estimated timing: 40 minutes

24.4 Instructions

24.4.1 Exercise 1

24.4.1.1 Task 1: Provision the lab environment

In this task, you will deploy an Azure virtual machine that you will use later in this lab.

1. Sign in to the Azure portal.

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

3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\07\az104-07-vm-template.json** and **\Allfiles\Labs\07\az104-07-vm-parameters.json** into the Cloud Shell home directory.

5. From the Cloud Shell pane, run the following to create the resource group that will be hosting the virtual machine (replace the `[Azure_region]` placeholder with the name of an Azure region where you intend to deploy the Azure virtual machine)

   > **Note**: To list the names of Azure regions, run `(Get-AzLocation).Location`

   ```
   $location = '[Azure_region]'

   $rgName = 'az104-07-rg0'

   New-AzResourceGroup -Name $rgName -Location $location
   ```

6. From the Cloud Shell pane, run the following to deploy the virtual machine by using the uploaded template and parameter files:

   ```
   New-AzResourceGroupDeployment `
      -ResourceGroupName $rgName `
      -TemplateFile $HOME/az104-07-vm-template.json `
      -TemplateParameterFile $HOME/az104-07-vm-parameters.json `
      -AsJob
   ```

   > **Note**: Do not wait for the deployments to complete, but proceed to the next task.

7. Close the Cloud Shell pane.

24.4.1.2 Task 2: Create and configure Azure Storage accounts

In this task, you will create and configure an Azure Storage account.

1. In the Azure portal, search for and select **Storage accounts**, and then click **+ Add**.

2. 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	the name of a **new** resource group **az104-07-rg1**
Storage account name	any globally unique name between 3 and 24 in length consisting of letters and digits
Location	the name of an Azure region where you can create an Azure Storage account
Performance	**Standard**
Redundancy	**Geo-redundant storage (GRS)**

3. Click **Next: Advanced >**, on the **Advanced** tab of the **Create storage account** blade, review the available options, accept the defaults, and click **Next: Networking >**.

4. On the **Networking** tab of the **Create storage account** blade, review the available options, accept the default option **Public endpoint (all networks}** and click **Next: Data protection >**.

5. On the **Data protection** tab of the **Create storage account** blade, review the available options, accept the defaults, 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.

6. On the deployment blade, click **Go to resource** to display the Azure Storage account blade.

7. On the Storage account blade, in the **Settings** section, click **Geo-replication** and note the secondary location.
8. Display again the **Configuration** blade of the Storage account, in the **Replication** drop-down list select **Locally redundant storage (LRS)** and save the change.
9. Switch back to the **Geo-replication** blade and note that, at this point, the Storage account has only the primary location.
10. Display again the **Configuration** blade of the Storage account and set **Access tier (default)** to **Cool**.

    Note: The cool access tier is optimal for data which is not accessed frequently.

24.4.1.3 Task 3: Manage blob storage

In this task, you will create a blob container and upload a blob into it.

1. On the Storage account blade, in the **Blob service** section, click **Containers**.
2. Click **+ Container** and create a container with the following settings:

Setting	Value
Name	**az104-07-container**
Public access level	**Private (no anonymous access)**

3. In the list of containers, click **az104-07-container** and then click **Upload**.
4. Browse to **\Allfiles\Labs\07\LICENSE** on your lab computer and click **Open**.
5. On the **Upload blob** blade, expand the **Advanced** section and specify the following settings (leave others with their default values):

Setting	Value
Authentication type	**Account key**
Blob type	**Block blob**
Block size	**4 MB**
Access tier	**Hot**
Upload to folder	**licenses**

    Note: Access tier can be set for individual blobs.

6. Click **Upload**.

    Note: Note that the upload automatically created a subfolder named **licenses**.

7. Back on the **az104-07-container** blade, click **licenses** and then click **LICENSE**.
8. On the **licenses/LICENSE** blade, review the available options.

    Note: You have the option to download the blob, change its access tier (it is currently set to **Hot**), acquire a lease, which would change its lease status to **Locked** (it is currently set to **Unlocked**) and protect the blob from being modified or deleted, as well as assign custom metadata (by specifying an arbitrary key and value pairs). You also have the ability to **Edit** the file directly within the Azure portal interface, without downloading it first. You can also create snapshots, as well as generate a SAS token (you will explore this option in the next task).

24.4.1.4 Task 4: Manage authentication and authorization for Azure Storage

In this task, you will configure authentication and authorization for Azure Storage.

1. On the **licenses/LICENSE** blade, on the **Overview** tab, click **Copy to clipboard** button next to the **URL** entry.
2. Open another browser window by using InPrivate mode and navigate to the URL you copied in the previous step.

3. You should be presented with an XML-formatted message stating **ResourceNotFound** or **PublicAccessNotPermitted**.

   > **Note**: This is expected, since the container you created has the public access level set to **Private (no anonymous access)**.

4. Close the InPrivate mode browser window, return to the browser window showing the **licenses/LICENSE** blade of the Azure Storage container, and switch to the the **Generate SAS** tab.

5. On the **Generate SAS** tab of the **licenses/LICENSE** blade, specify the following settings (leave others with their default values):

Setting	Value
Signing key	**Key 1**
Permissions	**Read**
Start date	yesterday's date
Start time	current time
Expiry date	tomorrow's date
Expiry time	current time
Allowed IP addresses	leave blank

6. Click **Generate SAS token and URL**.

7. Click **Copy to clipboard** button next to the **Blob SAS URL** entry.

8. Open another browser window by using InPrivate mode and navigate to the URL you copied in the previous step.

   > **Note**: If you are using Microsoft Edge, you should be presented with the **The MIT License (MIT)** page. If you are using Chrome, Microsoft Edge (Chromium) or Firefox, you should be able to view the content of the file by downloading it and opening it with Notepad.

   > **Note**: This is expected, since now your access is authorized based on the newly generated the SAS token.

   > **Note**: Save the blob SAS URL. You will need it later in this lab.

9. Close the InPrivate mode browser window, return to the browser window showing the **licenses/LICENSE** blade of the Azure Storage container, and from there, navigate back to the **az104-07-container** blade.

10. Click the **Switch to the Azure AD User Account** link next to the **Authentication method** label.

    > **Note**: You can see an error when you change the authentication method (the error is *"You do not have permissions to list the data using your user account with Azure AD"*). It is expected.

    > **Note**: At this point, you no longer have access to the container.

11. On the **az104-07-container** blade, click **Access Control (IAM)**.

12. In the **Add** section, click **Add a role assignment**.

13. On the **Add role assignment** blade, specify the following settings:

Setting	Value
Role	**Storage Blob Data Owner**
Assign access to	**User, group, or service principal**
Select	the name of your user account

14. Save the change and return to the **Overview** blade of the **az104-07-container** container and verify that you can access to container again.

    > **Note**: It might take about 5 minutes for the change to take effect.

24.4.1.5 Task 5: Create and configure an Azure Files shares

In this task, you will create and configure Azure Files shares.

Note: Before you start this task, verify that the virtual machine you provisioned in the first task of this lab is running.

1. In the Azure portal, navigate back to the blade of the storage account you created in the first task of this lab and, in the **File service** section, click **File shares**.
2. Click **+ File share** and create a file share with the following settings:

Setting	Value
Name	**az104-07-share**
Quota	**1024**

3. Click the newly created file share and click **Connect**.
4. On the **Connect** blade, ensure that the **Windows** tab is selected. Below you will find a grey textbox with a script, in the bottom right corner of that box hover over the pages icon and click **Copy to clipboard**.
5. In the Azure portal, search for and select **Virtual machines**, and, in the list of virtual machines, click **az104-07-vm0**.
6. On the **az104-07-vm0** blade, in the **Operations** section, click **Run command**.
7. On the **az104-07-vm0 - Run command** blade, click **RunPowerShellScript**.
8. On the **Run Command Script** blade, paste the script you copied earlier in this task into the **PowerShell Script** pane and click **Run**.
9. Verify that the script completed successfully.
10. Replace the content of the **PowerShell Script** pane with the following script and click **Run**:

```
New-Item -Type Directory -Path 'Z:\az104-07-folder'

New-Item -Type File -Path 'Z:\az104-07-folder\az-104-07-file.txt'
```

11. Verify that the script completed successfully.
12. Navigate back to the **az104-07-share** file share blade, click **Refresh**, and verify that **az104-07-folder** appears in the list of folders.
13. Click **az104-07-folder** and verify that **az104-07-file.txt** appears in the list of files.

24.4.1.6 Task 6: Manage network access for Azure Storage

In this task, you will configure network access for Azure Storage.

1. In the Azure portal, navigate back to the blade of the storage account you created in the first task of this lab and, in the **Security + Networking** section, click **Networking** and then click **Firewalls and virtual networks**.
2. Click the **Selected networks** option and review the configuration settings that become available once this option is enabled.

   > **Note**: You can use these settings to configure direct connectivity between Azure virtual machines on designated subnets of virtual networks and the storage account by using service endpoints.

3. Click the checkbox **Add your client IP address** and save the change.
4. Open another browser window by using InPrivate mode and navigate to the blob SAS URL you generated in the previous task.
5. You should be presented with the content of **The MIT License (MIT)** page.

   > **Note**: This is expected, since you are connecting from your client IP address.

6. Close the InPrivate mode browser window, return to the browser window showing the **licenses/LICENSE** blade of the Azure Storage container, and open Azure Cloud Shell pane.
7. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.

8. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.
9. From the Cloud Shell pane, run the following to attempt downloading of the LICENSE blob from the **az104-07-container** container of the storage account (replace the `[blob SAS URL]` placeholder with the blob SAS URL you generated in the previous task):

   ```
   Invoke-WebRequest -URI '[blob SAS URL]'
   ```

10. Verify that the download attempt failed.

    > **Note**: You should receive the message stating **AuthorizationFailure: This request is not authorized to perform this operation**. This is expected, since you are connecting from the IP address assigned to an Azure VM hosting the Cloud Shell instance.

11. Close the Cloud Shell pane.

24.4.1.7 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-07*'
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-07*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

24.4.1.8 Review

In this lab, you have:

- Provisioned the lab environment
- Created and configured Azure Storage accounts
- Managed blob storage
- Managed authentication and authorization for Azure Storage
- Created and configured an Azure Files shares
- Managed network access for Azure Storage

24.5 lab: title: '08 - Manage Virtual Machines' module: 'Module 08 - Virtual Machines'

25 Lab 08 - Manage Virtual Machines

26 Student lab manual

26.1 Lab scenario

You were tasked with identifying different options for deploying and configuring Azure virtual machines. First, you need to determine different compute and storage resiliency and scalability options you can implement when using Azure virtual machines. Next, you need to investigate compute and storage resiliency and scalability options that are available when using Azure virtual machine scale sets. You also want to explore the ability to automatically configure virtual machines and virtual machine scale sets by using the Azure Virtual Machine Custom Script extension.

