Contents

What are we doing?

- We are publishing the lab instructions and lab files on GitHub to allow for interaction between the course authors and MCTs. We hope this will help keep the content current as the Azure platform changes.
- There is a GitHub repository for the course AZ-120, Planning and Administering Microsoft Azure for SAP Workloads.
- For each delivery, trainers should download the latest files from GitHub. Trainers should also check the Issues tab to see if other MCTs have reported any errors.
- Lab timing estimates are provided but trainers should check to ensure this is accurate based on the audience.
- The lab content has been placed at the end of each course for consistency and convenience. However, as the instructor, you are the best judge to determine when the lab should be offered.
- To conduct you will need an internet connection and an Azure subscription. Please read the Instructor Prep Guide for more information.
- It is recommended that you provide these materials directly to your students rather than point them to the GitHub repository.

How are we doing?

- If as you are teaching these courses, you identify areas for improvement, please use the Issues tab to provide feedback. We will periodically create new files to incorporate the changes.

We hope using this GitHub repository brings a sense of collaboration to the labs and improves the overall quality of the lab experience.

Regards, Azure Courseware Team

1 AZ 120 Module 1: Foundations of SAP on Azure

2 Lab 1a: Implementing Linux clustering on Azure VMs

Estimated Time: 90 minutes

All tasks in this lab are performed from the Azure portal (including the Bash Cloud Shell session)

> **Note**: When not using Cloud Shell, the lab virtual machine must have Azure CLI installed **https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-windows?view=azure-cli-latest** and include an SSH client e.g. PuTTY, available from **https://www.chiark.greenend.org.uk/~sgtatham/p**

Lab files: none

2.1 Scenario

In preparation for deployment of SAP HANA on Azure, Adatum Corporation wants to explore the process of implementing clustering on Azure VMs running the SUSE distribution of Linux.

2.2 Objectives

After completing this lab, you will be able to:

- Provision Azure compute resources necessary to support highly available SAP HANA deployments
- Configure operating system of Azure VMs running Linux to support a highly available SAP HANA installation
- Provision Azure network resources necessary to support highly available SAP HANA deployments

2.3 Requirements

- A Microsoft Azure subscription with the sufficient number of available DSv3 vCPUs (2 x 4) and DSv2 (1 x 1) vCPUs
- A lab computer with an Azure Cloud Shell-compatible web browser and access to Azure

2.4 Exercise 1: Provision Azure compute resources necessary to support highly available SAP HANA deployments

Duration: 30 minutes

In this exercise, you will deploy Azure infrastructure compute components necessary to configure Linux clustering. This will involve creating a pair of Azure VMs running Linux SUSE in the same availability set.

2.4.1 Task 1: Deploy Azure VMs running Linux SUSE

1. From the lab computer, start a Web browser, and navigate to the Azure portal at https://portal.azure.com
2. If prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you will be using for this lab.
3. In the Azure portal, use the **Search resources, services, and docs** text box at the top of the Azure portal page to search for and navigate to the **Proximity placement groups** blade and, on the **Proximity placement groups** blade, select **+ Add**.
4. On the **Basics** tab of the **Create Proximity Placement Groups** blade, specify the following settings and select **Review + create**:
   - Subscription: *the name of your Azure subscription*
   - Resource group: Resource group: *the name of a new resource group* **az12001a-RG**

     Note: Consider using **East US** or **East US2** regions for deployment of your resources.
   - Region: *an Azure region where you can deploy Azure VMs*
   - Proximity placement group name: **az12001a-ppg**
5. On the **Review + create** tab of the **Create Proximity Placement Groups** blade, select **Create**.

   Note: Wait for the provisioning to complete. This should take less than a minute.
6. In the Azure portal, use the **Search resources, services, and docs** text box at the top of the Azure portal page to search for and navigate to the **Virtual machines** blade, then, on the **Virtual machines** blade, select **+ Add** and, in the drop-down menu, select **Virtual machine**.
7. On the **Basics** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Disks >** (leave all other settings with their default value):
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of the resource group you used earlier in this task*
   - Virtual machine name: **az12001a-vm0**
   - Region: *the same Azure region you chose when creating the proximity placement group*
   - Availability options: **Availability set**
   - Availability set: *a new availability set named* **az12001a-avset** *with 2 fault domains and 5 update domains*
   - Image: **SUSE Enterprise Linux for SAP 12 SP5 - BYOS**

     Note: To locate the image, click the **See all images** link, on the **Select an image** blade, in the search text box, type **SUSE Enterprise Linux for SAP 12 BYOS** and, in the list of results, click **SUSE Enterprise Linux for SAP 12 SP5 - BYOS**.
   - Azure Spot Instance: **No**
   - Size: **Standard D4s v3**
   - Authentication type: **Password**

- Username: **student**
- Password: **Pa55w.rd1234**

8. On the **Disks** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Networking >** (leave all other settings with their default value):
   - OS disk type: **Premium SSD**
   - Encryption type: **(Default) Encryption at rest with a platform-managed key**
9. On the **Networking** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Management >** (leave all other settings with their default value):
   - Virtual network: *a new virtual network named* **az12001a-RG-vnet**
   - Address space: **192.168.0.0/20**
   - Subnet name: **subnet-0**
   - Subnet address range: **192.168.0.0/24**
   - Public IP address: *a new IP address named* **az12001a-vm0-ip**
   - NIC network security group: **Advanced**

     Note: This image has preconfigured NSG rules
   - Accelerated networking: **On**
   - Place this virtual machine behind an existing load balancing solutions: **No**
10. On the **Management** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Advanced >** (leave all other settings with their default value):
    - Enable basic plan for free: **No**

      Note: This setting is not available if you have already selected the Azure Security Center plan.
    - Boot diagnostics: **Enable with managed storage account (recommended)**
    - OS guest diagnostics: **Off**
    - System assigned managed identity: **Off**
    - Enable auto-shutdown: **Off**
11. On the **Advanced** tab of the **Create a virtual machine** blade, specify the following settings and select **Review + create** (leave all other settings with their default value):
    - Proximity placement group: **az12001a-ppg**
12. On the **Review + create** tab of the **Create Proximity Placement Groups** blade, select **Create**.

    Note: Wait for the provisioning to complete. This should take less about 3 minutes.
13. In the Azure portal, use the **Search resources, services, and docs** text box at the top of the Azure portal page to search for and navigate to the **Virtual machines** blade, then, on the **Virtual machines** blade, select **+ Add** and, in the drop-down menu, select **Virtual machine**.
14. On the **Basics** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Disks >** (leave all other settings with their default value):
    - Subscription: *the name of your Azure subscription*
    - Resource group: *the name of the resource group you used earlier in this task*
    - Virtual machine name: **az12001a-vm1**
    - Region: *the same Azure region you chose when creating the first Azure VM*
    - Availability options: **Availability set**
    - Availability set: **az12001a-avset**
    - Image: **SUSE Enterprise Linux for SAP 12 SP5 - BYOS**

Note: To locate the image, click the **See all images** link, on the **Select an image** blade, in the search text box, type **SUSE Enterprise Linux for SAP 12 BYOS** and, in the list of results, click **SUSE Enterprise Linux for SAP 12 SP5 - BYOS**.

- Azure Spot Instance: **No**
- Size: **Standard D4s v3**
- Authentication type: **Password**
- Username: **student**
- Password: **Pa55w.rd1234**

15. On the **Disks** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Networking >** (leave all other settings with their default value):
    - OS disk type: **Premium SSD**
    - Encryption type: **(Default) Encryption at rest with a platform-managed key**
16. On the **Networking** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Management >** (leave all other settings with their default value):
    - Virtual network: **az12001a-RG-vnet**
    - Subnet: **subnet-0 (192.168.0.0/24)**
    - Public IP address: *a new IP address named* **az12001a-vm1-ip**
    - NIC network security group: **Advanced**

      Note: This image has preconfigured NSG rules
    - Accelerated networking: **On**
    - Place this virtual machine behind an existing load balancing solutions: **No**
17. On the **Management** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Advanced >** (leave all other settings with their default value):
    - Enable basic plan for free: **No**

      Note: This setting is not available if you have already selected the Azure Security Center plan.
    - Boot diagnostics: **Enable with managed storage account (recommended)**
    - OS guest diagnostics: **Off**
    - System assigned managed identity: **Off**
    - Enable auto-shutdown: **Off**
18. On the **Advanced** tab of the **Create a virtual machine** blade, specify the following settings and select **Review + create** (leave all other settings with their default value):
    - Proximity placement group: **az12001a-ppg**
19. On the **Review + create** tab of the **Create Proximity Placement Groups** blade, select **Create**.

    Note: Wait for the provisioning to complete. This should take less about 3 minutes.

2.4.2 Task 2: Create and configure Azure VMs disks

1. In the Azure Portal, start a Bash session in Cloud Shell.

   Note: If this is the first time you are launching Cloud Shell in the current Azure subscription, you will be asked to create an Azure file share to persist Cloud Shell files. If so, accept the defaults, which will result in creation of a storage account in an automatically generated resource group.
2. In the Cloud Shell pane, run the following command to set the value of the variable `RESOURCE_GROUP_NAME` to the name of the resource group containing the resources you provisioned in the previous task:

```
RESOURCE_GROUP_NAME='az12001a-RG'
```

3. In the Cloud Shell pane, run the following command to create the first set of 8 managed disks that you will attach to the first Azure VM you deployed in the previous task:

```
LOCATION=$(az group list --query "[?name == '$RESOURCE_GROUP_NAME'].location" --output tsv)

for I in {0..7}; do az disk create --resource-group $RESOURCE_GROUP_NAME --name az12001a-vm0-DataD
```

4. In the Cloud Shell pane, run the following command to create the second set of 8 managed disks that you will attach to the second Azure VM you deployed in the previous task:

```
for I in {0..7}; do az disk create --resource-group $RESOURCE_GROUP_NAME --name az12001a-vm1-DataD
```

5. In the Azure portal, navigate to the blade of the first Azure VM you provisioned in the previous task (**az12001a-vm0**).
6. From the **az12001a-vm0** blade, navigate to the **az12001a-vm0 | Disks** blade.
7. On the **az12001a-vm0 | Disks** blade, select **Attach existing disks** and attach data disk with the following settings to az12001a-vm0:
   - LUN: **0**
   - Disk name: **az12001a-vm0-DataDisk0**
   - Resource group: *the name of the resource group you used earlier in this task*
   - HOST CACHING: **Read-only**
8. Repeat the previous step to attach the remaining 7 disks with the prefix **az12001a-vm0-DataDisk** (for the total of 8). Assign the LUN number matching the last character of the disk name. Set HOST CACHING of the disk with LUN **1** to **Read-only** and, for all the remaining ones, set HOST CACHING to **None**.
9. Save your changes.
10. In the Azure portal, navigate to the blade of the second Azure VM you provisioned in the previous task (**az12001a-vm1**).
11. From the **az12001a-vm1** blade, navigate to the **az12001a-vm1 | Disks** blade.
12. From the **az12001a-vm1 | Disks** blade, attach data disks with the following settings to az12001a-vm1:
    - LUN: **0**
    - Disk name: **az12001a-vm1-DataDisk0**
    - Resource group: *the name of the resource group you used earlier in this task*
    - HOST CACHING: **Read-only**
13. Repeat the previous step to attach the remaining 7 disks with the prefix **az12001a-vm1-DataDisk** (for the total of 8). Assign the LUN number matching the last character of the disk name. Set HOST CACHING of the disk with LUN **1** to **Read-only** and, for all the remaining ones, set HOST CACHING to **None**.
14. Save your changes.

    Result: After you completed this exercise, you have provisioned Azure compute resources necessary to support highly available SAP HANA deployments.

2.5 Exercise 2: Configure operating system of Azure VMs running Linux to support a highly available SAP HANA installation

Duration: 30 minutes

In this exercise, you will configure operating system and storage on Azure VMs running SUSE Linux Enterprise Server to accommodate clustered installations of SAP HANA.

2.5.1 Task 1: Connect to Azure Linux VMs

1. In the Azure Portal, start a Bash session in Cloud Shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `RESOURCE_GROUP_NAME` to the name of the resource group containing the resources you provisioned in the previous exercise:

   ```
   RESOURCE_GROUP_NAME='az12001a-RG'
   ```

3. In the Cloud Shell pane, run the following command to identify the public IP address of the first Azure VM you deployed in the previous exercise:

   ```
   PIP=$(az network public-ip show --resource-group $RESOURCE_GROUP_NAME --name az12001a-vm0-ip --que
   ```

4. In the Cloud Shell pane, run the following command to establish an SSH session to the IP address you identified in the previou step:

   ```
   ssh student@$PIP
   ```

5. When prompted whether you are sure to continue connecting, type `yes` and press the **Enter** key.
6. When prompted for the password, type `Pa55w.rd1234` and press the **Enter** key.
7. Open another Cloud Shell Bash session by clicking the **Open new session** icon in the Cloud Shell toolbar.
8. In the newly opened Cloud Shell Bash session, repeat all of the steps in this tasks to connect to the **az12001a-vm1** Azure VM via its IP address **az12001a-vm0-ip**.

2.5.2 Task 2: Configure storage of Azure VMs running Linux

1. In the Cloud Shell pane, in the SSH session to az12001a-vm0, run the following command to elevate privileges:

   ```
   sudo su -
   ```

2. In the Cloud Shell pane, in the SSH session to az12001a-vm0, run the following command to identify the mapping between the newly attached devices and their LUN numbers:

   ```
   lsscsi
   ```

3. In the Cloud Shell pane, in the SSH session to az12001a-vm0, create physical volumes for 6 (out of 8) data disks by running:

   ```
   pvcreate /dev/sdc
   pvcreate /dev/sdd
   pvcreate /dev/sde
   pvcreate /dev/sdf
   pvcreate /dev/sdg
   pvcreate /dev/sdh
   ```

4. In the Cloud Shell pane, in the SSH session to az12001a-vm0, create volume groups by running:

   ```
   vgcreate vg_hana_data /dev/sdc /dev/sdd
   vgcreate vg_hana_log /dev/sde /dev/sdf
   vgcreate vg_hana_backup /dev/sdg /dev/sdh
   ```

5. In the Cloud Shell pane, in the SSH session to az12001a-vm0, create logical volumes by running:

   ```
   lvcreate -l 100%FREE -n hana_data vg_hana_data
   lvcreate -l 100%FREE -n hana_log vg_hana_log
   lvcreate -l 100%FREE -n hana_backup vg_hana_backup
   ```

   Note: We are creating a single logical volume per each volume group

6. In the Cloud Shell pane, in the SSH session to az12001a-vm0, format the logical volumes by running:

   ```
   mkfs.xfs /dev/vg_hana_data/hana_data -m crc=1
   mkfs.xfs /dev/vg_hana_log/hana_log -m crc=1
   mkfs.xfs /dev/vg_hana_backup/hana_backup -m crc=1
   ```

   Note: Starting with SUSE Linux Enterprise Server 12, you have the option to use the new on-disk format (v5) of the XFS file system, which offers automatic checksums of XFS metadata, file type support, and an increased limit on the number of access control lists per file. The

new format applies automatically when using YaST to create the XFS file systems. To create an XFS file system in the older format for compatibility reasons, use the mkfs.xfs command without the `-m crc=1` option.

7. In the Cloud Shell pane, in the SSH session to az12001a-vm0, partition the **/dev/sdi** disk by running:

   ```
   fdisk /dev/sdi
   ```

8. When prompted, type, in sequence, `n`, `p`, `1` (followed by the **Enter** key each time) press the **Enter** key twice, and then type `w` to complete the write.

9. In the Cloud Shell pane, in the SSH session to az12001a-vm0, partition the **/dev/sdj** disk by running:

   ```
   fdisk /dev/sdj
   ```

10. When prompted, type, in sequence, `n`, `p`, `1` (followed by the **Enter** key each time) press the **Enter** key twice, and then type `w` to complete the write.

11. In the Cloud Shell pane, in the SSH session to az12001a-vm0, format the newly created partition by running (type `y` and press the **Enter** key when prompted for confirmation) :

    ```
    mkfs.xfs /dev/sdi -m crc=1 -f
    mkfs.xfs /dev/sdj -m crc=1 -f
    ```

12. In the Cloud Shell pane, in the SSH session to az12001a-vm0, create the directories that will serve as mount points by running:

    ```
    mkdir -p /hana/data
    mkdir -p /hana/log
    mkdir -p /hana/backup
    mkdir -p /hana/shared
    mkdir -p /usr/sap
    ```

13. In the Cloud Shell pane, in the SSH session to az12001a-vm0, display the ids of logical volumes by running:

    ```
    blkid
    ```

    Note: Identify the **UUID** values associated with the newly created volume groups and partitions, including **/dev/sdi** (to be used for **/hana/shared**) and **dev/sdj** (to be used for **/usr/sap**).

14. In the Cloud Shell pane, in the SSH session to az12001a-vm0, open **/etc/fstab** in the vi editor (you are free to use any other editor) by running:

    ```
    vi /etc/fstab
    ```

15. In the editor, add the following entries to **/etc/fstab** (where `\<UUID of /dev/vg\_hana\_data-hana\_data\>`, `\<UUID of /dev/vg\_hana\_log-hana\_log\>`, `\<UUID of /dev/vg\_hana\_backup-hana\_backup\>`, `\<UUID of /dev/vg_hana_shared-hana_shared (/dev/sdi)\>`, and `\<UUID of /dev/vg_usr_sap-usr_sap (/dev/sdj)\>`, represent the ids you identified in the previous step):

    ```
    /dev/disk/by-uuid/<UUID of /dev/vg_hana_data-hana_data> /hana/data xfs  defaults,nofail  0  2
    /dev/disk/by-uuid/<UUID of /dev/vg_hana_log-hana_log> /hana/log xfs  defaults,nofail  0  2
    /dev/disk/by-uuid/<UUID of /dev/vg_hana_backup-hana_backup> /hana/backup xfs  defaults,nofail  0  2
    /dev/disk/by-uuid/<UUID of /dev/vg_hana_shared-hana_shared (/dev/sdi)> /hana/shared xfs  defaults,n
    /dev/disk/by-uuid/<UUID of /dev/vg_usr_sap-usr_sap (/dev/sdj)> /usr/sap xfs  defaults,nofail  0  2
    ```

16. Save the changes and close the editor.

17. In the Cloud Shell pane, in the SSH session to az12001a-vm0, mount the new volumes by running:

    ```
    mount -a
    ```

18. In the Cloud Shell pane, in the SSH session to az12001a-vm0, verify that the mount was successful by running:

    ```
    df -h
    ```

19. Switch to the Cloud Shell Bash session to az12001a-vm1 and repeat all of the steps in this tasks to configure storage on **az12001a-vm1**.

2.5.3 Task 3: Enable cross-node password-less SSH access

1. In the Cloud Shell pane, in the SSH session to az12001a-vm0, generate passphrase-less SSH key by running:

   ```
   ssh-keygen -tdsa
   ```

2. When prompted, press **Enter** three times and then display the public key by running:

   ```
   cat /root/.ssh/id_dsa.pub
   ```

3. Copy the value of the key into Clipboard.

4. Switch to the Cloud Shell pane containing the SSH session to **az12001a-vm1** and create the directory **/root/.ssh/** by running:

   ```
   mkdir /root/.ssh
   ```

5. In the Cloud Shell pane, in the SSH session to az12001a-vm1, create a file **/root/.ssh/authorized_keys** in the vi editor (you are free to use any other editor) by running:

   ```
   vi /root/.ssh/authorized_keys
   ```

6. In the editor window, paste the key you generated on az12001a-vm0.

7. Save the changes and close the editor.

8. In the Cloud Shell pane, in the SSH session to az12001a-vm1, generate passphrase-less SSH key by running:

   ```
   ssh-keygen -tdsa
   ```

9. When prompted, press **Enter** three times and then display the public key by running:

   ```
   cat /root/.ssh/id_dsa.pub
   ```

10. Copy the value of the key into Clipboard.

11. Switch to the Cloud Shell pane containing the SSH session to az12001a-vm0 and create a file **/root/.ssh/authorized_keys** in the vi editor (you are free to use any other editor) by running:

    ```
    vi /root/.ssh/authorized_keys
    ```

12. In the editor window, paste the key you generated on az12001a-vm1.

13. Save the changes and close the editor.

14. In the Cloud Shell pane, in the SSH session to az12001a-vm0, generate passphrase-less SSH key by running:

    ```
    ssh-keygen -t rsa
    ```

15. When prompted, press **Enter** three times and then display the public key by running:

    ```
    cat /root/.ssh/id_rsa.pub
    ```

16. Copy the value of the key into Clipboard.

17. Switch to the Cloud Shell pane containing the SSH session to **az12001a-vm1** and open the file **/root/.ssh/authorized_keys** in the vi editor (you are free to use any other editor) by running:

    ```
    vi /root/.ssh/authorized_keys
    ```

18. In the editor window, starting from a new line, paste the key you generated on az12001a-vm0.

19. Save the changes and close the editor.

20. In the Cloud Shell pane, in the SSH session to az12001a-vm1, generate passphrase-less SSH key by running:

    ```
    ssh-keygen -t rsa
    ```

21. When prompted, press **Enter** three times and then display the public key by running:

    ```
    cat /root/.ssh/id_rsa.pub
    ```

22. Copy the value of the key into Clipboard.

23. Switch to the Cloud Shell pane containing the SSH session to az12001a-vm0 and open the file **/root/.ssh/authorized_keys** in the vi editor (you are free to use any other editor) by running:

    ```
    vi /root/.ssh/authorized_keys
    ```

24. In the editor window, starting from a new line, paste the key you generated on az12001a-vm1.
25. Save the changes and close the editor.
26. In the Cloud Shell pane, in the SSH session to az12001a-vm0, open the file **/etc/ssh/sshd_config** in the vi editor (you are free to use any other editor) by running:

    ```
    vi /etc/ssh/sshd_config
    ```

27. In the **/etc/ssh/sshd_config** file, locate the **PermitRootLogin** and **AuthorizedKeysFile** entries, and configure them as follows (remove the leading # character if needed:

    ```
    PermitRootLogin yes
    AuthorizedKeysFile  /root/.ssh/authorized_keys
    ```

28. Save the changes and close the editor.
29. In the Cloud Shell pane, in the SSH session to az12001a-vm0, restart sshd daemon by running:

    ```
    systemctl restart sshd
    ```

30. Repeat the previous four steps on az12001a-vm1.
31. To verify that the configuration was successful, in the Cloud Shell pane, in the SSH session to az12001a-vm0, establish an SSH session as **root** from az12001a-vm0 to az12001a-vm1 by running:

    ```
    ssh root@az12001a-vm1
    ```

32. When prompted whether you are sure to continue connecting, type `yes` and press the **Enter** key.
33. Ensure that you are not prompted for the password.
34. Close the SSH session from az12001a-vm0 to az12001a-vm1 by running:

    ```
    exit
    ```

35. Sign out from az12001a-vm0 by running the following twice:

    ```
    exit
    ```

36. To verify that the configuration was successful, in the Cloud Shell pane, in the SSH session to az12001a-vm1, establish an SSH session as **root** from az12001a-vm1 to az12001a-vm0 by running:

    ```
    ssh root@az12001a-vm0
    ```

37. When prompted whether you are sure to continue connecting, type `yes` and press the **Enter** key.
38. Ensure that you are not prompted for the password.
39. Close the SSH session from az12001a-vm1 to az12001a-vm0 by running:

    ```
    exit
    ```

40. Sign out from az12001a-vm1 by running the following twice:

    ```
    exit
    ```

    Result: After you completed this exercise, you have configured operating system of Azure VMs running Linux to support a highly available SAP HANA installation

2.6 Exercise 3: Provision Azure network resources necessary to support highly available SAP HANA deployments

Duration: 30 minutes

In this exercise, you will implement Azure Load Balancers to accommodate clustered installations of SAP HANA.

2.6.1 Task 1: Configure Azure VMs to facilitate load balancing setup.

Note: Since you will be setting up a pair of Azure Load Balancer of the Stardard SKU, you need to first remove the public IP addresses associated with network adapters of two Azure VMs that will be serving as the load-balanced backend pool.

1. In the Azure portal, navigate to the blade of the **az12001a-vm0** Azure VM.

2. From the **az12001a-vm0** blade, navigate to the **az12001a-vm0 | Networking** blade and, on the **az12001a-vm0 | Networking** blade, select the entry representing the public IP address **az12001a-vm0-ip** associated with its network adapter.
3. On the **az12001a-vm0-ip** blade, select **Dissociate** to disconnect the public IP address from the network interface and then select **Delete** to delete it.
4. In the Azure portal, navigate to the blade of the **az12001a-vm1** Azure VM.
5. From the **az12001a-vm1** blade, navigate to the **az12001a-vm1 | Networking** blade and, on the **az12001a-vm1 | Networking** blade, select the entry representing the public IP address **az12001a-vm1-ip** associated with its network adapter.
6. On the **az12001a-vm1-ip** blade, select **Dissociate** to disconnect the public IP address from the network interface and then select **Delete** to delete it.
7. In the Azure portal, navigate to the blade of the **az12001a-vm0** Azure VM.
8. From the **az12001a-vm0** blade, navigate to the **az12001a-vm0 | Networking** blade.
9. From the **az12001a-vm0 | Networking** blade, select the entry representing the network interface of the az12001a-vm0.
10. From the blade of the network interface of the az12001a-vm0, navigate to its IP configurations blade and, from there, display its **ipconfig1** blade.
11. On the **ipconfig1** blade, set the private IP address assignment to **Static** and save the change.
12. In the Azure portal, navigate to the blade of the **az12001a-vm1** Azure VM.
13. From the **az12001a-vm1** blade, navigate to the **az12001a-vm1 | Networking** blade.
14. From the **az12001a-vm1 | Networking** blade, navigate to the network interface of the az12001a-vm1.
15. From the blade of the network interface of the az12001a-vm1, navigate to its IP configurations blade and, from there, display its **ipconfig1** blade.
16. On the **ipconfig1** blade, set the private IP address assignment to **Static** and save the change.

2.6.2 Task 2: Create and configure Azure Load Balancers handling inbound traffic

1. In the Azure portal, use the **Search resources, services, and docs** text box at the top of the Azure portal page to search for and navigate to the **Load balancers** blade and, on the **Load balancers** blade, select **+ Add**.
2. From the **Basics** tab of the **Create load balancer** blade, specify the following settings and select **Review + create** (leave others with their default values):
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of the resource group you used earlier in this lab*
   - Name: **az12001a-lb0**
   - Region: *the same Azure region where you deployed Azure VMs in the first exercise of this lab*
   - Type: **Internal**
   - SKU: **Standard**
   - Virtual network: **az12001a-RG-vnet**
   - Subnet: **subnet-0**
   - IP address assignment: **Static**
   - IP address: **192.168.0.240**
   - Availability zone: **Zone redundant**
3. On the **Review + create** blade, select **Create**.

   Note: Wait until the load balancer is provisioned. This shoudl take less than a minute.

4. In the Azure portal, navigate to the blade displaying the properties of the newly provisioned **az12001a-lb0** load balancer.
5. On the **az12001a-lb0** blade, select **Backend pools**, select **+ Add**, and, on the **Add backend pool** specify the following settings (leave others with their default values):
   - Name: **az12001a-lb0-bepool**
   - IP version: **IPv4**
   - Virtual machine: **az12001a-vm0** IP Configuration: **ipconfig1 (192.168.0.4)**
   - Virtual machine: **az12001a-vm1** IP Configuration: **ipconfig1 (192.168.0.5)**
6. On the **az12001a-lb0** blade, select **Health probes** select **+ Add**, and, on the **Add health probe** blade, specify the following settings (leave others with their defaults):
   - Name: **az12001a-lb0-hprobe**
   - Protocol: **TCP**
   - Port: **62500**
   - Interval: **5** *seconds*
   - Unhealthy threshold: **2** *consecutive failures*
7. On the **az12001a-lb0** blade, select **Load balancing rules**, select **+ Add**, and, on the **Add load balancing rule** blade, specify the following settings (leave others with their defaults):
   - Name: **az12001a-lb0-lbruleAll**
   - IP Version: **IPv4**
   - Frontend IP address: **192.168.0.240 (LoadBalancerFrontEnd)**
   - HA Ports: **Enabled**
   - Backend pool: **az12001a-lb0-bepool (2 virtual machines)**
   - Health probe:**az12001a-lb0-hprobe (TCP:62504)**
   - Session persistence: **None**
   - Idle timeout (minutes): **4**
   - TCP reset: **Disabled**
   - Floating IP (direct server return): **Enabled**

2.6.3 Task 3: Create and configure Azure Load Balancers handling outbound traffic

1. In the Azure Portal, start a Bash session in Cloud Shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `RESOURCE_GROUP_NAME` to the name of the resource group containing the resources you provisioned in the first exercise of this lab:

   ```
   RESOURCE_GROUP_NAME='az12001a-RG'
   ```

3. In the Cloud Shell pane, run the following command to create the public IP address to be used by the second load balancer:

   ```
   LOCATION=$(az group list --query "[?name == '$RESOURCE_GROUP_NAME'].location" --output tsv)

   PIP_NAME='az12001a-lb1-pip'

   az network public-ip create --resource-group $RESOURCE_GROUP_NAME --name $PIP_NAME --sku Standard
   ```

4. In the Cloud Shell pane, run the following command to create the second load balancer:

   ```
   LB_NAME='az12001a-lb1'

   LB_BE_POOL_NAME='az12001a-lb1-bepool'

   LB_FE_IP_NAME='az12001a-lb1-fe'
   ```

```
az network lb create --resource-group $RESOURCE_GROUP_NAME --name $LB_NAME --sku Standard --backend
```

5. In the Cloud Shell pane, run the following command to create the outbound rule of the second load balancer:

```
LB_RULE_OUTBOUND='az12001a-lb1-ruleoutbound'

az network lb outbound-rule create --resource-group $RESOURCE_GROUP_NAME --lb-name $LB_NAME --name
```

6. Close the Cloud Shell pane.
7. In the Azure portal, navigate to the blade displaying the properties of the newly created Azure Load Balancer **az12001a-lb1**.
8. On the **az12001a-lb1** blade, click **Backend pools**.
9. On the **az12001a-lb1 | Backend pools** blade, click **az12001a-lb1-bepool**.
10. On the **az12001a-lb1-bepool** blade, specify the following settings and click **Save**:
    - Virtual network: **az12001a-rg-vnet (2 VM)**
    - Virtual machine: **az12001a-vm0** IP Configuration: **ipconfig1 (192.168.0.4)**
    - Virtual machine: **az12001a-vm1** IP Configuration: **ipconfig1 (192.168.0.5)**

2.6.4 Task 4: Deploy a jump host

Note: Since two clustered Azure VMs are no longer directly accessible from Internet, you will deploy an Azure VM running Windows Server 2019 Datacenter that will serve as a jump host.