26.2 Objectives

In this lab, you will:

- Task 1: Deploy zone-resilient Azure virtual machines by using the Azure portal and an Azure Resource Manager template
- Task 2: Configure Azure virtual machines by using virtual machine extensions
- Task 3: Scale compute and storage for Azure virtual machines
- Task 4: Register the Microsoft.Insights and Microsoft.AlertsManagement resource providers
- Task 5: Deploy zone-resilient Azure virtual machine scale sets by using the Azure portal
- Task 6: Configure Azure virtual machine scale sets by using virtual machine extensions
- Task 7: Scale compute and storage for Azure virtual machine scale sets (optional)

26.3 Estimated timing: 50 minutes

26.4 Instructions

26.4.1 Exercise 1

26.4.1.1 Task 1: Deploy zone-resilient Azure virtual machines by using the Azure portal and an Azure Resource Manager template

In this task, you will deploy Azure virtual machines into different availability zones by using the Azure portal and an Azure Resource Manager template.

1. Sign in to the Azure portal.

2. In the Azure portal, search for and select **Virtual machines** and, on the **Virtual machines** blade, click **+ Add**.

3. 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 la
Resource group	the name of a new resource group **az104-08-rg01**
Virtual machine name	**az104-08-vm0**
Region	select one of the regions that support availability zones and w
Availability options	**Availability zone**
Availability zone	**1**
Image	**Windows Server 2019 Datacenter - Gen1**
Azure Spot instance	**No**
Size	**Standard D2s v3**
Username	**Student**
Password	**Pa55w.rd1234**
Public inbound ports	**None**
Would you like to use an existing Windows Server license?	**No**

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

Setting	Value
OS disk type	**Premium SSD**
Enable Ultra Disk compatibility	**No**

5. Click **Next: Networking >** and, on the **Networking** tab of the **Create a virtual machine** blade, click **Create new** below the **Virtual network** textbox.

6. On the **Create virtual network** blade, specify the following settings (leave others with their default values):

Setting	Value
Name	**az104-08-rg01-vnet**
Address range	**10.80.0.0/20**
Subnet name	**subnet0**
Subnet range	**10.80.0.0/24**

7. Click **OK** and, back on the **Networking** tab of the **Create a virtual machine** blade, specify the following settings (leave others with their default values):

Setting	Value
Subnet	**subnet0**
Public IP	**default**
NIC network security group	**basic**
Public inbound Ports	**None**
Accelerated networking	**Off**
Place this virtual machine behind an existing load balancing solution?	**No**

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

Setting	Value
Boot diagnostics	**Enable with custom storage account**
Diagnostics storage account	accept the default value
Patch orchestration options	**Manual updates**

> **Note**: If necessary, select an existing storage account in the dropdown list. Record the name of the storage account. You will use it in the next task.

9. Click **Next: Advanced >**, on the **Advanced** tab of the **Create a virtual machine** blade, review the available settings without modifying any of them, and click **Review + Create**.

10. On the **Review + Create** blade, click **Create**.

11. On the deployment blade, click **Template**.

12. Review the template representing the deployment in progress and click **Deploy**.

> **Note**: You will use this option to deploy the second virtual machine with matching configuration except for the availability zone.

13. On the **Custom deployment** blade, specify the following settings (leave others with their default values):

Setting	Value
Resource group	**az104-08-rg01**
Network Interface Name	**az104-08-vm1-nic1**
Public IP Address Name	**az104-08-vm1-ip**
Virtual Machine Name	**az104-08-vm1**
Virtual Machine Computer Name	**az104-08-vm1**
Admin Username	**Student**
Admin Password	**Pa55w.rd1234**
Enable Hotpatching	**false**
Zone	**2**

> **Note**: You need to modify parameters corresponding to the properties of the distinct resources you are deploying by using the template, including the virtual machine and its network interface.

14. Click **Review + Create**, on the **Review + Create** blade, click **Create**.

> **Note**: Wait for both deployments to complete before you proceed to the next task. This might

take about 5 minutes.

26.4.1.2 Task 2: Configure Azure virtual machines by using virtual machine extensions

In this task, you will install Windows Server Web Server role on the two Azure virtual machines you deployed in the previous task by using the Custom Script virtual machine extension.

1. In the Azure portal, search for and select **Storage accounts** and, on the **Storage accounts** blade, click the entry representing the diagnostics storage account you created in the previous task.
2. On the storage account blade, in the **Blob service** section, click **Containers** and then click **+ Container**.
3. On the **New container** blade, specify the following settings (leave others with their default values) and click **Create**:

Setting	Value
Name	**scripts**
Public access level	**Private (no anonymous access)**

4. Back on the storage account blade displaying the list of containers, click **scripts**.
5. On the **scripts** blade, click **Upload**.
6. On the **Upload blob** blade, click the folder icon, in the **Open** dialog box, navigate to the **\Allfiles\Labs\08** folder, select **az104-08-install_IIS.ps1**, click **Open**, and back on the **Upload blob** blade, click **Upload**.
7. In the Azure portal, search for and select **Virtual machines** and, on the **Virtual machines** blade, click **az104-08-vm0**.
8. On the **az104-08-vm0** virtual machine blade, in the **Settings** section, click **Extensions**, and the click **+ Add**.
9. On the **New resource** blade, click **Custom Script Extension** and then click **Create**.
10. From the **Install extension** blade, click **Browse**.
11. On the **Storage accounts** blade, click the name of the storage account into which you uploaded the **az104-08-install_IIS.ps1** script, on the **Containers** blade, click **scripts**, on the **scripts** blade, click **az104-08-install_IIS.ps1**, and then click **Select**.
12. Back on the **Install extension** blade, click **OK**.
13. In the Azure portal, search for and select **Virtual machines** and, on the **Virtual machines** blade, click **az104-08-vm1**.
14. On the **az104-08-vm1** blade, in the **Automation** section, click **Export template**.
15. On the **az104-08-vm1 - Export template** blade, click **Deploy**.
16. On the **Custom deployment** blade, click **Edit template**.

    Note: Disregard the message stating **The resource group is in a location that is not supported by one or more resources in the template. Please choose a different resource group**. This is expected and can be ignored in this case.

17. On the **Edit template** blade, in the section displaying the content of the template, insert the following code starting with line **20** (directly underneath the `"resources": [` line):

    Note: If you are using a tool that pastes the code in line by line intellisense may add extra brackets causing validation errors. You may want to paste the code into notepad first and then paste it into line 20.

```
{
    "type": "Microsoft.Compute/virtualMachines/extensions",
    "name": "az104-08-vm1/customScriptExtension",
    "apiVersion": "2018-06-01",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "az104-08-vm1"
    ],
    "properties": {
        "publisher": "Microsoft.Compute",
        "type": "CustomScriptExtension",
        "typeHandlerVersion": "1.7",
        "autoUpgradeMinorVersion": true,
        "settings": {
            "commandToExecute": "powershell.exe Install-WindowsFeature -name Web-Server -Incl
      }
    }
},
```

> **Note**: This section of the template defines the same Azure virtual machine custom script extension that you deployed earlier to the first virtual machine via Azure PowerShell.

18. Click **Save** and, back on the **Custom template** blade, click **Review + Create** and, on the **Review + Create** blade, click **Create**

    > **Note**: Wait for the template deployment to complete. You can monitor its progress from the **Extensions** blade of the **az104-08-vm0** and **az104-08-vm1** virtual machines. This should take no more than 3 minutes.

19. To verify that the Custom Script extension-based configuration was successful, navigate back on the **az104-08-vm1** blade, in the **Operations** section, click **Run command**, and, in the list of commands, click **RunPowerShellScript**.

20. On the **Run Command Script** blade, type the following and click **Run** to access the web site hosted on **az104-08-vm0**:

    ```
    Invoke-WebRequest -URI http://10.80.0.4 -UseBasicParsing
    ```

    > **Note**: The **-UseBasicParsing** parameter is necessary to eliminate dependency on Internet Explorer to complete execution of the cmdlet

    > **Note**: You can also connect to **az104-08-vm0** and run `Invoke-WebRequest -URI http://10.80.0.5 -UseBasicParsing` to access the web site hosted on **az104-08-vm1**.

26.4.1.3 Task 3: Scale compute and storage for Azure virtual machines

In this task you will scale compute for Azure virtual machines by changing their size and scale their storage by attaching and configuring their data disks.

1. In the Azure portal, search for and select **Virtual machines** and, on the **Virtual machines** blade, click **az104-08-vm0**.

2. On the **az104-08-vm0** virtual machine blade, click **Size** and set the virtual machine size to **Standard DS1__v2** and click **Resize**

    > **Note**: Choose another size if **Standard DS1__v2** is not available.

3. On the **az104-08-vm0** virtual machine blade, click **Disks**, Under **Data disks** click **+ Create and attach a new disk**.

4. Create a managed disk with the following settings (leave others with their default values):

Setting	Value
Disk name	**az104-08-vm0-datadisk-0**
Storage type	**Premium SSD**
Size (GiB	**1024**

5. Back on the **az104-08-vm0 - Disks** blade, Under **Data disks** click **+ Create and attach a new disk**.

6. Create a managed disk with the following settings (leave others with their default values):

Setting	Value
Disk name	**az104-08-vm0-datadisk-1**
Storage type	**Premium SSD**
Size (GiB)	**1024 GiB**

7. Back on the **az104-08-vm0 - Disks** blade, click **Save**.

8. On the **az104-08-vm0** blade, in the **Operations** section, click **Run command**, and, in the list of commands, click **RunPowerShellScript**.

9. On the **Run Command Script** blade, type the following and click **Run** to create a drive Z: consisting of the two newly attached disks with the simple layout and fixed provisioning:

   ```
   New-StoragePool -FriendlyName storagepool1 -StorageSubsystemFriendlyName "Windows Storage*" -Physi

   New-VirtualDisk -StoragePoolFriendlyName storagepool1 -FriendlyName virtualdisk1 -Size 2046GB -Res

   Initialize-Disk -VirtualDisk (Get-VirtualDisk -FriendlyName virtualdisk1)

   New-Partition -DiskNumber 4 -UseMaximumSize -DriveLetter Z
   ```

   > **Note**: Wait for the confirmation that the commands completed successfully.

10. In the Azure portal, search for and select **Virtual machines** and, on the **Virtual machines** blade, click **az104-08-vm1**.

11. On the **az104-08-vm1** blade, in the **Automation** section, click **Export template**.

12. On the **az104-08-vm1 - Export template** blade, click **Deploy**.

13. On the **Custom deployment** blade, click **Edit template**.

    > **Note**: Disregard the message stating **The resource group is in a location that is not supported by one or more resources in the template. Please choose a different resource group**. This is expected and can be ignored in this case.

14. On the **Edit template** blade, in the section displaying the content of the template, replace the line **30** `"vmSize": "Standard_D2s_v3"` with the following line):

    ```
    "vmSize": "Standard_DS1_v2"
    ```

    > **Note**: This section of the template defines the same Azure virtual machine size as the one you specified for the first virtual machine via the Azure portal.

15. On the **Edit template** blade, in the section displaying the content of the template, replace line **50** (`"dataDisks": [ ]` line) with the following code :

    ```
    "dataDisks": [
      {
        "lun": 0,
        "name": "az104-08-vm1-datadisk0",
        "diskSizeGB": "1024",
        "caching": "ReadOnly",
        "createOption": "Empty"
      },
      {
        "lun": 1,
        "name": "az104-08-vm1-datadisk1",
        "diskSizeGB": "1024",
        "caching": "ReadOnly",
        "createOption": "Empty"
      }
    ]
    ```

> **Note**: If you are using a tool that pastes the code in line by line intellisense may add extra brackets causing validation errors. You may want to paste the code into notepad first and then paste it into line 49.