1. From the lab computer, in the Azure portal, use the **Search resources, services, and docs** text box at the top of the Azure portal page to search for and navigate to the **Virtual machines** blade, then, on the **Virtual machines** blade, select **+ Add** and, in the drop-down menu, select **Virtual machine**.
2. On the **Basics** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Disks >** (leave all other settings with their default value):
    - Subscription: *the name of your Azure subscription*
    - Resource group: *the name of the resource group you used earlier in this lab*
    - Virtual machine name: **az12001a-vm2**
    - Region: *the same Azure region where you deployed Azure VMs in the first exercise of this lab*
    - Availability options: **No infrastructure redundancy required**
    - Image: **Windows Server 2019 Datacenter - Gen1**
    - Size: **Standard DS1 v2*** or similar*
    - Azure Spot Instance: **No**
    - Username: **Student**
    - Password: **Pa55w.rd1234**
    - Public inbound ports: **Allow selected ports**
    - Selected inbound ports: **RDP (3389)**
    - Would you like to use an existing Windows Server license?: **No**
3. On the **Disks** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Networking >** (leave all other settings with their default value):
    - OS disk type: **Standard HDD**
    - Encryption type: **(Default) Encryption at rest with a platform-managed key**
4. On the **Networking** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Management >** (leave all other settings with their default value):
    - Virtual network: **az12001a-RG-vnet**

- Subnet name: **subnet-0 (192.168.0.0/24)**
- Public IP address: *a new IP address named* **az12001a-vm2-ip**
- NIC network security group: **Basic**
- Public inbound ports: **Allow selected ports**
- Select inbound ports: **RDP (3389)**
- Accelerated networking: **Off**
- Place this virtual machine behind an existing load balancing solutions: **No**

5. On the **Management** tab of the **Create a virtual machine** blade, specify the following settings and select **Next: Advanced >** (leave all other settings with their default value):
   - Enable basic plan for free: **No**

     Note: This setting is not available if you have already selected the Azure Security Center plan.
   - Boot diagnostics: **Enable with managed storage account (recommended)**
   - OS guest diagnostics: **Off**
   - System assigned managed identity: **Off**
   - Enable auto-shutdown: **Off**
   - Enable backup: **Off**
   - Guest OS updates: **Manual patching: Install patches yourself or through a different patching solution**
6. On the **Advanced** tab of the **Create a virtual machine** blade, select **Review + create** (leave all other settings with their default value):
7. On the **Review + create** tab of the **Create Proximity Placement Groups** blade, select **Create**.

   Note: Wait for the provisioning to complete. This should take less about 3 minutes.
8. Connect to the newly provisioned Azure VM via RDP.
9. Within the RDP session to az12001a-vm2, download PuTTY from **https://www.chiark.greenend.org.uk/~sgtatha**
10. Ensure that you can establish SSH session to both az12001a-vm0 and az12001a-vm1 via their private IP addresses (192.168.0.4 and 192.168.0.5, respectively).

    Result: After you completed this exercise, you have provisioned Azure network resources necessary to support highly available SAP HANA deployments

2.7 Exercise 4: Remove lab resources

Duration: 10 minutes

In this exercise, you will remove resources provisioned in this lab.

2.7.0.1 Task 1: Open Cloud Shell

1. At the top of the portal, click the **Cloud Shell** icon to open Cloud Shell pane and choose Bash as the shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `RESOURCE_GROUP_PREFIX` to the prefix of the name of the resource group containing the resources you provisioned in this lab:

   ```
   RESOURCE_GROUP_PREFIX='az12001a-'
   ```
3. In the Cloud Shell pane, run the following command to list all resource groups you created in this lab:

   ```
   az group list --query "[?starts_with(name,'$RESOURCE_GROUP_PREFIX')]".name --output tsv
   ```
4. Verify that the output contains only the resource group you created in this lab. This resource group with all of their resources will be deleted in the next task.

2.7.0.2 Task 2: Delete resource groups

1. In the Cloud Shell pane, run the following command to delete the resource group and their resources.

   ```
   az group list --query "[?starts_with(name,'$RESOURCE_GROUP_PREFIX')]".name --output tsv | xargs -L
   ```

2. Close the Cloud Shell pane.

   Result: After you completed this exercise, you have removed the resources used in this lab.

3 AZ 120 Module 1: Foundations of SAP on Azure

4 Lab 1b: Implementing Windows clustering on Azure VMs

Estimated Time: 120 minutes

All tasks in this lab are performed from the Azure portal (including the PowerShell Cloud Shell session)

> **Note**: When not using Cloud Shell, the lab virtual machine must have Az PowerShell module installed **https://docs.microsoft.com/en-us/powershell/azure/install-az-ps-msi?view=azps-2.8.0**.

Lab files: none

4.1 Scenario

In preparation for deployment of SAP NetWeaver on Azure, with SQL Server as the database management system, Adatum Corporation wants to explore the process of implementing clustering on Azure VMs running Windows Server 2019.

4.2 Objectives

After completing this lab, you will be able to:

- Provision Azure compute resources necessary to support highly available SAP NetWeaver deployments.
- Configure operating system of Azure VMs running Windows Server 2019 to support a highly available SAP NetWeaver deployment.
- Provision Azure network resources necessary to support highly available SAP NetWeaver deployments.

4.3 Requirements

- A Microsoft Azure subscription with the sufficient number of available DSv2 and Dsv3 vCPUs (one Standard_DS1_v2 VM with 1 vCPU and four Standard_D4s_v3 VMs with 4 vCPUs each) in the Azure region you intend to use for this lab
- A lab computer with an Azure Cloud Shell-compatible web browser and access to Azure

  Note: Consider using **East US** or **East US2** regions for deployment of your resources.

4.4 Exercise 1: Provision Azure compute resources necessary to support highly available SAP NetWeaver deployments

Duration: 50 minutes

In this exercise, you will deploy Azure infrastructure compute components necessary to configure Failover Clustering on Azure VMs running Windows Server 2019. This will involve deploying a pair of Active Directory domain controllers, followed by a pair of Azure VMs running Windows Server 2019 in the same availability set within the same virtual network. To automate the deployment of domain controllers, you will use an Azure Resource Manager QuickStart template available from https://github.com/polichtm/azure-quickstart-templates/tree/master/active-directory-new-domain-ha-2-dc

4.4.1 Task 1: Deploy a pair of Azure VMs running highly available Active Directory domain controllers by using an Azure Resource Manager template

1. From the lab computer, start a Web browser, and navigate to the Azure portal at https://portal.azure.com
2. If prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you will be using for this lab.
3. Open a new web browser tab, navigate to Azure Quickstart Templates page at https://github.com/polichtm/azure-quickstart-templates, locate the template named **Create 2 new Windows VMs, create a new AD Forest, Domain, and 2 DCs in an availability set**, and initiate its deployment by clicking **Deploy to Azure** button.
4. On the **Custom deployment** blade, specify the following settings and click **Review + create**, followed by **Create** to initiate the deployment:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of a new resource group* **az12001b-ad-RG**
   - Location: *an Azure region where you can deploy Azure VMs*

     Note: Consider using **East US** or **East US2** regions for deployment of your resources.
   - Admin Username: **Student**
   - Password: **Pa55w.rd1234**
   - Domain Name: **adatum.com**
   - DnsPrefix: *any unique valid DNS prefix*
   - Pdc RDP Port: **3389**
   - Bdc RDP Port: **13389**
   - _artifacts Location: **https://raw.githubusercontent.com/polichtm/azure-quickstart-templates/master/active-directory-new-domain-ha-2-dc/**
   - _artifacts Location Sas Token: *leave blank*

     Note: The deployment should take about 35 minutes. Wait for the deployment to complete before you proceed to the next task.

     Note: If the deployment fails with the **Conflict** error message during deployment of the CustomScriptExtension component, use the following steps to remediate this issue:
   - in the Azure portal, on the **Deployment** blade, review the deployment details and identify the VM(s) where the installation of the CustomScriptExtension failed
   - in the Azure portal, navigate to the blade of the VM(s) you identified in the previous step, select **Extensions**, and from the **Extensions** blade, remove the CustomScript extension
   - in the Azure portal, navigate to the **az12003b-ad-RG** resource group blade, select **Deployments**, select the link to the failed deployment, and select **Redeploy**, select the target resource group (**az12003b-ad-RG**) and provide the password for the root account (**Pa55w.rd1234**).

4.4.2 Task 2: Deploy a pair of Azure VMs running Windows Server 2016 in the same availability set.

1. From the lab computer, in the Azure portal, click **+ Create a resource**.
2. From the **New** blade, initiate provisioning of a **Windows Server 2019 Datacenter** Azure VM with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of a new resource group* **az12001b-cl-RG**
   - Virtual machine name: **az12001b-cl-vm0**
   - Region: *the same Azure region where you deployed the Azure VMs in the previous task*
   - Availability options: **Availability set**

- Availability set: *a new availability set named* **az12001b-cl-avset** *with 2 fault domains and 5 update domains*
- Image: **Windows Server 2019 Datacenter**
- Size: **Standard D4s v3**
- Username: **Student**
- Password: **Pa55w.rd1234**
- Public inbound ports: **Allow selected ports**
- Select inbound ports: **RDP (3389)**
- You already have a Windows license?: **No**
- OS disk type: **Premium SSD**
- Virtual network: **adVNET**
- Subnet name: *a new subnet named* **clSubnet**
- Subnet address range: **10.0.1.0/24**
- Public IP address: *a new IP address named* **az12001b-cl-vm0-ip**
- NIC network security group: **Basic**
- Public inbound ports: **Allow selected ports**
- Select inbound ports: **RDP (3389)**
- Accelerated networking: **On**
- Place this virtual machine behind an existing load balancing solutions: **No**
- Enable basic plan for free: **No**
- Boot diagnostics: **Off**
- OS guest diagnostics: **Off**
- System assigned managed identity: **Off**
- Login with AAD credentials (Preview): **Off**
- Enable auto-shutdown: **Off**
- Enable backup: **Off**
- Extensions: *None*
- Tags: *None*

3. Do not wait for the provisioning to complete but continue to the next step.
4. Provision another **Windows Server 2019 Datacenter** Azure VM with the following settings:
    - Subscription: *the name of your Azure subscription*
    - Resource group: *the name of the resource group you used when deploying the first* ***Windows Server 2019 Datacenter*** *Azure VM in this task*
    - Virtual machine name: **az12001b-cl-vm1**
    - Region: *the same Azure region where you deployed the first* ***Windows Server 2019 Datacenter*** *Azure VM in this task*
    - Availability options: **Availability set**
    - Availability set: **az12001b-cl-avset**
    - Image: **Windows Server 2019 Datacenter**
    - Size: **Standard D4s v3**
    - Username: **Student**
    - Password: **Pa55w.rd1234**

- Public inbound ports: **Allow selected ports**
- Select inbound ports: **RDP (3389)**
- You already have a Windows license?: **No**
- OS disk type: **Premium SSD**
- Virtual network: **adVNET**
- Subnet name: **clSubnet**
- Public IP address: *a new IP address named* **az12001b-cl-vm1-ip**
- NIC network security group: **Basic**
- Public inbound ports: **Allow selected ports**
- Select inbound ports: **RDP (3389)**
- Accelerated networking: **On**
- Place this virtual machine behind an existing load balancing solutions: **No**
- Enable basic plan for free: **No**
- Boot diagnostics: **Off**
- OS guest diagnostics: **Off**
- System assigned managed identity: **Off**
- Login with AAD credentials (Preview): **Off**
- Enable auto-shutdown: **Off**
- Enable backup: **Off**
- Extensions: *None*
- Tags: *None*

5. Wait for the provisioning to complete. This should take a few minutes.

4.4.3 Task 3: Create and configure Azure VMs disks

1. In the Azure Portal, start a PowerShell session in Cloud Shell.

   Note: If this is the first time you are launching Cloud Shell in the current Azure subscription, you will be asked to create an Azure file share to persist Cloud Shell files. If so, accept the defaults, which will result in creation of a storage account in an automatically generated resource group.

2. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the resources you provisioned in the previous task:

   ```
   $resourceGroupName = 'az12001b-cl-RG'
   ```

3. In the Cloud Shell pane, run the following command, to create the first set of 4 managed disks that you will attach to the first Azure VM you deployed in the previous task:

   ```
   $location = (Get-AzResourceGroup -Name $resourceGroupName).Location

   $diskConfig = New-AzDiskConfig -Location $location -DiskSizeGB 128 -AccountType Premium_LRS -OsTyp

   for ($i=0;$i -lt 4;$i++) {New-AzDisk -ResourceGroupName $resourceGroupName -DiskName az12001b-cl-v
   ```

4. In the Cloud Shell pane, run the following command, to create the second set of 4 managed disks that you will attach to the second Azure VM you deployed in the previous task:

   ```
   for ($i=0;$i -lt 4;$i++) {New-AzDisk -ResourceGroupName $resourceGroupName -DiskName az12001b-cl-v
   ```

5. In the Azure portal, navigate to the blade of the first Azure VM you provisioned in the previous task (**az12001b-cl-vm0**).

6. From the **az12001b-cl-vm0** blade, navigate to the **az12001b-cl-vm0 - Disks** blade.

7. From the **az12001b-cl-vm0 - Disks** blade, attach data disks with the following settings to az12001b-cl-vm0:
   - LUN: **0**
   - Disk name: **az12001b-cl-vm0-DataDisk0**
   - Resource group: *the name of the resource group you used when deploying the pair of **Windows Server 2019 Datacenter** Azure VMs in the previous task*
   - HOST CACHING: **Read-only**
8. Repeat the previous step to attach the remaining 3 disks with the prefix **az12001b-cl-vm0-DataDisk** (for the total of 4). Assign the LUN number matching the last character of the disk name. For the last disk (LUN **4**), set HOST CACHING to **None**.
9. Save your changes.
10. In the Azure portal, navigate to the blade of the second Azure VM you provisioned in the previous task (**az12001b-cl-vm1**).
11. From the **az12001b-cl-vm1** blade, navigate to the **az12001b-cl-vm1 - Disks** blade.
12. From the **az12001b-cl-vm1 - Disks** blade, attach data disks with the following settings to az12001b-cl-vm1:
    - LUN: **0**
    - Disk name: **az12001b-cl-vm1-DataDisk0**
    - Resource group: *the name of the resource group you used when deploying the pair of **Windows Server 2019 Datacenter** Azure VMs in the previous task*
    - HOST CACHING: **Read-only**
13. Repeat the previous step to attach the remaining 3 disks with the prefix **az12001b-cl-vm1-DataDisk** (for the total of 4). Assign the LUN number matching the last character of the disk name. For the last disk (LUN **4**), set HOST CACHING to **None**.
14. Save your changes.

    Result: After you completed this exercise, you have provisioned Azure compute resources necessary to support highly available SAP NetWeaver deployments.

4.5 Exercise 2: Configure operating system of Azure VMs running Windows Server 2019 to support a highly available SAP NetWeaver installation

Duration: 40 minutes

4.5.1 Task 1: Join Windows Server 2019 Azure VMs to the Active Directory domain.

Note: Before you start this task, ensure that the template deployment you initiated in the last task of the previous exercise has successfully completed.

1. In the Azure Portal, navigate to the blade of the virtual network **adVNET**, which was provisioned automatically in the first exercise of this lab.
2. Display the **adVNET - DNS servers** blade and note that the virtual network is configured with the private IP addresses assigned to the domain controllers deployed in the first exercise of this lab as its DNS servers.
3. In the Azure Portal, start a PowerShell session in Cloud Shell.
4. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the previous exercise:

   ```
   $resourceGroupName = 'az12001b-cl-RG'
   ```

5. In the Cloud Shell pane, run the following command, to join the Windows Server 2019 Azure VMs you deployed in the second task of the previous exercise to the **adatum.com** Active Directory domain:

```
$location = (Get-AzureRmResourceGroup -Name $resourceGroupName).Location

$settingString = '{"Name": "adatum.com", "User": "adatum.com\\Student", "Restart": "true", "Option

$protectedSettingString = '{"Password": "Pa55w.rd1234"}'

$vmNames = @('az12001b-cl-vm0','az12001b-cl-vm1')

foreach ($vmName in $vmNames) { Set-AzVMExtension -ResourceGroupName $resourceGroupName -Extension
```

6. Wait for the script to complete before proceeding to the next task.

4.5.2 Task 2: Configure storage on Azure VMs running Windows Server 2019 to support a highly available SAP NetWeaver installation.

1. In the Azure Portal, navigate to the blade of the virtual virtual machine **az12001b-cl-vm0**, which you provisioned in the first exercise of this lab.
2. From the **az12001b-cl-vm0** blade, connect to the virtual machine guest operating system by using Remote Desktop. When prompted to authenticate, provide the following credentials:
   - User name: **ADATUM\Student**
   - Password: **Pa55w.rd1234**
3. Within the RDP session to az12001b-cl-vm0, in Server Manager, navigate to the **Local Server** view and turn off temporarily **IE Enhanced Security Configuration**.
4. Within the RDP session to az12001b-cl-vm0, in Server Manager, navigate to the **File and Storage Services** -> **Servers** node.
5. Navigate to the **Storage Pools** view and verify that you see all the disks you attached to the Azure VM in the previous exercise.
6. Use the **New Storage Pools Wizard** to create a new storage pool with the following settings:
   - Name: **Data Storage Pool**
   - Physical Disks: *select the 3 disks with disk numbers corresponding to the first three LUN numbers (0-2) and set their allocation to* **Automatic**

     Note: Use the entry in the **Chassis** column to identify the **LUN** number.
7. Use the **New Virtual Disk Wizard** to create a new virtual disk with the following settings:
   - Virtual Disk Name: **Data Virtual Disk**
   - Storage Layout: **Simple**
   - Provisioning: **Fixed**
   - Size: **Maximum size**
8. Use the **New Volume Wizard** to create a new volume with the following settings:
   - Server and Disk: *accept the default values*
   - Size: *accept the default values*
   - Drive letter: **M**
   - File system: **ReFS**
   - Allocation unit size: **Default**
   - Volume label: **Data**
9. Back in the **Storage Pools** view, use the **New Storage Pools Wizard** to create a new storage pool with the following settings:
   - Name: **Log Storage Pool**
   - Physical Disks: *select the last of 4 disks and set its allocation to* **Automatic**

10. Use the **New Virtual Disk Wizard** to create a new virtual disk with the following settings:
    - Virtual Disk Name: **Log Virtual Disk**
    - Storage Layout: **Simple**
    - Provisioning: **Fixed**
    - Size: **Maximum size**
11. Use the **New Volume Wizard** to create a new volume with the following settings:
    - Server and Disk: *accept the default values*
    - Size: *accept the default values*
    - Drive letter: **L**
    - File system: **ReFS**
    - Allocation unit size: **Default**
    - Volume label: **Log**
12. Repeat the previous step in this task to configure storage on az12001b-cl-vm1.

4.5.3 Task 3: Prepare for configuration of Failover Clustering on Azure VMs running Windows Server 2019 to support a highly available SAP NetWeaver installation.

1. Within the RDP session to az12001b-cl-vm0, start a Windows PowerShell ISE session and install Failover Clustering and Remote Administrative tools features on both az12001b-cl-vm0 and az12001b-cl-vm1 by running the following:

```
$nodes = @('az12001b-cl-vm1', 'az12001b-cl-vm0')

Invoke-Command $nodes {Install-WindowsFeature Failover-Clustering -IncludeAllSubFeature -IncludeMa

Invoke-Command $nodes {Install-WindowsFeature RSAT -IncludeAllSubFeature -Restart}
```

   Note: This will result in restart of the guest operating system of both Azure VMs.

2. On the lab computer, in the Azure Portal, click **+ Create a resource**.
3. From the **New** blade, initiate creation of a new **Storage account** with the following settings:
    - Subscription: *the name of your Azure subscription*
    - Resource group: *the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the previous exercise*
    - Storage account name: *any unique name consisting of between 3 and 24 letters and digits*
    - Location: *the same Azure region where you deployed the Azure VMs in the previous exercise*
    - Performance: **Standard**
    - Account kind: **Storage (general purpose v1)**
    - Replication: **Locally-redundant storage (LRS)**
    - Connectivity method: **Public endpoint (all networks)**
    - Secure transfer required: **Enabled**
    - Large file shares: **Disabled**
    - Blob soft delete: **Disabled**
    - Hierarchical namespace: **Disabled**

4.5.4 Task 4: Configure Failover Clustering on Azure VMs running Windows Server 2019 to support a highly available SAP NetWeaver installation.

1. In the Azure Portal, navigate to the blade of the virtual virtual machine **az12001b-cl-vm0**, which you provisioned in the first exercise of this lab.
2. From the **az12001b-cl-vm0** blade, connect to the virtual machine guest operating system by using Remote Desktop. When prompted to authenticate, provide the following credentials:
   - User name: **ADATUM\Student**
   - Password: **Pa55w.rd1234**
3. Within the RDP session to az12001b-cl-vm0, from the **Tools** menu in Server Manager, start **Active Directory Administrative Center**.
4. In Active Directory Administrative Center, create a new organizational unit named **Clusters** in the root of the adatum.com domain.
5. In Active Directory Administrative Center, move the computer accounts of **az12001b-cl-vm0** and **az12001b-cl-vm1** from the **Computers** container to the **Clusters** organizational unit.
6. Within the RDP session to az12001b-cl-vm0, start a Windows PowerShell ISE session and create a new cluster by running the following:

   ```
   $nodes = @('az12001b-cl-vm0','az12001b-cl-vm1')

   New-Cluster -Name az12001b-cl-cl0 -Node $nodes -NoStorage -StaticAddress 10.0.1.6
   ```

7. Within the RDP session to az12001b-cl-vm0, switch to the **Active Directory Administrative Center** console.
8. In Active Directory Administrative Center, navigate to the **Clusters** organizational unit and display its **Properties** window.
9. In the **Clusters** organizational unit **Properties** window, navigate to the **Extensions** section, display the **Security** tab.
10. On the **Security** tab, click the **Advanced** button to open the **Advanced Security Settings for Clusters** window.
11. On the **Permissions** tab of the **Advanced Security Settings for Computers** window, click **Add**.
12. In the **Permission Entry for Clusters** window, click **Select Principal**
13. In the **Select User, Service Account or Group** dialog box, click **Object Types**, enable the checkbox next to the **Computers** entry, and click **OK**.
14. Back in the **Select User, Computer, Service Account or Group** dialog box, in the **Enter the object name to select**, type **az12001b-cl-cl0** and click **OK**.
15. In the **Permission Entry for Clusters** window, ensure that **Allow** appears in the **Type** drop-down list. Next, in the **Applies to** drop-down list, select **This object and all descendant objects**. In the **Permissions** list, select the **Create Computer objects** and **Delete Computer objects** checkboxes, and click **OK** twice.
16. Within the Windows PowerShell ISE session, install the Az PowerShell module by running the following:

    ```
    Install-PackageProvider -Name NuGet -Force

    Install-Module -Name Az -Force
    ```

17. Within the Windows PowerShell ISE session, authenticate by using your Azure AD credentials by running the following:

    ```
    Add-AzAccount
    ```

    Note: When prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you are using for this lab.

18. Within the Windows PowerShell ISE session, set the Cloud Witness quorum of the new cluster by running the following:

```
$resourceGroupName = 'az12001b-cl-RG'

$cwStorageAccountName = (Get-AzStorageAccount -ResourceGroupName $resourceGroupName)[0].StorageAcc

$cwStorageAccountKey = (Get-AzStorageAccountKey -ResourceGroupName $resourceGroupName -Name $cwSto

Set-ClusterQuorum -CloudWitness -AccountName $cwStorageAccountName -AccessKey $cwStorageAccountKey
```

19. To verify the resulting configuration, within the RDP session to az12001b-cl-vm0, from the **Tools** menu in Server Manager, start **Failover Cluster Manager**.
20. In the **Failover Cluster Manager** console, review the **az12001b-cl-cl0** cluster configuration, including its nodes, as well as is witness and network settings. Note that the cluster does not have any shared storage.
21. Terminate the RDP session to az12001b-cl-vm0.

    Result: After you completed this exercise, you have configured operating system of Azure VMs running Windows Server 2019 to support a highly available SAP NetWeaver installation

4.6 Exercise 3: Provision Azure network resources necessary to support highly available SAP NetWeaver deployments

Duration: 30 minutes

In this exercise, you will implement Azure Load Balancers to accommodate clustered installations of SAP NetWeaver.

4.6.1 Task 1: Configure Azure VMs to facilitate load balancing setup.

Note: Since you will be setting up a pair of Azure Load Balancer of the Stardard SKU, you need to first remove the public IP addresses associated with network adapters of two Azure VMs that will be serving as the load-balanced backend pool.

1. On the lab computer, in the Azure portal, navigate to the blade of the Azure VM **az12001b-cl-vm0**.
2. From the **az12001b-cl-vm0** blade, navigate to the blade of the public IP address **az12001b-cl-vm0-ip** associated with its network adapter.
3. From the **az12001b-cl-vm0-ip** blade, first disassociate the public IP address from the network interface and then delete it.
4. In the Azure portal, navigate to the blade of the Azure VM **az12001b-cl-vm1**.
5. From the **az12001b-cl-vm1** blade, navigate to the blade of the public IP address **az12001b-cl-vm1-ip** associated with its network adapter.
6. From the **az12001b-cl-vm1-ip** blade, first disassociate the public IP address from the network interface and then delete it.
7. In the Azure portal, navigate to the blade of the **az12001a-vm0** Azure VM.
8. From the **az12001a-vm0** blade, navigate to its **Networking** blade.
9. From the **az12001a-vm0 - Networking** blade, navigate to the network interface of the az12001a-vm0.
10. From the blade of the network interface of the az12001a-vm0, navigate to its IP configurations blade and, from there, display its **ipconfig1** blade.
11. On the **ipconfig1** blade, set the private IP address assignment to **Static** and save the change.
12. In the Azure portal, navigate to the blade of the **az12001a-vm1** Azure VM.
13. From the **az12001a-vm1** blade, navigate to its **Networking** blade.
14. From the **az12001a-vm1 - Networking** blade, navigate to the network interface of the az12001a-vm1.
15. From the blade of the network interface of the az12001a-vm1, navigate to its IP configurations blade and, from there, display its **ipconfig1** blade.
16. On the **ipconfig1** blade, set the private IP address assignment to **Static** and save the change.