> **Note**: This section of the template creates two managed disks and attaches them to **az104-08-vm1**, similarly to the storage configuration of the first virtual machine via the Azure portal.

16. Click **Save** and, back on the **Custom template** blade, enable the checkbox **I agree to the terms and conditions stated above** and click **Purchase**.

    > **Note**: Wait for the template deployment to complete. You can monitor its progress from the **Disks** blade of the **az104-08-vm1** virtual machine. This should take no more than 3 minutes.

17. Back on the **az104-08-vm1** blade, in the **Operations** section, click **Run command**, and, in the list of commands, click **RunPowerShellScript**.

18. On the **Run Command Script** blade, type the following and click **Run** to create a drive Z: consisting of the two newly attached disks with the simple layout and fixed provisioning:

    ```
    New-StoragePool -FriendlyName storagepool1 -StorageSubsystemFriendlyName "Windows Storage*" -Physi

    New-VirtualDisk -StoragePoolFriendlyName storagepool1 -FriendlyName virtualdisk1 -Size 2046GB -Res

    Initialize-Disk -VirtualDisk (Get-VirtualDisk -FriendlyName virtualdisk1)

    New-Partition -DiskNumber 4 -UseMaximumSize -DriveLetter Z
    ```

    > **Note**: Wait for the confirmation that the commands completed successfully.

26.4.1.4 Task 4: Register the Microsoft.Insights and Microsoft.AlertsManagement resource providers

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

2. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

    > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

3. From the Cloud Shell pane, run the following to register the Microsoft.Insights and Microsoft.AlertsManagement resource providers.

    ```
    Register-AzResourceProvider -ProviderNamespace Microsoft.Insights

    Register-AzResourceProvider -ProviderNamespace Microsoft.AlertsManagement
    ```

26.4.1.5 Task 5: Deploy zone-resilient Azure virtual machine scale sets by using the Azure portal

In this task, you will deploy Azure virtual machine scale set across availability zones by using the Azure portal.

1. In the Azure portal, search for and select **Virtual machine scale sets** and, on the **Virtual machine scale sets** blade, click **+ Add**, click **+ Virtual machine**.

2. On the **Basics** tab of the **Create a virtual machine scale set** blade, specify the following settings (leave others with their default values) and click **Next : Disks >**:

Setting	Value
Subscription	the name of the Azure subscription you are using in this lab
Resource group	the name of a new resource group **az104-08-rg02**
Virtual machine scale set name	**az10408vmss0**
Region	select one of the regions that support availability zones and where you can provi
Availability zone	**Zones 1, 2, 3**
Image	**Windows Server 2019 Datacenter - Gen1**
Azure Spot instance	**No**

Setting	Value
Size	**Standard D2s_v3**
Username	**Student**
Password	**Pa55w.rd1234**
Already have a Windows Server license?	**No**

> **Note**: For the list of Azure regions which support deployment of Windows virtual machines to availability zones, refer to What are Availability Zones in Azure?

3. On the **Disks** tab of the **Create a virtual machine scale set** blade, accept the default values and click **Next : Networking >**.
4. On the **Networking** tab of the **Create a virtual machine scale set** blade, click the **Create virtual network** link below the **Virtual network** textbox and create a new virtual network with the following settings (leave others with their default values):

Setting	Value
Name	**az104-08-rg02-vnet**
Address range	**10.82.0.0/20**
Subnet name	**subnet0**
Subnet range	**10.82.0.0/24**

> **Note**: Once you create a new virtual network and return to the **Networking** tab of the **Create a virtual machine scale set** blade, the **Virtual network** value will be automatically set to **az104-08-rg02-vnet**.

5. Back on the **Networking** tab of the **Create a virtual machine scale set** blade, click the **Edit network interface** icon to the right of the network interface entry.
6. On the **Edit network interface** blade, in the **NIC network security group** section, click **Advanced** and click **Create new** under the **Configure network security group** drop-down list.
7. On the **Create network security group** blade, specify the following settings (leave others with their default values):

Setting	Value
Name	**az10408vmss0-nsg**

8. Click **Add an inbound rule** and add an inbound security rule with the following settings (leave others with their default values):

Setting	Value
Source	**Any**
Source port ranges	*****
Destination	**Any**
Destination port ranges	**80**
Protocol	**TCP**
Action	**Allow**
Priority	**1010**
Name	**custom-allow-http**

9. Click **Add** and, back on the **Create network security group** blade, click **OK**.
10. Back on the **Edit network interface** blade, in the **Public IP address** section, click **Enabled** and click **OK**.
11. Back on the **Networking** tab of the **Create a virtual machine scale set** blade, under the **Load balancing** section, ensure that the **Use a load balancer** entry is selected and specify the following **Load balancing settings** (leave others with their default values) and click **Next : Scaling >**:

Setting	Value
Load balancing options	**Azure load balancer**
Select a load balancer	**(new) az10408vmss0-lb**
Select a backend pool	**(new) bepool**

12. On the **Scaling** tab of the **Create a virtual machine scale set** blade, specify the following settings (leave others with their default values) and click **Next : Management >**:

Setting	Value
Initial instance count	**2**
Scaling policy	**Manual**

13. On the **Management** tab of the **Create a virtual machine scale set** blade, specify the following settings (leave others with their default values):

Setting	Value
Boot diagnostics	**Enable with custom storage account**
Diagnostics storage account	accept the default value

    Note: You will need the name of this storage account in the next task.

    Click **Next : Health >**:

14. On the **Health** tab of the **Create a virtual machine scale set** blade, review the default settings without making any changes and click **Next : Advanced >**.
15. On the **Advanced** tab of the **Create a virtual machine scale set** blade, specify the following settings (leave others with their default values) and click **Review + create**.

Setting	Value
Spreading algorithm	**Fixed spreading (not recommended with zones)**

    Note: The **Max spreading** setting is currently not functional.

16. On the **Review + create** tab of the **Create a virtual machine scale set** blade, ensure that the validation passed and click **Create**.

    Note: Wait for the virtual machine scale set deployment to complete. This should take about 5 minutes.

26.4.1.6 Task 6: Configure Azure virtual machine scale sets by using virtual machine extensions

In this task, you will install Windows Server Web Server role on the instances of the Azure virtual machine scale set you deployed in the previous task by using the Custom Script virtual machine extension.

1. In the Azure portal, search for and select **Storage accounts** and, on the **Storage accounts** blade, click the entry representing the diagnostics storage account you created in the previous task.
2. On the storage account blade, in the **Blob service** section, click **Containers** and then click **+ Container**.
3. On the **New container** blade, specify the following settings (leave others with their default values) and click **Create**:

Setting	Value
Name	**scripts**
Public access level	**Private (no anonymous access)**

4. Back on the storage account blade displaying the list of containers, click **scripts**.
5. On the **scripts** blade, click **Upload**.
6. On the **Upload blob** blade, click the folder icon, in the **Open** dialog box, navigate to the **\Allfiles\Labs\08** folder, select **az104-08-install_IIS.ps1**, click **Open**, and back on the **Upload blob** blade, click **Upload**.
7. In the Azure portal, navigate back to the **Virtual machine scale sets** blade and click **az10408vmss0**.
8. On the **az10408vmss0** blade, in the **Settings** section, click **Extensions**, and the click **+ Add**.
9. On the **New resource** blade, click **Custom Script Extension** and then click **Create**.
10. From the **Install extension** blade, **Browse** to and **Select** the **az104-08-install_IIS.ps1** script that was uploaded to the **scripts** container in the storage account earlier in this task, and then click **OK**.

    > **Note**: Wait for the installation of the extension to complete before proceeding to the next step.

11. In the **Settings** section of the **az10408vmss0** blade, click **Instances**, select the checkboxes next to the two instances of the virtual machine scale set, click **Upgrade**, and then, when prompted for confirmation, click **Yes**.

    > **Note**: Wait for the upgrade to complete before proceeding to the next step.

12. In the Azure portal, search for and select **Load balancers** and, in the list of load balancers, click **az10408vmss0-lb**.
13. On the **az10408vmss0-lb** blade, note the value of the **Public IP address** assigned to the frontend of the load balancer, open an new browser tab, and navigate to that IP address.

    > **Note**: Verify that the browser page displays the name of one of the instances of the Azure virtual machine scale set **az10408vmss0**.

26.4.1.7 Task 7: Scale compute and storage for Azure virtual machine scale sets

In this task, you will change the size of virtual machine scale set instances, configure their autoscaling settings, and attach disks to them.

1. In the Azure portal, search for and select **Virtual machine scale sets** and select the **az10408vmss0** scale set
2. In the **az10408vmss0** blade, in the **Settings** section, click **Size**.
3. In the list of available sizes, select **Standard DS1_v2** and click **Resize**.
4. In the **Settings** section, click **Instances**, select the checkboxes next to the two instances of the virtual machine scale set, click **Upgrade**, and then, when prompted for confirmation, click **Yes**.
5. In the list of instances, click the entry representing the first instance and, on the scale set instance blade, note its **Location** (it should be one of the zones in the target Azure region into which you deployed the Azure virtual machine scale set).
6. Return to the **az10408vmss0 - Instances** blade, click the entry representing the second instance and, on the scale set instance blade, note its **Location** (it should be one of the other two zones in the target Azure region into which you deployed the Azure virtual machine scale set).
7. Return to the **az10408vmss0 - Instances** blade, and in the **Settings** section, click **Scaling**.
8. On the **az10408vmss0 - Scaling** blade, select the **Custom autoscale** option and configure autoscale with the following settings (leave others with their default values):

Setting	Value
Scale mode	**Scale based on a metric**

9. Click the **+ Add a rule** link and, on the **Scale rule** blade, specify the following settings (leave others with their default values):

Setting	Value
Metric source	**Current resource (az10480vmss0)**
Time aggregation	**Average**
Metric namespace	**Virtual Machine Host**
Metric name	**Network In Total**
Operator	**Greater than**
Metric threshold to trigger scale action	**10**
Duration (in minutes)	**1**
Time grain statistic	**Average**
Operation	**Increase count by**
Instance count	**1**
Cool down (minutes)	**5**

> **Note**: Obviously these values do not represent a realistic configuration, since their purpose is to trigger autoscaling as soon as possible, without extended wait period.

10. Click **Add** and, back on the **az10408vmss0 - Scaling** blade, specify the following settings (leave others with their default values):

Setting	Value
Instance limits Minimum	**1**
Instance limits Maximum	**3**
Instance limits Default	**1**

11. Click **Save**.

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

13. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

14. From the Cloud Shell pane, run the following to identify the public IP address of the load balancer in front of the Azure virtual machine scale set **az10408vmss0**.

```
$rgName = 'az104-08-rg02'

$lbpipName = 'az10408vmss0-ip'

$pip = (Get-AzPublicIpAddress -ResourceGroupName $rgName -Name $lbpipName).IpAddress
```

15. From the Cloud Shell pane, run the following to start and infinite loop that sends the HTTP requests to the web sites hosted on the instances of Azure virtual machine scale set **az10408vmss0**.

```
while ($true) { Invoke-WebRequest -Uri "http://$pip" }
```

16. Minimize the Cloud Shell pane but do not close it, switch back to the **az10408vmss0 - Instances** blade and monitor the number of instances.

> **Note**: You might need to wait a couple of minutes and click **Refresh**.