4.6.2 Task 2: Create and configure Azure Load Balancers handling inbound traffic

1. In the Azure portal, click **+ Create a resource**.
2. From the **New** blade, initiate creation of a new Azure Load Balancer with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the first exercise of this lab*
   - Name: **az12001b-cl-lb0**
   - Region: *the same Azure region where you deployed Azure VMs in the first exercise of this lab*
   - Type: **Internal**
   - SKU: **Standard**
   - Virtual network: **adVNET**
   - Subnet: **clSubnet**
   - IP address assignment: **Static**
   - IP address: **10.0.1.240**
   - Availability zone: **No Zone**
3. Wait until the load balancer is provisioned and then navigate to its blade in the Azure portal.
4. From the **az12001b-cl-lb0** blade, add a backend pool with the following settings:
   - Name: **az12001b-cl-lb0-bepool**
   - Virtual network: **adVNET**
   - VIRTUAL MACHINE: **az12001b-cl-vm0** IP ADDRESS: **ipconfig1**
   - VIRTUAL MACHINE: **az12001b-cl-vm1** IP ADDRESS: **ipconfig1**
5. From the **az12001b-cl-lb0** blade, add a health probe with the following settings:
   - Name: **az12001b-cl-lb0-hprobe**
   - Protocol: **TCP**
   - Port: **59999**
   - Interval: **5** *seconds*
   - Unhealthy threshold: **2** *consecutive failures*
6. From the **az12001b-cl-lb0** blade, add a network load balancing rule with the following settings:
   - Name: **az12001b-cl-lb0-lbruletcp1433**
   - IP version: **IPv4**
   - Frontend IP address: **192.168.0.240 (LoadBalancerFrontEnd)**
   - HA Ports: **Disabled**
   - Protocol: **TCP**
   - Port: **1433**
   - Backend port: **1433**
   - Backend pool: **az12001b-cl-lb0-bepool (2 virtual machines)**
   - Health probe:**az12001b-cl-lb0-hprobe (TCP:59999)**
   - Session persistence: **None**
   - Idle timeout (minutes): **4**
   - Floating IP (direct server return): **Enabled**

4.6.3 Task 3: Create and configure Azure Load Balancers handling outbound traffic

1. From the Azure Portal, start a PowerShell session in Cloud Shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the first exercise of this lab:

   ```
   $resourceGroupName = 'az12001b-cl-RG'
   ```

3. In the Cloud Shell pane, run the following command to create the public IP address to be used by the second load balancer:

   ```
   $location = (Get-AzResourceGroup -Name $resourceGroupName).Location

   $pipName = 'az12001b-cl-lb0-pip'

   az network public-ip create --resource-group $resourceGroupName --name $pipName --sku Standard --l
   ```

4. In the Cloud Shell pane, run the following command to create the second load balancer:

   ```
   $lbName = 'az12001b-cl-lb1'

   $lbFeName = 'az12001b-cl-lb1-fe'

   $lbBePoolName = 'az12001b-cl-lb1-bepool'

   $pip = Get-AzPublicIpAddress -ResourceGroupName $resourceGroupName -Name $pipName

   $feIpconfiguration = New-AzLoadBalancerFrontendIpConfig -Name $lbFeName -PublicIpAddress $pip

   $bePoolConfiguration = New-AzLoadBalancerBackendAddressPoolConfig -Name $lbBePoolName

   New-AzLoadBalancer -ResourceGroupName $resourceGroupName -Location $location -Name $lbName -Sku St
   ```

5. Close the Cloud Shell pane.
6. In the Azure portal, navigate to the blade displaying the properties of the Azure Load Balancer **az12001b-cl-lb1**.
7. On the **az12001b-cl-lb1** blade, click **Backend pools**.
8. On the **az12001b-cl-lb1 - Backend pools** blade, click **az12001b-cl-lb1-bepool**.
9. On the **az12001b-cl-lb1-bepool** blade, specify the following settings and click **Save**:
   - Virtual network: **adVNET (4 VM)**
   - VIRTUAL MACHINE: **az12001b-cl-vm0** IP ADDRESS: **ipconfig1**
   - VIRTUAL MACHINE: **az12001b-cl-vm1** IP ADDRESS: **ipconfig1**
10. On the **az12001b-cl-lb1** blade, click **Health probes**.
11. From the **az12001b-cl-lb1 - Health probes** blade, add a health probe with the following settings:
    - Name: **az12001b-cl-lb1-hprobe**
    - Protocol: **TCP**
    - Port: **80**
    - Interval: **5** *seconds*
    - Unhealthy threshold: **2** *consecutive failures*
12. On the **az12001b-cl-lb1** blade, click **Load balancing rules**.
13. From the **az12001b-cl-lb1 - Load balancing rules** blade, add a network load balancing rule with the following settings:
    - Name: **az12001b-cl-lb1-lbharule**
    - IP version: **IPv4**

- Frontend IP address: *accept the default value*
- Protocol: **TCP**
- Port: **80**
- Backend port: **80**
- Backend pool: **az12001b-cl-lb1-bepool (2 virtual machines)**
- Health probe:**az12001b-cl-lb1-hprobe (TCP:80)**
- Session persistence: **None**
- Idle timeout (minutes): **4**
- Floating IP (direct server return): **Disabled**

4.6.4 Task 4: Deploy a jump host

Note: Since two clustered Azure VMs are no longer directly accessible from Internet, you will deploy an Azure VM running Windows Server 2019 Datacenter that will serve as a jump host.

1. From the lab computer, in the Azure portal, click **+ Create a resource**.
2. From the **New** blade, initiate creation of a new Azure VM based on the **Windows Server 2019 Datacenter** image.
3. Provision a Azure VM with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the first exercise of this lab*
   - Virtual machine name: **az12001b-vm2**
   - Region: *the same Azure region where you deployed Azure VMs in the first exercise of this lab*
   - Availability options: **No infrastructure redundancy required**
   - Image: **Windows Server 2019 Datacenter**
   - Size: **Standard DS1 v2*** or similar*
   - Username: **student**
   - Password: **Pa55w.rd1234**
   - Public inbound ports: **Allow selected ports**
   - Select inbound ports: **RDP (3389)**
   - You already have a Windows license?: **No**
   - OS disk type: **Standard HDD**
   - Virtual network: **adVNET**
   - Subnet: *a new subnet named* **bastionSubnet**
   - Address range: **10.0.255.0/24**
   - Public IP address: *a new IP address named* **az12001b-vm2-ip**
   - NIC network security group: **Basic**
   - Public inbound ports: **Allow selected ports**
   - Select inbound ports: **RDP (3389)**
   - Accelerated networking: **Off**
   - Place this virtual machine behind an existing load balancing solutions: **No**
   - Boot diagnostics: **Off**
   - OS guest diagnostics: **Off**

- System assigned managed identity: **Off**
- Login with AAD credentials (Preview): **Off**
- Enable auto-shutdown: **Off**
- Enable backup: **Off**
- Extensions: *None*
- Tags: *None*

4. Wait for the provisioning to complete. This should take a few minutes.
5. Connect to the newly provisioned Azure VM via RDP.
6. Within the RDP session to az12001b-vm2, ensure that you can establish RDP session to both az12001b-cl-vm0 and az12001b-cl-vm1 via their private IP addresses (10.0.1.4 and 10.0.1.5, respectively).

   Result: After you completed this exercise, you have provisioned Azure network resources necessary to support highly available SAP NetWeaver deployments

4.7 Exercise 4: Remove lab resources

Duration: 10 minutes

In this exercise, you will remove resources provisioned in this lab.

4.7.0.1 Task 1: Open Cloud Shell

1. At the top of the portal, click the **Cloud Shell** icon to open Cloud Shell pane and choose PowerShell as the shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the first exercise of this lab:

   ```
   $resourceGroupNamePrefix = 'az12001b-'
   ```

3. In the Cloud Shell pane, run the following command to list all resource groups you created in this lab:

   ```
   Get-AzResourceGroup | Where-Object {$_.ResourceGroupName -like "$resourceGroupNamePrefix*"} | Sele
   ```

4. Verify that the output contains only the resource groups you created in this lab. These groups will be deleted in the next task.

4.7.0.2 Task 2: Delete resource groups

1. In the Cloud Shell pane, run the following command to delete the resource groups you created in this lab

   ```
   Get-AzResourceGroup | Where-Object {$_.ResourceGroupName -like "$resourceGroupNamePrefix*"} | Remo
   ```

2. Close the Cloud Shell pane.

   Result: After you completed this exercise, you have removed the resources used in this lab.

5 AZ 120 Module 3: Implementing SAP on Azure

6 Lab 3a: Implement SAP architecture on Azure VMs running Linux

Estimated Time: 100 minutes

All tasks in this lab are performed from the Azure portal (including the Bash Cloud Shell session)

> **Note**: When not using Cloud Shell, the lab virtual machine must have Azure CLI installed **https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-windows?view=azure-cli-latest**.

Lab files: none

6.1 Scenario

In preparation for deployment of SAP NetWeaver on Azure, Adatum Corporation wants to implement a demo that will illustrate highly available implementation of SAP NetWeaver on Azure VMs running the SUSE distribution of Linux.

6.2 Objectives

After completing this lab, you will be able to:

- Provision Azure resources necessary to support a highly available SAP NetWeaver deployment
- Configure operating system of Azure VMs running Linux to support a highly available SAP NetWeaver deployment
- Configure clustering on Azure VMs running Linux to support a highly available SAP NetWeaver deployment

6.3 Requirements

- A Microsoft Azure subscription with the sufficient number of available DSv2 and Dsv3 vCPUs (four Standard_DS1_v2 VMs with 1 vCPU each and two Standard_D4s_v3 VMs with 4 vCPUs each) in an Azure region that supports availability zones
- A lab computer with an Azure Cloud Shell-compatible web browser and access to Azure

6.4 Exercise 1: Provision Azure resources necessary to support highly available SAP NetWeaver deployments

Duration: 30 minutes

In this exercise, you will deploy Azure infrastructure compute components necessary to configure Linux clustering. This will involve creating a pair of Azure VMs running Linux SUSE in the same availability set.

6.4.1 Task 1: Create a virtual network that will host a highly available SAP NetWeaver deployment.

1. From the lab computer, start a Web browser, and navigate to the Azure portal at https://portal.azure.com
2. If prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you will be using for this lab and the the Global Administrator role in the Azure AD tenant associated with your subscription.
3. In the Azure Portal, start a Bash session in Cloud Shell.

   Note: If this is the first time you are launching Cloud Shell in the current Azure subscription, you will be asked to create an Azure file share to persist Cloud Shell files. If so, accept the defaults, which will result in creation of a storage account in an automatically generated resource group.

4. In the Cloud Shell pane, run the following command to specify the Azure region that supports availability zones and where you want to create resources for this lab (replace `<region>` with the name of the Azure region which supports availablity zones):

   ```
   LOCATION='<region>'
   ```

   Note: Consider using **East US** or **East US2** regions for deployment of your resources.

   Note: Ensure to use the proper notation for the Azure region (short name which does not include a space, e.g. **eastus** rather than **US East**)

   Note: To identify Azure regions which support availability zones, refer to https://docs.microsoft.com/en-us/azure/availability-zones/az-region

5. In the Cloud Shell pane, run the following command to set the value of the variable `RESOURCE_GROUP_NAME` to the name of the resource group containing the resources you provisioned in the previous task:

   ```
   RESOURCE_GROUP_NAME='az12003a-sap-RG'
   ```

6. In the Cloud Shell pane, run the following command to create a resource group in the region you specified:

```
az group create --resource-group $RESOURCE_GROUP_NAME --location $LOCATION
```

7. In the Cloud Shell pane, run the following command to create a virtual network with a single subnet in the resource group you created:

```
VNET_NAME='az12003a-sap-vnet'

VNET_PREFIX='10.3.0.0/16'

SUBNET_NAME='sapSubnet'

SUBNET_PREFIX='10.3.0.0/24'

az network vnet create --resource-group $RESOURCE_GROUP_NAME --location $LOCATION --name $VNET_NAM
```

8. In the Cloud Shell pane, run the following command to identify the Resource Id of the subnet of the newly created virtual network:

```
az network vnet subnet list --resource-group $RESOURCE_GROUP_NAME --vnet-name $VNET_NAME --query "
```

9. Copy the resulting value to Clipboard. You will need it in the next task.

6.4.2 Task 2: Deploy Azure Resource Manager template provisioning Azure VMs running Linux SUSE that will host a highly available SAP NetWeaver deployment

1. On the lab computer, start a browser and browse to **https://github.com/Azure/azure-quickstart-templates/tree/master/sap-3-tier-marketplace-image-md**

   Note: Make sure to use Microsoft Edge or a third party browser. Do not use Internet Explorer.

2. On the page titled **SAP NetWeaver 3-tier compatible template using a Marketplace image - MD**, click **Deploy to Azure**. This will automatically redirect your browser to the Azure portal and display the **SAP NetWeaver 3-tier (managed disk)** blade.
3. On the **SAP NetWeaver 3-tier (managed disk)** blade, select **Edit template**.
4. On the **Edit template** blade, apply the following changes and select **Save**:
   - in the line **197**, replace `"dbVMSize": "Standard_E8s_v3",` with `"dbVMSize": "Standard_D4s_v3",`
   - in the line **198**, replace `"ascsVMSize": "Standard_D2s_v3",` with `"ascsVMSize": "Standard_DS1_v2",`
   - in the line **199**, replace `"diVMSize": "Standard_D2s_v3",` with `"diVMSize": "Standard_DS1_v2",`
5. On the **SAP NetWeaver 3-tier (managed disk)** blade, initiate deployment with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of the resource group you used in the previous task*
   - Location: *the same Azure region that you specified in the first task of this exercise*
   - SAP System Id: **I20**
   - Stack Type: **ABAP**
   - Os Type: **SLES 12**
   - Dbtype: **HANA**
   - Sap System Size: **Demo**
   - System Availability: **HA**
   - Admin Username: **student**
   - Authentication Type: **password**
   - Admin Password Or Key: **Pa55w.rd1234**
   - Subnet Id: *the value you copied into Clipboard in the previous task*
   - Availability Zones: **1,2**
   - Location: [**resourceGroup().location**]

- _artifacts Location: **https://raw.githubusercontent.com/Azure/azure-quickstart-templates/master/sap-3-tier-marketplace-image-md/**
- _artifacts Location Sas Token: *leave blank*

6. Do not wait for the deployment to complete but instead proceed to the next task.

   Note: If the deployment fails with the **Conflict** error message during deployment of the CustomScriptExtension component, use the following steps to remediate this issue:

- in the Azure portal, on the **Deployment** blade, review the deployment details and identify the VM(s) where the installation of the CustomScriptExtension failed
- in the Azure portal, navigate to the blade of the VM(s) you identified in the previous step, select **Extensions**, and from the **Extensions** blade, remove the CustomScript extension
- in the Azure portal, navigate to the **az12003a-sap-RG** resource group blade, select **Deployments**, select the link to the failed deployment, and select **Redeploy**, select the target resource group (**az12003a-sap-RG**) and provide the password for the root account (**Pa55w.rd1234**).

6.4.3 Task 3: Deploy a jump host

Note: Since Azure VMs you deployed in the previous task are not accessible from Internet, you will deploy an Azure VM running Windows Server 2019 Datacenter that will serve as a jump host.

1. From the lab computer, in the Azure portal, click **+ Create a resource**.
2. From the **New** blade, initiate creation of a new Azure VM based on the **Windows Server 2019 Datacenter** image.
3. Provision a Azure VM with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of a new resource group* **az12003a-dmz-RG**
   - Virtual machine name: **az12003a-vm0**
   - Region: *the same Azure region where you deployed Azure VMs in the previous tasks of this exercise*
   - Availability options: **No infrastructure redundancy required**
   - Image: **Windows Server 2019 Datacenter**
   - Size: **Standard DS1 v2*** or similar*
   - Username: **Student**
   - Password: **Pa55w.rd1234**
   - Public inbound ports: **Allow selected ports**
   - Select inbound ports: **RDP (3389)**
   - Already have a Windows license?: **No**
   - OS disk type: **Standard HDD**
   - Virtual network: **az12003a-sap-vnet**
   - Subnet: a new subnet named **bastionSubnet (10.3.255.0/24)**
   - Public IP: *a new IP address named* **az12003a-vm0-ip**
   - NIC network security group: **Basic**
   - Public inbound ports: **Allow selected ports**
   - Select inbound ports: **RDP (3389)**
   - Accelerated networking: **Off**
   - Place this virtual machine behind an existing load balancing solutions: **No**
   - Enable basic plan for free: **No**
   - Boot diagnostics: **Off**

- OS guest diagnostics: **Off**
- System assigned managed identity: **Off**
- Login with AAD credentials (Preview): **Off**
- Enable auto-shutdown: **Off**
- Enable backup: **Off**
- Extensions: *None*
- Tags: *None*

4. Wait for the provisioning to complete. This should take a few minutes.

   Result: After you completed this exercise, you have provisioned Azure resources necessary to support highly available SAP NetWeaver deployments

6.5 Exercise 2: Configure Azure VMs running Linux to support a highly available SAP NetWeaver deployment

Duration: 30 minutes

In this exercise, you will configure Azure VMs running SUSE Linux Enterprise Server to accommodate a highly available SAP NetWeaver deployment.

6.5.1 Task 1: Configure networking of the database tier Azure VMs.

Note: Before you start this task, ensure that the template deployments you initiated in the previous exercise have successfully completed.

1. From the lab computer, in the Azure portal, navigate to the blade of the **i20-db-0** Azure VM.
2. From the **i20-db-0** blade, navigate to its **Networking** blade.
3. From the **i20-db-0 - Networking** blade, navigate to the network interface of the i20-db-0.
4. From the blade of the network interface of the i20-db-0, navigate to its IP configurations blade and, from there, display its **ipconfig1** blade.
5. On the **ipconfig1** blade, set the private IP address to **10.3.0.20**, change its assignment to **Static** and save the change.
6. In the Azure portal, navigate to the blade of the **i20-db-1** Azure VM.
7. From the **i20-db-1** blade, navigate to its **Networking** blade.
8. From the **i20-db-1 - Networking** blade, navigate to the network interface of the i20-db-1.
9. From the blade of the network interface of the i20-db-1, navigate to its IP configurations blade and, from there, display its **ipconfig1** blade.
10. On the **ipconfig1** blade, set the private IP address to **10.3.0.21**, change its assignment to **Static** and save the change.

6.5.2 Task 2: Connect to the database tier Azure VMs.

1. From the lab computer, in the Azure portal, navigate to the **az12003a-vm0** blade.
2. From the **az12003a-vm0** blade, connect to the Azure VM az12003a-vm0 via Remote Desktop.
3. Within the RDP session to az12003a-vm0, in Server Manager, navigate to the **Local Server** view and turn off **IE Enhanced Security Configuration**.
4. Within the RDP session to az12003a-vm0, download and install PuTTY from **https://www.chiark.greenend.org.uk**
5. Use PuTTY to connect via SSH to **i20-db-0** Azure VM. Acknowledge the security alert and, when prompted, provide the following credentials:
   - Login as: **student**
   - Password: **Pa55w.rd1234**

6. Use PuTTY to connect via SSH to **i20-db-1** Azure VM with the same credentials.

6.5.3 Task 3: Examine the storage configuration of the database tier Azure VMs.

1. From within the PuTTY SSH session to i20-db-0 Azure VM, run the following command to elevate privileges:

   ```
   sudo su -
   ```

2. If prompted for the password, type **Pa55w.rd1234** and press the **Enter** key.
3. In the SSH session to i20-db-0, verify that all of the SAP HANA related volumes (including **/usr/sap**, **/hana/shared**, **/hana/backup**, **/hana/data**, and **/hana/logs**) are propertly mounted by running:

   ```
   df -h
   ```

4. Repeat the previous steps on the i20-db-1 Azure VM.

6.5.4 Task 4: Enable cross-node password-less SSH access

1. In the SSH session to i20-db-0, generate passphrase-less SSH key by running:

   ```
   ssh-keygen
   ```

2. When prompted, press **Enter** three times and then display the key by running:

   ```
   cat /root/.ssh/id_rsa.pub
   ```

3. Copy the value of the key into Clipboard.
4. In the SSH session to i20-db-1, create the file **/root/.ssh/authorized_keys** in the vi editor by running:

   ```
   vi /root/.ssh/authorized_keys
   ```

5. In the vi editor, paste the key you generated on i20-db-0.
6. Save the changes and close the editor.
7. In the SSH session to i20-db-1, generate passphrase-less SSH key by running:

   ```
   ssh-keygen
   ```

8. When prompted, press **Enter** three times and then display the key by running:

   ```
   cat /root/.ssh/id_rsa.pub
   ```

9. Copy the value of the key into Clipboard.
10. In the SSH session to i20-db-0, create the file **/root/.ssh/authorized_keys** in the vi editor by running:

    ```
    vi /root/.ssh/authorized_keys
    ```

11. In the vi editor, paste the key you generated on i20-db-1 starting from a new line.
12. Save the changes and close the editor.
13. To verify that the configuration on was successful, in the SSH session to i20-db-0, establish an SSH session as **root** from i20-db-0 to i20-db-1 by running:

    ```
    ssh root@i20-db-1
    ```

14. When prompted whether you are sure to continue connecting, type `yes` and press the **Enter** key.
15. Ensure that you are not prompted for the password.
16. Close the SSH session from i20-db-0 to i20-db-1 by running:

    ```
    exit
    ```

17. In the SSH session to i20-db-1, establish an SSH session as **root** from i20-db-1 to i20-db-0 by running:

    ```
    ssh root@i20-db-0
    ```

18. When prompted whether you are sure to continue connecting, type `yes` and press the **Enter** key.
19. Ensure that you are not prompted for the password.
20. Close the SSH session from i20-db-1 to i20-db-0 by running:

```
exit
```

6.5.5 Task 5: Add YaST packages, update the Linux operating system, and install HA Extensions

1. In the SSH session to i20-db-0, run the following to launch YaST:

   ```
   yast
   ```

2. In **YaST Control Center**, select **Software -> Add-On Products** and press **Enter**. This will load **Package Manager**.

3. On the **Installed Add-on Products** screen, verify that **Public Cloud Module** is already installed. Then, press **F9** twice to return to the shell prompt.

4. In the SSH session to i20-db-0, run the following to update operating system (when prompted, type **y** and press the **Enter** key):

   ```
   zypper update
   ```

5. In the SSH session to i20-db-0, run the following to install the packages required by cluster resources (when prompted, type **y** and press the **Enter** key):

   ```
   zypper in socat
   ```

6. In the SSH session to i20-db-0, run the following to install the azure-lb component required by cluster resources:

   ```
   zypper in resource-agents
   ```

7. In the SSH session to i20-db-0, open the file **/etc/systemd/system.conf** in the vi editor by running:

   ```
   vi /etc/systemd/system.conf
   ```

8. In the vi editor, replace `#DefaultTasksMax=512` with `DefaultTasksMax=4096`.

   > **Note**: In some cases, Pacemaker might create many processes, reaching the default limit imposed on their number and triggering a failover. This change increases the maximum number of allowed processes.

9. Save the changes and close the editor.

10. In the SSH session to i20-db-0, run the following to activate the configuration change:

    ```
    systemctl daemon-reload
    ```

11. In the SSH session to i20-db-0, run the following to install the fence agents package:

    ```
    zypper install fence-agents
    ```

12. In the SSH session to i20-db-0, run the following to install Azure Python SDK required by the fence agent (when prompted, type **y** and press the **Enter** key):

    ```
    zypper install python-azure-mgmt-compute
    ```

13. Repeat the previous steps in this task on i20-db-1.

    Result: After you completed this exercise, you have onfigured operating system of Azure VMs running Linux to support a highly available SAP NetWeaver deployment

6.6 Exercise 3: Configure clustering on Azure VMs running Linux to support a highly available SAP NetWeaver deployment

Duration: 30 minutes

In this exercise, you will configure clustering on Azure VMs running Linux to support a highly available SAP NetWeaver deployment.

6.6.1 Task 1: Configure clustering

1. Within the RDP session to az12003a-vm0, in the PuTTY-based SSH session to i20-db-0, run the following to initiate configuration of an HA cluster on i20-db-0:

   ```
   ha-cluster-init -u
   ```

2. When prompted, provide the following answers:
   - Do you want to continue anyway (y/n)?: **y**
   - Address for ring0 [10.3.0.20]: **ENTER**
   - Port for ring0 [5405]: **ENTER**
   - Do you wish to use SBD (y/n)?: **n**
   - Do you wish to configure a virtual IP address (y/n)?: **n**

     Note: The clustering setup generates an **hacluster** account with its password set to **linux**. You will change it later in this task.
3. Within the RDP session to az12003a-vm0, in the PuTTY-based SSH session to i20-db-1, run the following to join the HA cluster on i20-db-0 from i20-db-1:

   ```
   ha-cluster-join
   ```
4. When prompted, provide the following answers:
   - Do you want to continue anyway (y/n)? **y**
   - IP address or hostname of existing node (e.g.: 192.168.1.1) []: **i20-db-0**
   - Address for ring0 [10.3.0.21]: **ENTER**
5. In the PuTTY-based SSH session to i20-db-0, run the following to set the password of the **hacluster** account to **Pa55w.rd1234** (type the new password when prompted):

   ```
   passwd hacluster
   ```
6. Repeat the previous step on i20-db-1.

6.6.2 Task 2: Review corosync configuration

1. Within the RDP session to az12003a-vm0, in the PuTTY-based SSH session to i20-db-0, open the **/etc/corosync/corosync.conf** file by running:

   ```
   vi /etc/corosync/corosync.conf
   ```
2. In the vi editor, note the `transport: udpu` entry and the `nodelist` section:

```
[...]
   interface {
       [...]
   }
   transport:      udpu
}
nodelist {
   node {
     ring0_addr:     10.3.0.20
     nodeid:     1
   }
   node {
     ring0_addr:     10.3.0.21
     nodeid:     2
   }
}
logging {
    [...]
```

3. In the vi editor, replace the entry `token: 5000` with `token: 30000`.

   Note: This change allows for memory preserving maintenance. For more information, refer to Microsoft documentation regarding maintenance of virtual machines in Azure
4. Save the changes and close the editor.
5. Repeat the previous steps on i20-db-1.

6.6.3 Task 3: Identify the value of the Azure subscription Id and the Azure AD tenant Id

1. From the lab computer, in the browser window, in the Azure portal at **https://portal.azure.com**, ensure that you are signed in with the user account that has the Global Administrator role in the Azure AD tenant associated with your subscription.
2. In the Azure Portal, start a Bash session in Cloud Shell.
3. In the Cloud Shell pane, run the following command to identify the id of your Azure subscription and the id of the corresponding Azure AD tenant:

   ```
   az account show --query '{id:id, tenantId:tenantId}' --output json
   ```

4. Copy the resulting values to Notepad. You will need it in the next task.

6.6.4 Task 4: Create an Azure AD application for the STONITH device

1. In the Azure portal, navigate to the Azure Active Directory blade.
2. From the Azure Active Directory blade, navigate to the **App registrations** blade and then click **+ New registration**:
3. On the **Register an application** blade, specify the following settings, and click **Register**:
   - Name: **Stonith app**
   - Supported account type: **Accounts in this organizational directory only**
4. On the **Stonith app** blade, copy the value of **Application (client) ID** to Notepad. This will be referred to as **login_id** later in this exercise:
5. On the **Stonith app** blade, click **Certificates & secrets**.
6. On the **Stonith app - Certificates & secrets** blade, click **+ New client secret**.
7. In the **Add a client secret** pane, in the **Description** text box, type **STONITH app key**, in the **Expires** section, leave the default **In 1 year**, and then click **Add**.
8. Copy the resulting secret value to Notepad (this entry is displayed only once, after you click **Add**). This will be referred to as **password** later in this exercise:

6.6.5 Task 5: Grant permissions to Azure VMs to the service principal of the STONITH app

1. In the Azure portal, navigate to the blade of the **i20-db-0** Azure VM
2. From the **i20-db-0** blade, display the **i20-db-0 - Access control (IAM)** blade.
3. From the **i20-db-0 - Access control (IAM)** blade, add a role assignment with the following settings:
   - Role: **Virtual Machine Contributor**
   - Assign access to: **Azure AD user, group, or service principal**
   - Select: **Stonith app**
4. Repeat the previous steps to assign the Stonith app the Virtual Machine Contributor role to the **i20-db-1** Azure VM

6.6.6 Task 6: Configure the STONITH cluster device

1. Within the RDP session to az12003a-vm0, switch to the PuTTY-based SSH session to i20-db-0.
2. Within the RDP session to az12003a-vm0, in the PuTTY-based SSH session to i20-db-0, run the following commands (make sure to replace the `subscription_id`, `tenant_id`, `login_id,` and `password` placeholders with the values you identified in Exercise 3 Task 4:

   ```
   crm configure property stonith-enabled=true

   crm configure property concurrent-fencing=true

   crm configure primitive rsc_st_azure stonith:fence_azure_arm \
     params subscriptionId="subscription_id" resourceGroup="az12003a-sap-RG" tenantId="tenant_id" logi
   ```

```
  pcmk_monitor_retries=4 pcmk_action_limit=3 power_timeout=240 pcmk_reboot_timeout=900 \
  op monitor interval=3600 timeout=120

sudo crm configure property stonith-timeout=900
```

6.6.7 Task 7: Review clustering configuration on Azure VMs running Linux by using Hawk

1. Within the RDP session to az12003a-vm0, start Internet Explorer and navigate to **https://i20-db-0:7630**. This should display the SUSE Hawk sign-in page.

   Note: Ignore **This site is not secure** message.

1. On the SUSE Hawk sign in page, login by using the following credentials:

   - Username: **hacluster**
   - Password: **Pa55w.rd1234**

2. Verify that the cluster status is healthy. If you are seeing a message indicating that one of two cluster nodes is unclean, restart that node from the Azure portal.

   Result: After you completed this exercise, you have configured clustering on Azure VMs running Linux to support a highly available SAP NetWeaver deployment

6.7 Exercise 4: Remove lab resources

Duration: 10 minutes

In this exercise, you will remove resources provisioned in this lab.

6.7.0.1 Task 1: Open Cloud Shell

1. At the top of the portal, click the **Cloud Shell** icon to open Cloud Shell pane and choose Bash as the shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `RESOURCE_GROUP_PREFIX` to the prefix of the name of the resource group containing the resources you provisioned in this lab:

   ```
   RESOURCE_GROUP_PREFIX='az12003a-'
   ```

3. In the Cloud Shell pane, run the following command to list all resource groups you created in this lab:

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

4. Verify that the output contains only the resource group you created in this lab. This resource group with all of their resources will be deleted in the next task.

6.7.0.2 Task 2: Delete resource groups

1. In the Cloud Shell pane, run the following command to delete the resource group and their resources.

   ```
   az group list --query "[?starts_with(name,'$RESOURCE_GROUP_PREFIX')]".name --output tsv | xargs -L
   ```

2. Close the Cloud Shell pane.

   Result: After you completed this exercise, you have removed the resources used in this lab.

7 AZ 120 Module 3: Implementing SAP on Azure

8 Lab 3b: Implement SAP architecture on Azure VMs running Windows

Estimated Time: 150 minutes

All tasks in this lab are performed from the Azure portal (including a PowerShell Cloud Shell session)

> **Note**: When not using Cloud Shell, the lab virtual machine must have Az PowerShell module installed **https://docs.microsoft.com/en-us/powershell/azure/install-az-ps-msi?view=azps-2.8.0**.

Lab files: none

8.1 Scenario

In preparation for deployment of SAP NetWeaver on Azure, Adatum Corporation wants to implement a demo that will illustrate highly available implementation of SAP NetWeaver on Azure VMs running Windows Server 2016.