17. Once the third instance is provisioned, navigate to its blade to determine its **Location** (it should be different than the first two zones you identified earlier in this task.

18. Close Cloud Shell pane.

19. On the **az10408vmss0** blade, in the **Settings** section, click **Disks**, click **+ Create and attach a new disk**, and attach a new managed disk with the following settings (leave others with their default values):

Setting	Value
LUN	**0**
Storage type	**Standard HDD**
Size (GiB)	**32**

20. Save the change, in the **Settings** section of the **az10408vmss0** blade, click **Instances**, select the checkboxes next to the two instances of the virtual machine scale set, click **Upgrade**, and then, when prompted for confirmation, click **Yes**.

    > **Note**: The disk attached in the previous step is a raw disk. Before it can be used, it is necessary to create a partition, create a filesystem, and mount it. To accomplish this, you will use Azure virtual machine Custom Script extension. First, you will need to remove the existing Custom Script Extension.

21. In the **Settings** section of the **az10408vmss0** blade, click **Extensions**, click **CustomScriptExtension**, and then click **Uninstall**.

    > **Note**: Wait for uninstallation to complete.

22. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
23. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.
24. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the file **\Allfiles\Labs\08\az104-08-configure_VMSS_disks.ps1** into the Cloud Shell home directory.
25. From the Cloud Shell pane, run the following to display the content of the script:

    ```
    Set-Location -Path $HOME

    Get-Content -Path ./az104-08-configure_VMSS_disks.ps1
    ```

    > **Note**: The script installs a custom script extension that configures the attached disk.

26. From the Cloud Shell pane, run the following to excecute the script and configure disks of Azure virtual machine scale set:

    ```
    ./az104-08-configure_VMSS_disks.ps1
    ```

27. Close the Cloud Shell pane.
28. In the **Settings** section of the **az10408vmss0** blade, click **Instances**, select the checkboxes next to the two instances of the virtual machine scale set, click **Upgrade**, and then, when prompted for confirmation, click **Yes**.

26.4.1.8 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. Remove az104-08-configure_VMSS_disks.ps1 by running the following command:

   ```
   rm ~\az104-08*
   ```

3. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-08*'
   ```

4. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-08*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

26.4.1.9 Review

In this lab, you have:

- Deployed zone-resilient Azure virtual machines by using the Azure portal and an Azure Resource Manager template
- Configured Azure virtual machines by using virtual machine extensions
- Scaled compute and storage for Azure virtual machines
- Deployed zone-reslient Azure virtual machine scale sets by using the Azure portal
- Configured Azure virtual machine scale sets by using virtual machine extensions
- Scaled compute and storage for Azure virtual machine scale sets

---

26.5 lab: title: '09a - Implement Web Apps' module: 'Module 09 - Serverless Computing'

27 Lab 09a - Implement Web Apps

28 Student lab manual

28.1 Lab scenario

You need to evaluate the use of Azure Web apps for hosting Contoso's web sites, hosted currently in the company's on-premises data centers. The web sites are running on Windows servers using PHP runtime stack. You also need to determine how you can implement DevOps practices by leveraging Azure web apps deployment slots.

28.2 Objectives

In this lab, you will:

- Task 1: Create an Azure web app
- Task 2: Create a staging deployment slot
- Task 3: Configure web app deployment settings
- Task 4: Deploy code to the staging deployment slot
- Task 5: Swap the staging slots
- Task 6: Configure and test autoscaling of the Azure web app

28.3 Estimated timing: 30 minutes

28.4 Instructions

28.4.1 Exercise 1

28.4.1.1 Task 1: Create an Azure web app

In this task, you will create an Azure web app.

1. Sign in to the **Azure portal**.

2. In the Azure portal, search for and select **App services**, and, on the **App Services** blade, click **+ Add**.

3. On the **Basics** tab of the **Web App** 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	the name of a new resource group **az104-09a-rg1**
Web app name	any globally unique name
Publish	**Code**
Runtime stack	**PHP 7.3**
Operating system	**Windows**
Region	the name of an Azure region where you can provision Azure web apps
App service plan	accept the default configuration

4. Click **Review + create**. On the **Review + create** tab of the **Create Web App** blade, ensure that

the validation passed and click **Create**.

> **Note**: Wait until the web app is created before you proceed to the next task. This should take about a minute.

5. On the deployment blade, click **Go to resource**.

28.4.1.2 Task 2: Create a staging deployment slot

In this task, you will create a staging deployment slot.

1. On the blade of the newly deployed web app, click the **URL** link to display the default web page in a new browser tab.
2. Close the new browser tab and, back in the Azure portal, in the **Deployment** section of the web app blade, click **Deployment slots**.

   > **Note**: The web app, at this point, has a single deployment slot labeled **PRODUCTION**.

3. Click **+ Add slot**, and add a new slot with the following settings:

Setting	Value
Name	**staging**
Clone settings from	**Do not clone settings**

4. Back on the **Deployment slots** blade of the web app, click the entry representing the newly created staging slot.

   > **Note**: This will open the blade displaying the properties of the staging slot.

5. Review the staging slot blade and note that its URL differs from the one assigned to the production slot.

28.4.1.3 Task 3: Configure web app deployment settings

In this task, you will configure web app deployment settings.

1. On the staging deployment slot blade, in the **Deployment** section, click **Deployment Center** and then select the **Settings** tab.

   > **Note:** Make sure you are on the staging slot blade (rather than the production slot).

2. On the **Settings** tab, in the **Source** drop-down list, select **Local Git** and click the **Save** button
3. On the **Deployment Center** blade, copy the **Git Clone Url** entry to Notepad.

   > **Note:** You will need the Git Clone Url value in the next task of this lab.

4. On the **Deployment Center** blade, select the **Local Git/FTPS credentials** tab, in the **User Scope** section, specify the following settings, and click **Save**.

Setting	Value
User name	any globally unique name (must not contain `@` character)
Password	any password that satisfies complexity requirements

> **Note:** The password must be at least eight characters long, with two of the following three elements: letters, numbers, and non-alphanumeric characters.

> **Note:** You will need these credentials in the next task of this lab.

28.4.1.4 Task 4: Deploy code to the staging deployment slot

In this task, you will deploy code to the staging deployment slot.

1. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
2. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

> **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

3. From the Cloud Shell pane, run the following to clone the remote repository containing the code for the web app.

   ```
   git clone https://github.com/Azure-Samples/php-docs-hello-world
   ```

4. From the Cloud Shell pane, run the following to set the current location to the newly created clone of the local repository containing the sample web app code.

   ```
   Set-Location -Path $HOME/php-docs-hello-world/
   ```

5. From the Cloud Shell pane, run the following to add the remote git (make sure to replace the `[deployment_user_name]` and `[git_clone_url]` placeholders with the value of the **Deployment Credentials** user name and **Git Clone Url**, respectively, which you identified in previous task):

   ```
   git remote add [deployment_user_name] [git_clone_url]
   ```

   > **Note**: The value following `git remote add` does not have to match the **Deployment Credentials** user name, but has to be unique

6. From the Cloud Shell pane, run the following to push the sample web app code from the local repository to the Azure web app staging deployment slot (make sure to replace the `[deployment_user_name]` placeholder with the value of the **Deployment Credentials** user name, which you identified in previous task):

   ```
   git push [deployment_user_name] master
   ```

7. If prompted to authenticate, type the `[deployment_user_name]` and the corresponding password (which you set in the previous task).
8. Close the Cloud Shell pane.
9. On the staging slot blade, click **Overview** and then click the **URL** link to display the default web page in a new browser tab.
10. Verify that the browser page displays the **Hello World!** message and close the new tab.

28.4.1.5 Task 5: Swap the staging slots

In this task, you will swap the staging slot with the production slot

1. Navigate back to the blade displaying the production slot of the web app.
2. In the **Deployment** section, click **Deployment slots** and then, click **Swap** toolbar icon.
3. On the **Swap** blade, review the default settings and click **Swap**.
4. Click **Overview** on the production slot blade of the web app and then click the **URL** link to display the web site home page in a new browser tab.
5. Verify the default web page has been replaced with the **Hello World!** page.

28.4.1.6 Task 6: Configure and test autoscaling of the Azure web app

In this task, you will configure and test autoscaling of Azure web app.

1. On the blade displaying the production slot of the web app, in the **Settings** section, click **Scale out (App Service plan)**.
2. Click **Custom autoscale**.

   > **Note**: You also have the option of scaling the web app manually.

3. Leave the default option **Scale based on a metric** selected and click **+ Add a rule**
4. On the **Scale rule** blade, specify the following settings (leave others with their default values):

Setting	Value
Metric source	**Current resource**

Setting	Value
Time aggregation	**Maximum**
Metric namespace	**App Service plans standard metrics**
Metric name	**CPU Percentage**
Operator	**Greater than**
Metric threshold to trigger scale action	**10**
Duration (in minutes)	**1**
Time grain statistic	**Maximum**
Operation	**Increase count by**
Instance count	**1**
Cool down (minutes)	**5**

> **Note**: Obviously these values do not represent a realistic configuration, since their purpose is to trigger autoscaling as soon as possible, without extended wait period.

5. Click **Add** and, back on the App Service plan scaling blade, specify the following settings (leave others with their default values):

Setting	Value
Instance limits Minimum	**1**
Instance limits Maximum	**2**
Instance limits Default	**1**

6. Click **Save**.
7. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
8. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.
9. From the Cloud Shell pane, run the following to identify the URL of the Azure web app.

```
$rgName = 'az104-09a-rg1'

$webapp = Get-AzWebApp -ResourceGroupName $rgName
```

10. From the Cloud Shell pane, run the following to start and infinite loop that sends the HTTP requests to the web app:

```
while ($true) { Invoke-WebRequest -Uri $webapp.DefaultHostName }
```

11. Minimize the Cloud Shell pane (but do not close it) and, on the web app blade, in the **Monitoring** section, click **Process explorer**.

> **Note**: Process explorer facilitates monitoring the number of instances and their resource utilization.

12. Monitor the utilization and the number of instances for a few minutes.

> **Note**: You may need to **Refresh** the page.

13. Once you notice that the number of instances has increased to 2, reopen the Cloud Shell pane and terminate the script by pressing **Ctrl+C**.
14. Close the Cloud Shell pane.

28.4.1.7 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