8.2 Objectives

After completing this lab, you will be able to:

- Provision Azure resources necessary to support a highly available SAP NetWeaver deployment
- Configure operating system of Azure VMs running Windows to support a highly available SAP NetWeaver deployment
- Configure clustering on Azure VMs running Windows to support a highly available SAP NetWeaver deployment

8.3 Requirements

- A Microsoft Azure subscription with the sufficient number of available DSv2 and Dsv3 vCPUs (four Standard_DS1_v2 VM with 1 vCPU and six Standard_D4s_v3 VMs with 4 vCPUs each) in an Azure region that supports availability zones
- A lab computer with an Azure Cloud Shell-compatible web browser and access to Azure

8.4 Exercise 1: Provision Azure resources necessary to support highly available SAP NetWeaver deployments

Duration: 60 minutes

In this exercise, you will deploy Azure infrastructure compute components necessary to configure Windows clustering. This will involve creating a pair of Azure VMs running Windows Server 2016 in the same availability set.

8.4.1 Task 1: Deploy a pair of Azure VMs running highly available Active Directory domain controllers by using an Azure Resource Manager template

1. From the lab computer, start a Web browser, and navigate to the Azure portal at https://portal.azure.com
2. If prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you will be using for this lab.
3. In the Azure portal interface, click **+ Create a resource**.
4. From the **New** blade, initiate creation of a new **Template deployment (deploy using custom templates)**
5. From the **Custom deployment** blade, in the **Load a GitHub quickstart template** drop-down list, select the entry **active-directory-new-domain-ha-2-dc-zones**, and click **Select template**.

   Note: Alternatively, you can launch the deployment by navigating to Azure Quickstart Templates page at https://github.com/Azure/azure-quickstart-templates, locating the template named **Create 2 new Windows VMs, a new AD Forest, Domain and 2 DCs in separate availability zones**, and initiating its deployment by clicking **Deploy to Azure** button.

6. On the blade **Create a new AD Domain with 2 DCs using Availability Zones**, specify the following settings and click **Purchase** to initiate the deployment:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of a new resource group* **az12003b-ad-RG**
   - Location: *an Azure region where you can deploy Azure VMs*

     Note: Consider using **East US** or **East US2** regions for deployment of your resources.

- Admin Username: **Student**
- Location: *the same Azure region you specified above*
- Password: **Pa55w.rd1234**
- Domain Name: **adatum.com**
- DnsPrefix: *accept the default value*
- Vm Size: **Standard D4S_v3**
- _artifacts Location: *accept the default value*
- _artifacts Location Sas Token: *leave blank*
- I agree to the terms and conditions stated above: *enabled*

Note: The deployment should take about 35 minutes. Wait for the deployment to complete before you proceed to the next task.

Note: If the deployment fails with the **Conflict** error message during deployment of the CustomScriptExtension component, use the following steps to remediate this issue:

- in the Azure portal, on the **Deployment** blade, review the deployment details and identify the VM(s) where the installation of the CustomScriptExtension failed
- in the Azure portal, navigate to the blade of the VM(s) you identified in the previous step, select **Extensions**, and from the **Extensions** blade, remove the CustomScript extension
- in the Azure portal, navigate to the **az12003b-sap-RG** resource group blade, select **Deployments**, select the link to the failed deployment, and select **Redeploy**, select the target resource group (**az12003b-sap-RG**) and provide the password for the root account (**Pa55w.rd1234**).

8.4.2 Task 2: Provision subnets that will host Azure VMs running highly available SAP NetWeaver deployment and the S2D cluster.

1. In the Azure Portal, navigate to the blade of the **az12003b-ad-RG** resource group.
2. On the **az12003b-ad-RG** resource group blade, in the list of resources, locate the **adVNET** virtual network and click its entry to display the **adVNET** blade.
3. From the **adVNET** blade, navigate to its **adVNET - Subnets** blade.
4. From the **adVNET - Subnets** blade, create a new subnet with the following settings:
   - Name: **sapSubnet**
   - Address ranges (CIDR block): **10.0.1.0/24**
5. From the **adVNET - Subnets** blade, create a new subnet with the following settings:
   - Name: **s2dSubnet**
   - Address ranges (CIDR block): **10.0.2.0/24**
6. In the Azure Portal, start a PowerShell session in Cloud Shell.

   Note: If this is the first time you are launching Cloud Shell in the current Azure subscription, you will be asked to create an Azure file share to persist Cloud Shell files. If so, accept the defaults, which will result in creation of a storage account in an automatically generated resource group.
7. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the resources you provisioned in the previous task:

   ```
   $resourceGroupName = 'az12003b-ad-RG'
   ```
8. In the Cloud Shell pane, run the following command to identify the virtual network created in the previous task:

   ```
   $vNetName = 'adVNet'

   $subnetName = 'sapSubnet'
   ```

9. In the Cloud Shell pane, run the following command to identify the Resource Id of the newly created subnet:

```
$vNet = Get-AzVirtualNetwork -ResourceGroupName $resourceGroupName -Name $vNetName

(Get-AzVirtualNetworkSubnetConfig -Name $subnetName -VirtualNetwork $vNet).Id
```

10. Copy the resulting value to Clipboard. You will need it in the next task.

8.4.3 Task 3: Deploy Azure Resource Manager template provisioning Azure VMs running Windows Server 2016 that will host a highly available SAP NetWeaver deployment

1. On the lab computer, in the Azure portal, search for and select **Template deployment (deploy using custom template)**.
2. On the **Custom deployment** blade, in the **Select a template (disclaimer)** drop-down list, type **sap-3-tier-marketplace-image-md** and click **Select template**.

   Note: Make sure to use Microsoft Edge or a third party browser. Do not use Internet Explorer.
3. On the **SAP NetWeaver 3-tier (managed disk)** blade, select **Edit template**.
4. On the **Edit template** blade, apply the following changes and select **Save**:
   - in the line **197**, replace `"dbVMSize": "Standard_E8s_v3",` with `"dbVMSize": "Standard_D4s_v3",`
   - in the line **198**, replace `"ascsVMSize": "Standard_D2s_v3",` with `"ascsVMSize": "Standard_DS1_v2",`
   - in the line **199**, replace `"diVMSize": "Standard_D2s_v3",` with `"diVMSize": "Standard_DS1_v2",`
5. Back on the **SAP NetWeaver 3-tier (managed disk)** blade, initiate deployment with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of a new resource group* **az12003b-sap-RG**
   - Location: *the same Azure region that you specified in the first task of this exercise*
   - SAP System Id: **I20**
   - Stack Type: **ABAP**
   - Os Type: **Windows Server 2016 Datacenter**
   - Dbtype: **SQL**
   - Sap System Size: **Demo**
   - System Availability: **HA**
   - Admin Username: **Student**
   - Authentication Type: **password**
   - Admin Password Or Key: **Pa55w.rd1234**
   - Subnet Id: *the value you copied into Clipboard in the previous task*
   - Availability Zones: **1,2**
   - Location: [**resourceGroup().location**]
   - _artifacts Location: **https://raw.githubusercontent.com/Azure/azure-quickstart-templates/master/sap-3-tier-marketplace-image-md/**
   - _artifacts Location Sas Token: *leave blank*
   - I agree to the terms and conditions stated above: *enabled*
6. Do not wait for the deployment to complete but instead proceed to the next task.

8.4.4 Task 5: Deploy the Scale-Out File Server (SOFS) cluster

In this task, you will deploy the scale-out file server (SOFS) cluster that will be hosting a file share for the SAP ASCS servers by using an Azure Resource Manager QuickStart template from GitHub available at **https://github.com/robotechredmond/301-storage-spaces-direct-md**.

1. On the lab computer, start a browser and browse to **https://github.com/robotechredmond/301-storage-spaces-direct-md**.

   Note: Make sure to use Microsoft Edge or a third party browser. Do not use Internet Explorer.

2. On the page titled **Use Managed Disks to Create a Storage Spaces Direct (S2D) Scale-Out File Server (SOFS) Cluster with Windows Server 2016**, click **Deploy to Azure**. This will automatically redirect your browser to the Azure portal and display the **Custom deployment** blade.

3. From the **Custom deployment** blade, initiate a deployment with the following settings:

   - Subscription: **Your Azure subscription name**.
   - Resource group: *the name of a new resource group* **az12003b-s2d-RG**
   - Region: *the same Azure region where you deployed Azure VMs in the previous tasks of this exercise*
   - Name Prefix: **i20**
   - Vm Size: **Standard D4S_v3**
   - Enable Accelerated Networking: **true**
   - Image Sku: **2016-Datacenter-Server-Core**
   - VM Count: **2**
   - VM Disk Size: **128**
   - VM Disk Count: **3**
   - Existing Domain Name: **adatum.com**
   - Admin Username: **Student**
   - Admin Password: **Pa55w.rd1234**
   - Existing Virtual Network RG Name: **az12003b-ad-RG**
   - Existing Virtual Network Name: **adVNet**
   - Existing Subnet Name: **s2dSubnet**
   - Sofs Name: **sapglobalhost**
   - Share Name: **sapmnt**
   - Scheduled Update Day: **Sunday**
   - Scheduled Update Time: **3:00 AM**
   - Realtime Antimalware Enabled: **false**
   - Scheduled Antimalware Enabled: **false**
   - Scheduled Antimalware Time: **120**
   - _artifacts Location: **Accept the default value**
   - _artifacts Location Sas Token: **Leave the default value**
   - I agree to the terms and conditions stated above: *enabled*

4. The deployment might take about 20 minutes. Do not wait for the deployment to complete but instead proceed to the next task.

8.4.5 Task 6: Deploy a jump host

Note: Since Azure VMs you deployed in the previous task are not accessible from Internet, you will deploy an Azure VM running Windows Server 2016 Datacenter that will serve as a jump host.

1. From the lab computer, in the Azure portal interface, click **+ Create a resource**.
2. From the **New** blade, initiate creation of a new Azure VM based on the **Windows Server 2019 Datacenter** image.
3. Provision a Azure VM with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of a new resource group* **az12003b-dmz-RG**
   - Virtual machine name: **az12003b-vm0**
   - Region: *the same Azure region where you deployed Azure VMs in the previous tasks of this exercise*
   - Availability options: **No infrastructure redundancy required**
   - Image: **Windows Server 2019 Datacenter**
   - Size: **Standard DS1 v2**
   - Username: **Student**
   - Password: **Pa55w.rd1234**
   - Public inbound ports: **Allow selected ports**
   - Select inbound ports: **RDP (3389)**
   - Already have a Windows license?: **No**
   - OS disk type: **Standard HDD**
   - Virtual network: **adVNET**
   - Subnet: *a new subnet named* **dmzSubnet (10.0.255.0/24)**
   - Public IP: *a new IP address named* **az12003b-vm0-ip**
   - NIC network security group: **Basic**
   - Public inbound ports: **Allow selected ports**
   - Select inbound ports: **RDP (3389)**
   - Accelerated networking: **Off**
   - Place this virtual machine behind an existing load balancing solutions: **No**
   - Boot diagnostics: **Off**
   - OS guest diagnostics: **Off**
   - System assigned managed identity: **Off**
   - Login with AAD credentials (Preview): **Off**
   - Enable auto-shutdown: **Off**
   - Enable backup: **Off**
   - Extensions: *None*
   - Tags: **None**
4. Wait for the provisioning to complete. This should take a few minutes.

Result: After you completed this exercise, you have provisioned Azure resources necessary to support highly available SAP NetWeaver deployments

8.5 Exercise 2: Configure operating system of Azure VMs running Windows to support a highly available SAP NetWeaver deployment

Duration: 60 minutes

In this exercise, you will configure operating system of Azure VMs running Windows Server to accommodate a highly available SAP NetWeaver deployment.

8.5.1 Task 1: Join Windows Server 2016 Azure VMs to the Active Directory domain.

Note: Before you start this task, ensure that the template deployments you initiated in the previous exercise have successfully completed.

1. In the Azure Portal, navigate to the blade of the virtual network named **adVNET**, which was provisioned automatically in the first exercise of this lab.
2. Display the **adVNET - DNS servers** blade and note that the virtual network is configured with the private IP addresses assigned to the domain controllers deployed in the first exercise of this lab as its DNS servers.
3. In the Azure Portal, start a PowerShell session in Cloud Shell.
4. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the resources you provisioned in the previous task:

   ```
   $resourceGroupName = 'az12003b-sap-RG'
   ```

5. In the Cloud Shell pane, run the following command, to join the Windows Server Azure VMs you deployed in the third task of the previous exercise to the **adatum.com** Active Directory domain:

   ```
   $location = (Get-AzResourceGroup -Name $resourceGroupName).Location

   $settingString = '{"Name": "adatum.com", "User": "adatum.com\\Student", "Restart": "true", "Option

   $protectedSettingString = '{"Password": "Pa55w.rd1234"}'

   $vmNames = @('i20-ascs-0','i20-ascs-1','i20-db-0','i20-db-1','i20-di-0','i20-di-1')

   foreach ($vmName in $vmNames) { Set-AzVMExtension -ResourceGroupName $resourceGroupName -Extension
   ```

8.5.2 Task 2: Examine the storage configuration of the database tier Azure VMs.

1. From the lab computer, in the Azure portal, navigate to the **az12003b-vm0** blade.
2. From the **az12003b-vm0** blade, connect to the Azure VM az12003b-vm0 via Remote Desktop. When prompted, provide the following credentials:
   - Login as: **student**
   - Password: **Pa55w.rd1234**
3. From the RDP session to az12003b-vm0, use Remote Desktop to connect to **i20-db-0.adatum.com** Azure VM. When prompted, provide the following credentials:
   - Login as: **ADATUM\Student**
   - Password: **Pa55w.rd1234**
4. Use Remote Desktop to connect to **i20-db-1.adatum.com** Azure VM with the same credentials.
5. Within the RDP session to i20-db-0.adatum.com, use File and Storage Services in the Server Manager to examine the disk configuration. Note that a single data disk has been configured via volume mounts to provide storage for database and log files.
6. Within the RDP session to i20-db-1.adatum.com, use File and Storage Services in the Server Manager to examine the disk configuration. Note that a single data disk has been configured via volume mounts to provide storage for database and log files.

8.5.3 Task 3: Prepare for configuration of Failover Clustering on Azure VMs running Windows Server 2016 to support a highly available SAP NetWeaver installation.

1. Within the RDP session to i20-db-0.adatum.com, start a Windows PowerShell ISE session and install Failover Clustering and Remote Administrative tools features by running the following on the pair of the ASCS and DB servers that will become nodes of the ASCS and SQL Server clusters, respectively:

   ```
   $nodes = @('i20-ascs-0','i20-ascs-1','i20-db-0','i20-db-1')

   Invoke-Command $nodes {Install-WindowsFeature Failover-Clustering -IncludeAllSubFeature -IncludeMa

   Invoke-Command $nodes {Install-WindowsFeature RSAT -IncludeAllSubFeature -Restart}
   ```

   Note: This might result in restart of the guest operating system of all four Azure VMs.