```
Get-AzResourceGroup -Name 'az104-09a*'
```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-09a*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

28.4.1.8 Review

In this lab, you have:

- Created an Azure web app
- Created a staging deployment slot
- Configured web app deployment settings
- Deployed code to the staging deployment slot
- Swapped the staging slots
- Configured and test autoscaling of the Azure web app

28.5 lab: title: '09b - Implement Azure Container Instances' module: 'Module 09 - Serverless Computing'

29 Lab 09b - Implement Azure Container Instances

30 Student lab manual

30.1 Lab scenario

Contoso wants to find a new platform for its virtualized workloads. You identified a number of container images that can be leveraged to accomplish this objective. Since you want to minimize container management, you plan to evaluate the use of Azure Container Instances for deployment of Docker images.

30.2 Objectives

In this lab, you will:

- Task 1: Deploy a Docker image by using the Azure Container Instance
- Task 2: Review the functionality of the Azure Container Instance

30.3 Estimated timing: 20 minutes

30.4 Instructions

30.4.1 Exercise 1

30.4.1.1 Task 1: Deploy a Docker image by using the Azure Container Instance

In this task, you will create a new container instance for the web application.

1. Sign in to the Azure portal.

2. In the Azure portal, search for locate **Container instances** and then, on the **Container instances** blade, click **+ New**.

3. On the **Basics** tab of the **Create container instance** 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	the name of a new resource group **az104-09b-rg1**
Container name	**az104-9b-c1**
Region	the name of a region where you can provision Azure container instances

Setting	Value
Image Source	**Quickstart images**
Image	**microsoft/aci-helloworld (Linux)**

4. Click **Next: Networking >** and, on the **Networking** tab of the **Create container instance** blade, specify the following settings (leave others with their default values):

Setting	Value
DNS name label	any valid, globally unique DNS host name

> **Note**: Your container will be publicly reachable at dns-name-label.region.azurecontainer.io. If you receive a **DNS name label not available** error message, specify a different value.

5. Click **Next: Advanced >**, review the settings on the **Advanced** tab of the **Create container instance** blade without making any changes, click **Review + Create**, ensure that the validation passed and click **Create**.

> **Note**: Wait for the deployment to complete. This should take about 3 minutes.
>
> **Note**: While you wait, you may be interested in viewing the code behind the sample application. To view it, browse the \app folder.

30.4.1.2 Task 2: Review the functionality of the Azure Container Instance

In this task, you will review the deployment of the container instance.

1. On the deployment blade, click the **Go to resource** link.
2. On the **Overview** blade of the container instance, verify that **Status** is reported as **Running**.
3. Copy the value of the container instance **FQDN**, open a new browser tab, and navigate to the corresponding URL.
4. Verify that the **Welcome to Azure Container Instance** page is displayed.
5. Close the new browser tab, back in the Azure portal, in the **Settings** section of the container instance blade, click **Containers**, and then click **Logs**.
6. Verify that you see the log entries representing the HTTP GET request generated by displaying the application in the browser.

30.4.1.3 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

```
Get-AzResourceGroup -Name 'az104-09b*'
```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

```
Get-AzResourceGroup -Name 'az104-09b*' | Remove-AzResourceGroup -Force -AsJob
```

> **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

30.4.1.4 Review

In this lab, you have:

- Deployed a Docker image by using the Azure Container Instance

- Reviewed the functionality of the Azure Container Instance

30.5 lab: title: '09c - Implement Azure Kubernetes Service' module: 'Module 09 - Serverless Computing'

31 Lab 09c - Implement Azure Kubernetes Service

32 Student lab manual

32.1 Lab scenario

Contoso has a number of multi-tier applications that are not suitable to run by using Azure Container Instances. In order to determine whether they can be run as containerized workloads, you want to evaluate using Kubernetes as the container orchestrator. To further minimize management overhead, you want to test Azure Kubernetes Service, including its simplified deployment experience and scaling capabilities.

32.2 Objectives

In this lab, you will:

- Task 1: Register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.
- Task 2: Deploy an Azure Kubernetes Service cluster
- Task 3: Deploy pods into the Azure Kubernetes Service cluster
- Task 4: Scale containerized workloads in the Azure Kubernetes service cluster

32.3 Estimated timing: 40 minutes

32.4 Instructions

32.4.1 Exercise 1

32.4.1.1 Task 1: Register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.

In this task, you will register resource providers necessary to deploy an Azure Kubernetes Services cluster.

1. Sign in to the Azure portal.

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

3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. From the Cloud Shell pane, run the following to register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.

   ```
   Register-AzResourceProvider -ProviderNamespace Microsoft.Kubernetes

   Register-AzResourceProvider -ProviderNamespace Microsoft.KubernetesConfiguration
   ```

5. Close the Cloud Shell pane.

32.4.1.2 Task 2: Deploy an Azure Kubernetes Service cluster

In this task, you will deploy an Azure Kubernetes Services cluster by using the Azure portal.

1. In the Azure portal, search for locate **Kubernetes services** and then, on the **Kubernetes services** blade, click **+ Add**, and then click **+ Add Kubernetes cluster**.

2. On the **Basics** tab of the **Create Kubernetes cluster** 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	the name of a new resource group **az104-09c-rg1**
Kubernetes cluster name	**az104-9c-aks1**
Region	the name of a region where you can provision a Kubernetes cluster
Kubernetes version	accept the default
Node size	accept the default
Node count	**1**

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

Setting	Value
Enable virtual nodes	**Disabled** (default)
Enable virtual machine scale sets	**Enabled** (default)

4. Click **Next: Authentication >** and, on the **Authentication** tab of the **Create Kubernetes cluster** blade, specify the following settings (leave others with their default values):

Setting	Value
Authentication method	**System-assigned managed identity** (default)
Role-based access control (RBAC)	**Enabled**

5. Click **Next: Networking >** and, on the **Networking** tab of the **Create Kubernetes cluster** blade, specify the following settings (leave others with their default values):

Setting	Value
Network configuration	**kubenet**
DNS name prefix	any valid, globally unique DNS host name

6. Click **Next: Integration >**, on the **Integration** tab of the **Create Kubernetes cluster** blade, set **Container monitoring** to **Disabled**, click **Review + create**, ensure that the validation passed and click Create.

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

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

32.4.1.3 Task 3: Deploy pods into the Azure Kubernetes Service cluster

In this task, you will deploy a pod into the Azure Kubernetes Service cluster.

1. On the deployment blade, click the **Go to resource** link.
2. On the **az104-9c-aks1** Kubernetes service blade, in the **Settings** section, click **Node pools**.
3. On the **az104-9c-aks1 - Node pools** blade, verify that the cluster consists of a single pool with one node.
4. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
5. Switch the **Azure Cloud Shell** to **Bash** (black background).
6. From the Cloud Shell pane, run the following to retrieve the credentials to access the AKS cluster:

```
RESOURCE_GROUP='az104-09c-rg1'
```

```
AKS_CLUSTER='az104-9c-aks1'

az aks get-credentials --resource-group $RESOURCE_GROUP --name $AKS_CLUSTER
```

7. From the **Cloud Shell** pane, run the following to verify connectivity to the AKS cluster:

   ```
   kubectl get nodes
   ```

8. In the **Cloud Shell** pane, review the output and verify that the one node which the cluster consists of at this point is reporting the **Ready** status.

9. From the **Cloud Shell** pane, run the following to deploy the **nginx** image from the Docker Hub:

   ```
   kubectl create deployment nginx-deployment --image=nginx
   ```

   > **Note**: Make sure to use lower case letters when typing the name of the deployment (nginx-deployment)

10. From the **Cloud Shell** pane, run the following to verify that a Kubernetes pod has been created:

    ```
    kubectl get pods
    ```

11. From the **Cloud Shell** pane, run the following to identify the state of the deployment:

    ```
    kubectl get deployment
    ```

12. From the **Cloud Shell** pane, run the following to make the pod available from Internet:

    ```
    kubectl expose deployment nginx-deployment --port=80 --type=LoadBalancer
    ```

13. From the **Cloud Shell** pane, run the following to identify whether a public IP address has been provisioned:

    ```
    kubectl get service
    ```

14. Re-run the command until the value in the **EXTERNAL-IP** column for the **nginx-deployment** entry changes from **<pending>** to a public IP address. Note the public IP address in the **EXTERNAL-IP** column for **nginx-deployment**.

15. Open a browser window and navigate to the IP address you obtained in the previous step. Verify that the browser page displays the **Welcome to nginx!** message.

32.4.1.4 Task 4: Scale containerized workloads in the Azure Kubernetes service cluster

In this task, you will scale horizontally the number of pods and then number of cluster nodes.

1. From the **Cloud Shell** pane, and run the following to scale the deployment by increasing of the number of pods to 2:

   ```
   RESOURCE_GROUP='az104-09c-rg1'

   AKS_CLUSTER='az104-9c-aks1'

   kubectl scale --replicas=2 deployment/nginx-deployment
   ```

2. From the **Cloud Shell** pane, run the following to verify the outcome of scaling the deployment:

   ```
   kubectl get pods
   ```

   > **Note**: Review the output of the command and verify that the number of pods increased to 2.

3. From the **Cloud Shell** pane, run the following to scale out the cluster by increasing the number of nodes to 2:

   ```
   az aks scale --resource-group $RESOURCE_GROUP --name $AKS_CLUSTER --node-count 2
   ```

   > **Note**: Wait for the provisioning of the additional node to complete. This might take about 3 minutes. If it fails, rerun the `az aks scale` command.

4. From the **Cloud Shell** pane, run the following to verify the outcome of scaling the cluster:

   ```
   kubectl get nodes
   ```

> **Note**: Review the output of the command and verify that the number of nodes increased to 2.

5. From the **Cloud Shell** pane, run the following to scale the deployment:

   ```
   kubectl scale --replicas=10 deployment/nginx-deployment
   ```

6. From the **Cloud Shell** pane, run the following to verify the outcome of scaling the deployment:

   ```
   kubectl get pods
   ```

   > **Note**: Review the output of the command and verify that the number of pods increased to 10.

7. From the **Cloud Shell** pane, run the following to review the pods distribution across cluster nodes:

   ```
   kubectl get pod -o=custom-columns=NODE:.spec.nodeName,POD:.metadata.name
   ```

   > **Note**: Review the output of the command and verify that the pods are distributed across both nodes.

8. From the **Cloud Shell** pane, run the following to delete the deployment:

   ```
   kubectl delete deployment nginx-deployment
   ```

9. Close the **Cloud Shell** pane.

32.4.1.5 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **Bash** shell session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   az group list --query "[?starts_with(name,'az104-09c')].name" --output tsv
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   az group list --query "[?starts_with(name,'az104-09c')].[name]" --output tsv | xargs -L1 bash -c '
   ```

   > **Note**: The command executes asynchronously (as determined by the --nowait parameter), so while you will be able to run another Azure CLI command immediately afterwards within the same Bash session, it will take a few minutes before the resource groups are actually removed.

32.4.1.6 Review

In this lab, you have:

- Deployed an Azure Kubernetes Service cluster
- Deployed pods into the Azure Kubernetes Service cluster
- Scaled containerized workloads in the Azure Kubernetes service cluster

---

32.5 lab: title: '10 - Implement Data Protection' module: 'Module 10 - Data Protection'

33 Lab 10 - Backup virtual machines

34 Student lab manual

34.1 Lab scenario

You have been tasked with evaluating the use of Azure Recovery Services for backup and restore of files hosted on Azure virtual machines and on-premises computers. In addition, you want to identify methods of protecting data stored in the Recovery Services vault from accidental or malicious data loss.

34.2 Objectives

In this lab, you will:

- Task 1: Provision the lab environment
- Task 2: Create a Recovery Services vault
- Task 3: Implement Azure virtual machine-level backup
- Task 4: Implement File and Folder backup
- Task 5: Perform file recovery by using Azure Recovery Services agent
- Task 6: Perform file recovery by using Azure virtual machine snapshots (optional)
- Task 7: Review the Azure Recovery Services soft delete functionality (optional)

34.3 Estimated timing: 50 minutes

34.4 Instructions

34.4.1 Exercise 1

34.4.1.1 Task 1: Provision the lab environment

In this task, you will deploy two virtual machines that will be used to test different backup scenarios.

1. Sign in to the Azure portal.

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

3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\10\az104-10-vms-template.json** and **\Allfiles\Labs\10\az104-10-vms-parameters.json** into the Cloud Shell home directory.

5. From the Cloud Shell pane, run the following to create the resource group that will be hosting the virtual machines (replace the `[Azure_region]` placeholder with the name of an Azure region where you intend to deploy Azure virtual machines):

   ```
   $location = '[Azure_region]'

   $rgName = 'az104-10-rg0'

   New-AzResourceGroup -Name $rgName -Location $location
   ```

6. From the Cloud Shell pane, run the following to create the first virtual network and deploy a virtual machine into it by using the template and parameter files you uploaded:

   ```
   New-AzResourceGroupDeployment `
      -ResourceGroupName $rgName `
      -TemplateFile $HOME/az104-10-vms-template.json `
      -TemplateParameterFile $HOME/az104-10-vms-parameters.json `
      -AsJob
   ```

7. Minimize Cloud Shell (but do not close it).

   > **Note**: Do not wait for the deployment to complete but instead proceed to the next task. The deployment should take about 5 minutes.

34.4.1.2 Task 2: Create a Recovery Services vault

In this task, you will create a recovery services vault.

1. In the Azure portal, search for and select **Recovery Services vaults** and, on the **Recovery Services vaults** blade, click **+ New**.

2. On the **Create Recovery Services vault** blade, specify the following settings:

Settings	Value
Subscription	the name of the Azure subscription you are using in this lab
Resource group	the name of a new resource group **az104-10-rg1**
Name	**az104-10-rsv1**
Region	the name of a region where you deployed the two virtual machines in the previous task

> **Note**: Make sure that you specify the same region into which you deployed virtual machines in the previous task.

3. Click **Review + Create**, ensure that the validation passed and click **Create**.

   > **Note**: Wait for the deployment to complete. The deployment should take less than 1 minute.

4. When the deployment is completed, click **Go to Resource**.
5. On the **az104-10-rsv1** Recovery Services vault blade, in the **Settings** section, click **Properties**.
6. On the **az104-10-rsv1 - Properties** blade, click the **Update** link under **Backup Configuration** label.
7. On the **Backup Configuration** blade, note that you can set the **Storage replication type** to either **Locally-redundant** or **Geo-redundant**. Leave the default setting of **Geo-redundant** in place and close the blade.

   > **Note**: This setting can be configured only if there are no existing backup items.

8. Back on the **az104-10-rsv1 - Properties** blade, click the **Update** link under **Security Settings** label.
9. On the **Security Settings** blade, note that **Soft Delete (For Azure Virtual Machines)** is **Enabled**.
10. Close the **Security Settings** blade and, back on the **az104-10-rsv1** Recovery Services vault blade, click **Overview**.

34.4.1.3 Task 3: Implement Azure virtual machine-level backup

In this task, you will implement Azure virtual-machine level backup.

> **Note**: Before you start this task, make sure that the deployment you initiated in the first task of this lab has successfully completed.

1. On the **az104-10-rsv1** Recovery Services vault blade, click **Overview**, then click **+ Backup**.
2. On the **Backup Goal** blade, specify the following settings:

Settings	Value
Where is your workload running?	**Azure**
What do you want to backup?	**Virtual machine**

3. On the **Backup Goal** blade, click **Backup**.
4. On the **Backup policy**, review the **DefaultPolicy** settings and select **Create a new policy**.
5. Define a new backup policy with the following settings (leave others with their default values):

Setting	Value
Policy name	**az104-10-backup-policy**
Frequency	**Daily**
Time	**12:00 AM**
Timezone	the name of your local time zone
Retain instant recovery snapshot(s) for	**2** Days(s)

6. Click **OK** to create the policy and then, in the **Virtual Machines** section, select **Add**.
7. On the **Select virtual machines** blade, select **az-104-10-vm0**, click **OK**, and, back on the **Backup** blade, click **Enable backup**.

Note: Wait for the backup to be enabled. This should take about 2 minutes.

8. Navigate back to the **az104-10-rsv1** Recovery Services vault blade, in the **Protected items** section, click **Backup items**, and then click the **Azure virtual machines** entry.
9. On the **Backup Items (Azure Virtual Machine)** blade of **az104-10-vm0**, review the values of the **Backup Pre-Check** and **Last Backup Status** entries, and click the **az104-10-vm0** entry.
10. On the **az104-10-vm0** Backup Item blade, click **Backup now**, accept the default value in the **Retain Backup Till** drop-down list, and click **OK**.

    Note: Do not wait for the backup to complete but instead proceed to the next task.

34.4.1.4 Task 4: Implement File and Folder backup

In this task, you will implement file and folder backup by using Azure Recovery Services.

1. In the Azure portal, search for and select **Virtual machines**, and on the **Virtual machines** blade, click **az104-10-vm1**.
2. On the **az104-10-vm1** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

    Note: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

    Note: You can ignore any warning prompts when connecting to the target virtual machines.

3. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
4. Within the Remote Desktop session to the **az104-10-vm1** Azure virtual machine, in the **Server Manager** window, click **Local Server**, click **IE Enhanced Security Configuration** and turn it **Off** for Administrators.
5. Within the Remote Desktop session to the **az104-10-vm1** Azure virtual machine, start a web browser, browse to the Azure portal, and sign in using your credentials.
6. In the Azure portal, search for and select **Recovery Services vaults** and, on the **Recovery Services vaults**, click **az104-10-rsv1**.
7. On the **az104-10-rsv1** Recovery Services vault blade, click **+ Backup**.
8. On the **Backup Goal** blade, specify the following settings:

Settings	Value
Where is your workload running?	**On-premises**
What do you want to backup?	**Files and folders**

    Note: Even though the virtual machine you are using in this task is running in Azure, you can leverage it to evaluate the backup capabilities applicable to any on-premises computer running Windows Server operating system.

9. On the **Backup Goal** blade, click **Prepare infrastructure**.
10. On the **Prepare infrastructure** blade, click the **Download Agent for Windows Server or Windows Client** link.
11. When prompted, click **Run** to start installation of **MARSAgentInstaller.exe** with the default settings.

    Note: On the **Microsoft Update Opt-In** page of the **Microsoft Azure Recovery Services Agent Setup Wizard**, select the **I do not want to use Microsoft Update** installation option.

12. On the **Installation** page of the **Microsoft Azure Recovery Services Agent Setup Wizard**, click **Proceed to Registration**. This will start **Register Server Wizard**.
13. Switch to the web browser window displaying the Azure portal, on the **Prepare infrastructure** blade, select the checkbox **Already downloaded or using the latest Recovery Server Agent**, and click **Download**.

14. When prompted, whether to open or save the vault credentials file, click **Save**. This will save the vault credentials file to the local Downloads folder.
15. Switch back to the **Register Server Wizard** window and, on the **Vault Identification** page, click **Browse**.
16. In the **Select Vault Credentials** dialog box, browse to the **Downloads** folder, click the vault credentials file you downloaded, and click **Open**.
17. Back on the **Vault Identification** page, click **Next**.
18. On the **Encryption Setting** page of the **Register Server Wizard**, click **Generate Passphrase**.
19. On the **Encryption Setting** page of the **Register Server Wizard**, click the **Browse** button next to the **Enter a location to save the passphrase** drop-down list.
20. In the **Browse For Folder** dialog box, select the **Documents** folder and click **OK**.
21. Click **Finish**, review the **Microsoft Azure Backup** warning and click **Yes**, and wait for the registration to complete.

    > **Note**: In a production environment, you should store the passphrase file in a secure location other than the server being backed up.

22. On the **Server Registration** page of the **Register Server Wizard**, review the warning regarding the location of the passphrase file, ensure that the **Launch Microsoft Azure Recovery Services Agent** checkbox is selected and click **Close**. This will automatically open the **Microsoft Azure Backup** console.
23. In the **Microsoft Azure Backup** console, in the **Actions** pane, click **Schedule Backup**.
24. In the **Schedule Backup Wizard**, on the **Getting started** page, click **Next**.
25. On the **Select Items to Backup** page, click **Add Items**.
26. In the **Select Items** dialog box, expand **C:\Windows\System32\drivers\etc\**, select **hosts**, and then click **OK**:
27. On the **Select Items to Backup** page, click **Next**.
28. On the **Specify Backup Schedule** page, ensure that the **Day** option is selected, in the first drop-down list box below the **At following times (Maximum allowed is three times a day)** box, select **4:30 AM**, and then click **Next**.
29. On the **Select Retention Policy** page, accept the defaults, and then click **Next**.
30. On the **Choose Initial Backup type** page, accept the defaults, and then click **Next**.
31. On the **Confirmation** page, click **Finish**. When the backup schedule is created, click **Close**.
32. In the **Microsoft Azure Backup** console, in the Actions pane, click **Back Up Now**.

    > **Note**: The option to run backup on demand becomes available once you create a scheduled backup.

33. In the Back Up Now Wizard, on the **Select Backup Item** page, ensure that the **Files and Folders** option is selected and click **Next**.
34. On the **Retain Backup Till** page, accept the default setting and click **Next**.
35. On the **Confirmation** page, click **Back Up**.
36. When the backup is complete, click **Close**, and then close Microsoft Azure Backup.
37. Switch to the web browser window displaying the Azure portal, navigate back to the **Recovery Services vault** blade, in the **Protected items** section, and click **Backup items**.
38. On the **az104-10-rsv1 - Backup items** blade, click **Azure Backup Agent**.
39. On the **Backup Items (Azure Backup Agent)** blade, verify that there is an entry referencing the **C:\** drive of **az104-10-vm1.**.

34.4.1.5 Task 5: Perform file recovery by using Azure Recovery Services agent (optional)

In this task, you will perform file restore by using Azure Recovery Services agent.

1. Within the Remote Desktop session to **az104-10-vm1**, open File Explorer, navigate to the **C:\Windows\System32\drivers\etc\** folder and delete the **hosts** file.
2. Open Microsoft Azure Backup and click **Recover data** in the **Actions** pane. This will start **Recover Data Wizard**.
3. On the **Getting Started** page of **Recover Data Wizard**, ensue that **This server (az104-10-vm1.)** option is selected and click **Next**.
4. On the **Select Recovery Mode** page, ensure that **Individual files and folders** option is selected, and click **Next**.
5. On the **Select Volume and Date** page, in the **Select the volume** drop down list, select **C:\**, accept the default selection of the available backup, and click **Mount**.

    > **Note**: Wait for the mount operation to complete. This might take about 2 minutes.

6. On the **Browse And Recover Files** page, note the drive letter of the recovery volume and review the tip regarding the use of robocopy.
7. Click **Start**, expand the **Windows System** folder, and click **Command Prompt**.
8. From the Command Prompt, run the following to copy the restore the **hosts** file to the original location (replace `[recovery_volume]` with the drive letter of the recovery volume you identified earlier):

    ```
    robocopy [recovery_volume]:\Windows\System32\drivers\etc C:\Windows\system32\drivers\etc hosts /r:
    ```

9. Switch back to the **Recover Data Wizard** and, on the **Browse and Recover Files**, click **Unmount** and, when prompted to confirm, click **Yes**.
10. Terminate the Remote Desktop session.

34.4.1.6 Task 6: Perform file recovery by using Azure virtual machine snapshots (optional)

In this task, you will restore a file from the Azure virtual machine-level snapshot-based backup.

1. Switch to the browser window running on your lab computer and displaying the Azure portal.
2. In the Azure portal, search for and select **Virtual machines**, and on the **Virtual machines** blade, click **az104-10-vm0**.
3. On the **az104-10-vm0** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

    > **Note**: This step refers to connecting via Remote Desktop from a Windows computer. On a Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

    > **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

4. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
5. Within the Remote Desktop session to the **az104-10-vm0** Azure virtual machine, in the **Server Manager** window, click **Local Server**, click **IE Enhanced Security Configuration** and turn it **Off** for Administrators.
6. Within the Remote Desktop session to the **az104-10-vm0**, click **Start**, expand the **Windows System** folder, and click **Command Prompt**.
7. From the Command Prompt, run the following to delete the **hosts** file:

    ```
    del C:\Windows\system32\drivers\etc\hosts
    ```

    > **Note**: You will restore this file from the Azure virtual machine-level snapshot-based backup later in this task.

8. Within the Remote Desktop session to the **az104-10-vm0** Azure virtual machine, start a web browser, browse to the Azure portal, and sign in using your credentials.

9. In the Azure portal, search for and select **Recovery Services vaults** and, on the **Recovery Services vaults**, click **az104-10-rsv1**.
10. On the **az104-10-rsv1** Recovery Services vault blade, in the **Protected items** section, click **Backup items**.
11. On the **az104-10-rsv1 - Backup items** blade, click **Azure Virtual Machine**.
12. On the **Backup Items (Azure Virtual Machine)** blade, click **az104-10-vm0**.
13. On the **az104-10-vm0** Backup Item blade, click **File Recovery**.

    Note: You have the option of running recovery shortly after backup starts based on the application consistent snapshot.
14. On the **File Recovery** blade, accept the default recovery point and click **Download Executable**.

    Note: The script mounts the disks from the selected recovery point as local drives within the operating system from which the script is run.
15. Click **Download** and, when prompted whether to run or save **IaaSVMILRExeForWindows.exe**, click **Save**.
16. Start File Explorer, navigate to the **Downloads** folder, right-click the newly downloaded file, select **Properties** in the right-click menu, in the **Properties** dialog box, select the **Unblock** checkbox, and click **OK**.
17. Back in the File Explorer window, double-click the newly downloaded file.
18. When prompted to provide the password from the portal, copy the password from the **Password to run the script** text box on the **File Recovery** blade, paste it at the Command Prompt, and press **Enter**.

    Note: This will open a Windows PowerShell window displaying the progress of the mount.

    Note: If you receive an error message at this point, refresh the web browser window and repeat the last three steps.
19. Wait for the mount process to complete, review the informational messages in the Windows PowerShell window, note the drive letter assigned to the volume hosting **Windows**, and start File Explorer.
20. In File Explorer, navigate to the drive letter hosting the snapshot of the operating system volume you identified in the previous step and review its content.
21. Switch to the **Command Prompt** window.
22. From the Command Prompt, run the following to copy the restore the **hosts** file to the original location (replace `[os_volume]` with the drive letter of the operating system volume you identified earlier):

    ```
    robocopy [os_volume]:\Windows\System32\drivers\etc C:\Windows\system32\drivers\etc hosts /r:1 /w:1
    ```
23. Switch back to the **File Recovery** blade in the Azure portal and click **Unmount Disks**.
24. Terminate the Remote Desktop session.

34.4.1.7 Task 7: Review the Azure Recovery Services soft delete functionality

1. On the lab computer, in the Azure portal, search for and select **Recovery Services vaults** and, on the **Recovery Services vaults**, click **az104-10-rsv1**.
2. On the **az104-10-rsv1** Recovery Services vault blade, in the **Protected items** section, click **Backup items**.
3. On the **az104-10-rsv1 - Backup items** blade, click **Azure Backup Agent**.
4. On the **Backup Items (Azure Backup Agent)** blade, click the entry representing the backup of **az104-10-vm1**.
5. On the **C:\ on az104-10-vm1.** blade, click the **az104-10-vm1.** link.
6. On the **az104-10-vm1.** Protected Servers blade, click **Delete**.
7. On the **Delete** blade, specify the following settings.

Settings	Value
TYPE THE SERVER NAME	**az104-10-vm1.**
Reason	**Recycling Dev/Test server**
Comments	**az104 10 lab**

Note: Make sure to include the trailing period when typing the server name

8. Enable the checkbox next to the label **There is backup data of 1 backup items associated with this server. I understand that clicking "Confirm" will permanently delete all the cloud backup data. This action cannot be undone. An alert may be sent to the administrators of this subscription notifying them of this deletion** and click **Delete**.
9. Navigate back to the **az104-10-rsv1 - Backup items** blade and click **Azure Virtual Machines**.
10. On the **az104-10-rsv1 - Backup items** blade, click **Azure Virtual Machine**.
11. On the **Backup Items (Azure Virtual Machine)** blade, click **az104-10-vm0**.
12. On the **az104-10-vm0** Backup Item blade, click **Stop backup**.
13. On the **Stop backup** blade, select **Delete Backup Data**, specify the following settings and click **Stop backup**:

Settings	Value
Type the name of Backup item	**az104-10-vm0**
Reason	**Others**
Comments	**az104 10 lab**

14. Navigate back to the **az104-10-rsv1 - Backup items** blade and click **Refresh**.

    Note: The **Azure Virtual Machine** entry is still lists **1** backup item.

15. Click the **Azure Virtual Machine** entry and, on the **Backup Items (Azure Virtual Machine)** blade, click the **az104-10-vm0** entry.
16. On the **az104-10-vm0** Backup Item blade, note that you have the option to **Undelete** the deleted backup.

    Note: This functionality is provided by the soft-delete feature, which is, by default, enabled for Azure virtual machine backups.

17. Navigate back to the **az104-10-rsv1** Recovery Services vault blade, and in the **Settings** section, click **Properties**.
18. On the **az104-10-rsv1 - Properties** blade, click the **Update** link under **Security Settings** label.
19. On the **Security Settings** blade, Disable **Soft Delete (For Azure Virtual Machines)** and click **Save**.

    Note: This will not affect items already in soft delete state.

20. Close the **Security Settings** blade and, back on the **az104-10-rsv1** Recovery Services vault blade, click **Overview**.
21. Navigate back to the **az104-10-vm0** Backup Item blade and click **Undelete**.
22. On the **Undelete az104-10-vm0** blade, click **Undelete**.
23. Wait for the undelete operation to complete, refresh the web browser page, if needed, navigate back to the **az104-10-vm0** Backup Item blade, and click **Delete backup data**.
24. On the **Delete Backup Data** blade, specify the following settings and click **Delete**:

Settings	Value
Type the name of Backup item	**az104-10-vm0**
Reason	**Others**
Comments	**az104 10 lab**

34.4.1.8 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-10*'
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-10*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: Optionally, you might consider deleting the auto-generated resource group with the prefix **AzureBackupRG_** (there is no additional charge associated with its existence).

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

34.4.1.9 Review

In this lab, you have:

- Provisioned the lab environment
- Created a Recovery Services vault
- Implemented Azure virtual machine-level backup
- Implemented File and Folder backup
- Performed file recovery by using Azure Recovery Services agent
- Performed file recovery by using Azure virtual machine snapshots
- Reviewed the Azure Recovery Services soft delete functionality

---

34.5 lab: title: '11 - Implement Monitoring' module: 'Module 11 - Monitoring'

35 Lab 11 - Implement Monitoring

36 Student lab manual

36.1 Lab scenario

You need to evaluate Azure functionality that would provide insight into performance and configuration of Azure resources, focusing in particular on Azure virtual machines. To accomplish this, you intend to examine the capabilities of Azure Monitor, including Log Analytics.

36.2 Objectives

In this lab, you will:

- Task 1: Provision the lab environment
- Task 2: Create and configure an Azure Log Analytics workspace and Azure Automation-based solutions
- Task 3: Review default monitoring settings of Azure virtual machines
- Task 4: Configure Azure virtual machine diagnostic settings
- Task 5: Review Azure Monitor functionality
- Task 6: Review Azure Log Analytics functionality

36.3 Estimated timing: 45 minutes

36.4 Instructions

36.4.1 Exercise 1

36.4.1.1 Task 1: Provision the lab environment

In this task, you will deploy a virtual machine that will be used to test monitoring scenarios.

1. Sign in to the Azure portal.
2. In the Azure portal, open the **Azure Cloud Shell** by clicking on the icon in the top right of the Azure Portal.
3. If prompted to select either **Bash** or **PowerShell**, select **PowerShell**.

   > **Note**: If this is the first time you are starting **Cloud Shell** and you are presented with the **You have no storage mounted** message, select the subscription you are using in this lab, and click **Create storage**.

4. In the toolbar of the Cloud Shell pane, click the **Upload/Download files** icon, in the drop-down menu, click **Upload** and upload the files **\Allfiles\Labs\11\az104-11-vm-template.json** and **\Allfiles\Labs\11\az104-11-vm-parameters.json** into the Cloud Shell home directory.
5. From the Cloud Shell pane, run the following to create the resource group that will be hosting the virtual machines (replace the `[Azure_region]` placeholder with the name of an Azure region where you intend to deploy Azure virtual machines):

   > **Note**: Make sure to choose one of the regions listed as **Log Analytics Workspace Region** in the referenced in Workspace mappings documentation

   ```
   $location = '[Azure_region]'

   $rgName = 'az104-11-rg0'

   New-AzResourceGroup -Name $rgName -Location $location
   ```

6. From the Cloud Shell pane, run the following to create the first virtual network and deploy a virtual machine into it by using the template and parameter files you uploaded:

   ```
   New-AzResourceGroupDeployment `
      -ResourceGroupName $rgName `
      -TemplateFile $HOME/az104-11-vm-template.json `
      -TemplateParameterFile $HOME/az104-11-vm-parameters.json `
      -AsJob
   ```

   > **Note**: Do not wait for the deployment to complete but instead proceed to the next task. The deployment should take about 3 minutes.

36.4.1.2 Task 2: Register the Microsoft.Insights and Microsoft.AlertsManagement resource providers.

1. From the Cloud Shell pane, run the following to register the Microsoft.Insights and Microsoft.AlertsManagement resource providers.

   ```
   Register-AzResourceProvider -ProviderNamespace Microsoft.Insights

   Register-AzResourceProvider -ProviderNamespace Microsoft.AlertsManagement
   ```

2. Minimize Cloud Shell pane (but do not close it).

36.4.1.3 Task 3: Create and configure an Azure Log Analytics workspace and Azure Automation-based solutions

In this task, you will create and configure an Azure Log Analytics workspace and Azure Automation-based solutions

1. In the Azure portal, search for and select **Log Analytics workspaces** and, on the **Log Analytics workspaces** blade, click **+ Add**.
2. On the **Basics** tab of the **Create Log Analytics workspace** blade, the following settings, click **Review + Create** and then click **Create**:

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

Settings	Value
Resource group	the name of a new resource group **az104-11-rg1**
Log Analytics Workspace	any unique name
Region	the name of the Azure region into which you deployed the virtual machine in the previous task

> **Note**: Make sure that you specify the same region into which you deployed virtual machines in the previous task.

> **Note**: Wait for the deployment to complete. The deployment should take about 1 minute.

3. In the Azure portal, search for and select **Automation Accounts**, and on the **Automation Accounts** blade, click **+ Add**.
4. On the **Add Automation Account** blade, specify the following settings, and click **Create**:

Settings	Value
Name	any unique name
Subscription	the name of the Azure subscription you are using in this lab
Resource group	**az104-11-rg1**
Location	the name of the Azure region determined based on Workspace mappings documentation
Create Azure Run As account	**Yes**

> **Note**: Make sure that you specify the Azure region based on the Workspace mappings documentation

> **Note**: Wait for the deployment to complete. The deployment might take about 3 minutes.

5. On the **Add Automation Account** blade, click **Refresh** and then click the entry representing your newly created Automation account.
6. On the Automation account blade, in the **Configuration Management** section, click **Inventory**.
7. In the **Inventory** pane, in the **Log Analytics workspace** drop-down list, select the Log Analytics workspace you created earlier in this task and click **Enable**.

> **Note**: Wait for the installation of the corresponding Log Analytics solution to complete. This might take about 3 minutes.

> **Note**: This automatically installs the **Change tracking** solution as well.

8. On the Automation account blade, in the **Update Management** section, click **Update management** and click **Enable**.

> **Note**: Wait for the installation to complete. This might take about 5 minutes.

36.4.1.4 Task 4: Review default monitoring settings of Azure virtual machines

In this task, you will review default monitoring settings of Azure virtual machines

1. In the Azure portal, search for and select **Virtual machines**, and on the **Virtual machines** blade, click **az104-11-vm0**.
2. On the **az104-11-vm0** blade, in the **Monitoring** section, click **Metrics**.
3. On the **az104-11-vm0 | Metrics** blade, on the default chart, note that the only available **Metrics Namespace** is **Virtual Machine Host**.

> **Note**: This is expected, since no guest-level diagnostic settings have been configured yet. You do have, however, the option of enabling guest memory metrics directly from the **Metrics Namespace** drop down-list. You will enable it later in this exercise.

4. In the **Metric** drop-down list, review the list of available metrics.

> **Note**: The list includes a range of CPU, disk, and network-related metrics that can be collected from the virtual machine host, without having access into guest-level metrics.

5. In the **Metric** drop-down list, select **Percentage CPU**, in the **Aggregation** drop-down list, select **Avg**, and review the resulting chart.

36.4.1.5 Task 5: Configure Azure virtual machine diagnostic settings

In this task, you will configure Azure virtual machine diagnostic settings.

1. On the **az104-11-vm0** blade, in the **Monitoring** section, click **Diagnostic settings**.
2. On the **Overview** tab of the **az104-11-vm0 | Diagnostic settings** blade, click **Enable guest-level monitoring**.

    > **Note**: Wait for the operation to take effect. This might take about 3 minutes.

3. Switch to the **Performance counters** tab of the **az104-11-vm0 | Diagnostic settings** blade and review the available counters.

    > **Note**: By default, CPU, memory, disk, and network counters are enabled. You can switch to the **Custom** view for more detailed listing.

4. Switch to the **Logs** tab of the **az104-11-vm0 | Diagnostic settings** blade and review the available event log collection options.

    > **Note**: By default, log collection includes critical, error, and warning entries from the Application Log and System log, as well as Audit failure entries from the Security log. Here as well you can switch to the **Custom** view for more detailed configuration settings.

5. On the **az104-11-vm0** blade, in the **Monitoring** section, click **Logs** and then click **Enable**.
6. On the **az104-11-vm0 - Logs** blade, ensure that the Log Analytics workspace you created earlier in this lab is selected in the **Choose a Log Analytics Workspace** drop-down list and click **Enable**.

    > **Note**: Do not wait for the operation to complete but instead proceed to the next step. The operation might take about 5 minutes.

7. On the **az104-11-vm0 | Logs** blade, in the **Monitoring** section, click **Metrics**.
8. On the **az104-11-vm0 | Metrics** blade, on the default chart, note that at this point, the **Metrics Namespace** drop-down list, in addition to the **Virtual Machine Host** entry includes also the **Guest (classic)** entry.

    > **Note**: This is expected, since you enabled guest-level diagnostic settings. You also have the option to **Enable new guest memory metrics**.

9. In the **Metrics Namespace** drop-down list, select the **Guest (classic)** entry.
10. In the **Metric** drop-down list, review the list of available metrics.

    > **Note**: The list includes additional guest-level metrics not available when relying on the host-level monitoring only.

11. In the **Metric** drop-down list, select **Memory\Available Bytes**, in the **Aggregation** drop-down list, select **Max**, and review the resulting chart.

36.4.1.6 Task 6: Review Azure Monitor functionality

1. In the Azure portal, search for and select **Monitor** and, on the **Monitor | Overview** blade, click **Metrics**.
2. On the **Select a scope** blade, on the **Browse** tab, navigate to the **az104-11-rg0** resource group, expand it, select the checkbox next to the **az104-11-vm0** virtual machine entry within that resource group, and click **Apply**.

    > **Note**: This gives you the same view and options as those available from the **az104-11-vm0 - Metrics** blade.

3. In the **Metric** drop-down list, select **Percentage CPU**, in the **Aggregation** drop-down list, select **Avg**, and review the resulting chart.
4. On the **Monitor | Metrics** blade, on the **Avg Percentage CPU for az104-11-vm0** pane, click **New alert rule**.

    > **Note**: Creating an alert rule from Metrics is not supported for metrics from the Guest (classic) metric namespace. This can be accomplished by using Azure Resource Manager templates, as

described in the document Send Guest OS metrics to the Azure Monitor metric store using a Resource Manager template for a Windows virtual machine

5. On the **Create alert rule** blade, in the **Condition** section, click the existing condition entry.
6. On the **Configure signal logic** blade, in the list of signals, in the **Alert logic** section, specify the following settings (leave others with their default values) and click **Done**:

Settings	Value
Threshold	**Static**
Operator	**Greater than**
Aggregation type	**Average**
Threshold value	**2**
Aggregation granularity (Period)	**1 minute**
Frequency of evaluation	**Every 1 Minute**

7. On the **Create alert rule** blade, in the **Action group** section, click **Add action groups** and then click the + **Create action group** button.
8. On the **Basics** tab of the **Create action group** blade, specify the following settings (leave others with their default values) and select **Next: Notifications >**:

Settings	Value
Subscription	the name of the Azure subscription you are using in this lab
Resource group	**az104-11-rg1**
Action group name	**az104-11-ag1**
Display name	**az104-11-ag1**

9. On the **Notifications** tab of the **Create action group** blade, in the **Notification type** drop-down list, select **Email/SMS/Push/Voice**. In the **Name** text box, type **admin email**. Click the **Edit details** (pencil) icon.
10. On the **Email/SMS/Push/Voice** blade, select the **Email** checkbox, type your email address in the **Email** textbox, leave others with their default values, click **OK**, back on the **Notifications** tab of the **Create action group** blade, select **Next: Actions >**.
11. On the **Actions** tab of the **Create action group** blade, review items available in the **Action type** drop-down list without making any changes and select **Review + create**.
12. On the **Review + create** tab of the **Create action group** blade, select **Create**.
13. Back on the **Create alert rule** blade, in the **Alert rule details** section, specify the following settings (leave others with their default values):

Settings	Value
Alert rule name	**CPU Percentage above the test threshold**
Description	**CPU Percentage above the test threshold**
Severity	**Sev 3**
Enable rule upon creation	**Yes**

14. Click **Create alert rule**.

    Note: It can take up to 10 minutes for a metric alert rule to become active.

15. In the Azure portal, search for and select **Virtual machines**, and on the **Virtual machines** blade, click **az104-11-vm0**.
16. On the **az104-11-vm0** blade, click **Connect**, in the drop-down menu, click **RDP**, on the **Connect with RDP** blade, click **Download RDP File** and follow the prompts to start the Remote Desktop session.

    Note: This step refers to connecting via Remote Desktop from a Windows computer. On a

Mac, you can use Remote Desktop Client from the Mac App Store and on Linux computers you can use an open source RDP client software.

> **Note**: You can ignore any warning prompts when connecting to the target virtual machines.

17. When prompted, sign in by using the **Student** username and **Pa55w.rd1234** password.
18. Within the Remote Desktop session, click **Start**, expand the **Windows System** folder, and click **Command Prompt**.
19. From the Command Prompt, run the following to trigger increased CPU utilization on the **az104-11-vm0** Azure VM:

    ```
    for /l %a in (0,0,1) do echo a
    ```

    > **Note**: This will initiate the infinite loop that should increase the CPU utilization above the threshold of the newly created alert rule.

20. Leave the Remote Desktop session open and switch back to the browser window displaying the Azure portal on your lab computer.
21. In the Azure portal, navigate back to the **Monitor** blade and click **Alerts**.
22. Note the number of **Sev 3** alerts and then click the **Sev 3** row.

    > **Note**: You might need to wait for a few minutes and click **Refresh**.

23. On the **All Alerts** blade, review generated alerts.

36.4.1.7 Task 7: Review Azure Log Analytics functionality

1. In the Azure portal, navigate back to the **Monitor** blade, click **Logs**.

   > **Note**: You might need to click **Get Started** if this is the first time you access Log Analytics.

2. If necessary, click **Select scope**, on the **Select a scope** blade, select the **Recent** tab, select **a104-11-vm0**, and click **Apply**.
3. In the query window, paste the following query, click **Run**, and review the resulting chart:

   ```
   // Virtual Machine available memory
   // Chart the VM's available memory over the last hour.
   InsightsMetrics
   | where TimeGenerated > ago(1h)
   | where Name == "AvailableMB"
   | project TimeGenerated, Name, Val
   | render timechart
   ```

4. Click **Queries** in the toolbar, on the **Queries** pane, locate the **Track VM availability** tile, click the **Run** command button in the tile, and review the results.
5. On the **New Query 1** tab, select the **Tables** header, and review the list of tables in the **Virtual machines** section.

   > **Note**: The names of several tables correspond to the solutions you installed earlier in this lab.

6. Hover the mouse over the **VMComputer** entry and click the **Preview data** icon.
7. If any data is available, in the **Update** pane, click **Use in editor**.

   > **Note**: You might need to wait a few minutes before the update data becomes available.

36.4.1.8 Clean up resources

> **Note**: Remember to remove any newly created Azure resources that you no longer use. Removing unused resources ensures you will not see unexpected charges.

1. In the Azure portal, open the **PowerShell** session within the **Cloud Shell** pane.
2. List all resource groups created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-11*'
   ```

3. Delete all resource groups you created throughout the labs of this module by running the following command:

   ```
   Get-AzResourceGroup -Name 'az104-11*' | Remove-AzResourceGroup -Force -AsJob
   ```

   > **Note**: The command executes asynchronously (as determined by the -AsJob parameter), so while you will be able to run another PowerShell command immediately afterwards within the same PowerShell session, it will take a few minutes before the resource groups are actually removed.

36.4.1.9 Review

In this lab, you have:

- Provisioned the lab environment
- Created and configured an Azure Log Analytics workspace and Azure Automation-based solutions
- Reviewed default monitoring settings of Azure virtual machines
- Configured Azure virtual machine diagnostic settings
- Reviewed Azure Monitor functionality
- Reviewed Azure Log Analytics functionality