2. On the lab computer, in the Azure Portal, click **+ Create a resource**.
3. From the **New** blade, initiate creation of a new **Storage account** with the following settings:
   - Subscription: *the name of your Azure subscription*
   - Resource group: *the name of the resource group into which you deployed the Azure VMs which will host highly available SAP NetWeaver deployment*
   - Storage account name: *any unique name consisting of between 3 and 24 letters and digits*
   - Location: *the same Azure region where you deployed the Azure VMs in the previous exercise*
   - Performance: **Standard**
   - Account kind: **Storage (general purpose v1)**
   - Replication: **Locally-redundant storage (LRS)**
   - Connectivity method: **Public endpoint (all networks)**
   - Secure transfer required: **Enabled**
   - Large file shares: **Disabled**
   - Blob soft delete: **Disabled**
   - Hierarchical namespace: **Disabled**

8.5.4 Task 4: Configure Failover Clustering on Azure VMs running Windows Server 2016 to support a highly available database tier of the SAP NetWeaver installation.

1. If needed, from the RDP session to az12003b-vm0, use Remote Desktop to re-connect to **i20-db-0.adatum.com** Azure VM. When prompted, provide the following credentials:
   - Login as: **ADATUM\Student**
   - Password: **Pa55w.rd1234**
2. Within the RDP session to i20-db-0.adatum.com, in Server Manager, navigate to the **Local Server** view and turn off **IE Enhanced Security Configuration**.
3. Within the RDP session to i20-db-0.adatum.com, from the **Tools** menu in Server Manager, start **Active Directory Administrative Center**.
4. In Active Directory Administrative Center, create a new organizational unit named **Clusters** in the root of the adatum.com domain.
5. In Active Directory Administrative Center, move the computer accounts of i20-db-0 and i20-db-1 from the Computers container to the Clusters organizational unit.
6. Within the RDP session to i20-db-0, start a Windows PowerShell ISE session and create a new cluster by running the following:

   ```
   $nodes = @('i20-db-0','i20-db-1')

   New-Cluster -Name az12003b-db-cl0 -Node $nodes -NoStorage -StaticAddress 10.0.1.6
   ```

7. Within the RDP session to i20-db-0.adatum.com, switch to the **Active Directory Administrative Center** console.
8. In Active Directory Administrative Center, navigate to the **Clusters** organizational unit and display its **Properties** window.
9. In the **Clusters** organizational unit **Properties** window, navigate to the **Extensions** section, display the **Security** tab.
10. On the **Security** tab, click the **Advanced** button to open the **Advanced Security Settings for Clusters** window.
11. On the **Permissions** tab of the **Advanced Security Settings for Computers** window, click **Add**.
12. In the **Permission Entry for Clusters** window, click **Select Principal**
13. In the **Select User, Service Account or Group** dialog box, click **Object Types**, enable the checkbox next to the **Computers** entry, and click **OK**.
14. Back in the **Select User, Computer, Service Account or Group** dialog box, in the **Enter the object name to select**, type **az12003b-db-cl0** and click **OK**.
15. In the **Permission Entry for Clusters** window, ensure that **Allow** appears in the **Type** drop-down list. Next, in the **Applies to** drop-down list, select **This object and all descendant objects**. In the **Permissions** list, select the **Create Computer objects** and **Delete Computer objects** checkboxes, and click **OK** twice.
16. Within the Windows PowerShell ISE session, install the Az PowerShell module by running the following:

```
Install-PackageProvider -Name NuGet -Force

Install-Module -Name Az -Force
```

17. Within the Windows PowerShell ISE session, authenticate by using your Azure AD credentials by running the following:

```
Add-AzAccount
```

> **Note**: When prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you are using for this lab.

18. Within the Windows PowerShell ISE session, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the storage account you provisioned in the previous task:

```
$resourceGroupName = 'az12003b-sap-RG'
```

19. Within the Windows PowerShell ISE session, run the following to set the Cloud Witness quorum of the new cluster:

```
$cwStorageAccountName = (Get-AzStorageAccount -ResourceGroupName $resourceGroupName)[0].StorageAcc

$cwStorageAccountKey = (Get-AzStorageAccountKey -ResourceGroupName $resourceGroupName -Name $cwSto

Set-ClusterQuorum -CloudWitness -AccountName $cwStorageAccountName -AccessKey $cwStorageAccountKey
```

20. To verify the resulting configuration, within the RDP session to i20-db-0.adatum.com, from the **Tools** menu in Server Manager, start **Failover Cluster Manager**.
21. In the **Failover Cluster Manager** console, review the **az12003b-db-cl0** cluster configuration, including its nodes, as well as is witness and network settings. Note that the cluster does not have any shared storage.

8.5.5 Task 6: Configure Failover Clustering on Azure VMs running Windows Server 2016 to support a highly available ASCS tier of the SAP NetWeaver installation.

Note: Ensure that the deployment of the S2D cluster you initiated in task 4 of exercise 1 has successfully completed before starting this task.

1. From the RDP session to az12003b-vm0, use Remote Desktop to connect to **i20-ascs-0.adatum.com** Azure VM. When prompted, provide the following credentials:

- Login as: **ADATUM\Student**
- Password: **Pa55w.rd1234**

2. Within the RDP session to i20-ascs-0.adatum.com, in Server Manager, navigate to the **Local Server** view and turn off **IE Enhanced Security Configuration**.
3. Within the RDP session to i20-ascs-0.adatum.com, from the **Tools** menu in Server Manager, start **Active Directory Administrative Center**.
4. In Active Directory Administrative Center, navigate to the **Computers** container.
5. In Active Directory Administrative Center, move the computer accounts of i20-ascs-0 and i20-ascs-1 from the Computers container to the Clusters organizational unit.
6. Within the RDP session to i20-ascs-0.adatum.com, start a Windows PowerShell ISE session and create a new cluster by running the following:

```
$nodes = @('i20-ascs-0','i20-ascs-1')

New-Cluster -Name az12003b-ascs-cl0 -Node $nodes -NoStorage -StaticAddress 10.0.1.7
```

7. Within the RDP session to i20-ascs-0.adatum.com, switch to the **Active Directory Administrative Center** console.
8. In Active Directory Administrative Center, navigate to the **Clusters** organizational unit and display its **Properties** window.
9. In the **Clusters** organizational unit **Properties** window, navigate to the **Extensions** section, display the **Security** tab.
10. On the **Security** tab, click the **Advanced** button to open the **Advanced Security Settings for Clusters** window.
11. On the **Permissions** tab of the **Advanced Security Settings for Computers** window, click **Add**.
12. In the **Permission Entry for Clusters** window, click **Select Principal**
13. In the **Select User, Service Account or Group** dialog box, click **Object Types**, enable the checkbox next to the **Computers** entry, and click **OK**.
14. Back in the **Select User, Computer, Service Account or Group** dialog box, in the **Enter the object name to select**, type **az12003b-ascs-cl0** and click **OK**.
15. In the **Permission Entry for Clusters** window, ensure that **Allow** appears in the **Type** drop-down list. Next, in the **Applies to** drop-down list, select **This object and all descendant objects**. In the **Permissions** list, select the **Create Computer objects** and **Delete Computer objects** checkboxes, and click **OK** twice.
16. Within the Windows PowerShell ISE session, install the Az PowerShell module by running the following:

```
Install-PackageProvider -Name NuGet -Force

Install-Module -Name Az -Force
```

17. Within the Windows PowerShell ISE session, authenticate by using your Azure AD credentials by running the following:

```
Add-AzAccount
```

> **Note**: When prompted, sign in with the work or school or personal Microsoft account with the owner or contributor role to the Azure subscription you are using for this lab.

18. Within the Windows PowerShell ISE session, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the storage account you provisioned earlier in this exercise:

```
$resourceGroupName = 'az12003b-sap-RG'
```

19. Within the Windows PowerShell ISE session, run the following to set the Cloud Witness quorum of the cluster:

```
$cwStorageAccountName = (Get-AzStorageAccount -ResourceGroupName $resourceGroupName)[0].StorageAcc

$cwStorageAccountKey = (Get-AzStorageAccountKey -ResourceGroupName $resourceGroupName -Name $cwSto

Set-ClusterQuorum -CloudWitness -AccountName $cwStorageAccountName -AccessKey $cwStorageAccountKey
```

20. To verify the resulting configuration, Within the RDP session to i20-ascs-0.adatum.com, from the **Tools** menu in Server Manager, start **Failover Cluster Manager**.
21. In the **Failover Cluster Manager** console, review the **az12003b-ascs-cl0** cluster configuration, including its nodes, as well as is witness and network settings. Note that the cluster does not have any shared storage.

8.5.6 Task 7: Set permissions on the \\GLOBALHOST\sapmnt share

In this task, you will set share-level permissions on the **\\GLOBALHOST\sapmnt** share.

Note: By default, the Full Control permissions are granted only to the ADATUM\Student account.

1. Within the Remote Desktop session to i20-ascs-0.adatum.com, from the **Windows PowerShell ISE** window, run the following:

```
$remoteSession = New-CimSession -ComputerName SAPGLOBALHOST

Grant-SmbShareAccess -Name sapmnt -AccountName 'ADATUM\Domain Admins' -AccessRight Full -CimSessio
```

8.5.7 Task 8: Configure operating system prerequisites for installing SAP NetWeaver ASCS and database components

1. Within the Remote Desktop session to i20-ascs-0.adatum.com, from the Windows PowerShell ISE session, run the following to configure registry entries required to faciliate the installation of SAP ASCS components and the use of virtual names:

```
$nodes = ('i20-db-0','i20-db-1')

Invoke-Command $nodes {
    $registryPath = 'HKLM:\SYSTEM\CurrentControlSet\Services\lanmanworkstation\parameters'
    $registryEntry = 'DisableCARetryOnInitialConnect'
    $registryValue = 1
    New-ItemProperty -Path $registryPath -Name $registryEntry -Value $registryValue -PropertyType
}

Invoke-Command $nodes {
    $registryPath = 'HKLM:\SYSTEM\CurrentControlSet\Control\LSA'
    $registryEntry = 'DisableLoopbackCheck'
    $registryValue = 1
    New-ItemProperty -Path $registryPath -Name $registryEntry -Value $registryValue -PropertyType
}

Invoke-Command $nodes {
    $registryPath = 'HKLM:\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters'
    $registryEntry = 'DisableStrictNameChecking'
    $registryValue = 1
    New-ItemProperty -Path $registryPath -Name $registryEntry -Value $registryValue -PropertyType
}
```

Result: After you completed this exercise, you have configured operating system of Azure VMs running Windows to support a highly available SAP NetWeaver deployment

8.6 Exercise 3: Remove lab resources

Duration: 10 minutes

In this exercise, you will remove resources provisioned in this lab.

8.6.0.1 Task 1: Open Cloud Shell

1. At the top of the portal, click the **Cloud Shell** icon to open Cloud Shell pane and choose PowerShell as the shell.
2. In the Cloud Shell pane, run the following command to set the value of the variable `$resourceGroupName` to the name of the resource group containing the pair of **Windows Server 2019 Datacenter** Azure VMs you provisioned in the first exercise of this lab:

   ```
   $resourceGroupNamePrefix = 'az12003b-'
   ```

3. In the Cloud Shell pane, run the following command to list all resource groups you created in this lab:

   ```
   Get-AzResourceGroup | Where-Object {$_.ResourceGroupName -like "$resourceGroupNamePrefix*"} | Sele
   ```

4. Verify that the output contains only the resource groups you created in this lab. These groups will be deleted in the next task.

8.6.0.2 Task 2: Delete resource groups

1. In the Cloud Shell pane, run the following command to delete the resource groups you created in this lab

   ```
   Get-AzResourceGroup | Where-Object {$_.ResourceGroupName -like "$resourceGroupNamePrefix*"} | Remo
   ```

2. Close the Cloud Shell pane.

   Result: After you completed this exercise, you have removed the resources used in this lab.

9 AZ 120: Lab prerequisites

9.1 vCPU core requirements

- To complete the last lab of this course, you will need a Microsoft Azure subscription with at least 28 vCPU available in the Azure region that supports availability zones where the Azure VMs deployed in this lab will reside.
  - 4 x Standard_DS1_v2 (1 vCPUs each) = 4
  - 6 x Standard_D4s_v3 (4 vCPUs each) = 24

    Note: Consider using **East US** or **East US2** regions for deployment of your resources.

    Note: To identify the Azure regions that support availability zones, refer to [https://docs.microsoft.com/en-us/azure/availability-zones/az-overview]https://docs.microsoft.com/en-us/azure/availability-zones/az-overview

While the vCPU requirements for the first three labs of this course are lower, we recommend that you request increase of quotas to satisfy requirements for all of the labs, since the process of increasing quotas might take some time (even though quota increase requests are typically completed during the same business day).

9.2 Before the hands-on lab

Timeframe: 120 minutes

9.2.1 Task 1: Validate sufficient number of vCPU cores

1. In the Azure portal at http://portal.azure.com,
2. In the Azure Portal, start a PowerShell session in Cloud Shell.

   Note: If this is the first time you are launching Cloud Shell in the current Azure subscription, you will be asked to create an Azure file share to persist Cloud Shell files. If so, accept the defaults, which will result in creation of a storage account in an automatically generated resource group.

3. In the Azure portal, in the **Cloud Shell** pane, at the PowerShell prompt, run the following: where `<Azure_region>` designates the target Azure region that you intend to use for this lab (e.g. `eastus`):

```
Get-AzVMUsage -Location '<Azure_region>' | Where-Object {$_.Name.Value -eq 'StandardDSv3Family'}

Get-AzVMUsage -Location '<Azure_region>' | Where-Object {$_.Name.Value -eq 'StandardDSv2Family'}
```

Note: To identify the names of Azure regions, in the **Cloud Shell**, at the Bash prompt, run `(Get-AzLocation).Location`

4. Review the current value and the limit entries in the output of the commands executed in the previous step and ensure that you have sufficient number of vCPUs in the target Azure region.
5. If the number of vCPUs is not sufficient, in the Azure portal, navigate back to the subscription blade, and click **Usage + quotas**.
6. On the subscription's **Usage + quotas** blade, click **Request Increase**.
7. On the **Basic** blade, specify the following and click **Next**:
   - Issue type: **Service and subscription limits (quotas)**
   - Subscription: the name of the Azure subscription you will be using in this lab
   - Quota type: **Compute/VM (cores/vCPUs) subscription limit increases**
8. On the **Details** blade, click the **Provide details** blade.
9. On the **Quota details** blade, specify the following and click **Save and continue**:
   - Deployment model: **Resource Manager**
   - Location: the target Azure region you intend to use in this lab
   - SKU family: **DSv3 Series** and **DSv2 Series**
10. On the **Details** blade, specify the new limit for each SKU series and click **Next: Review + create**:
    - Severity: **C - Minimal impact**
    - Preferred contact method: choose your preferred option and provide contact details
11. On the **Review + create** blade, click **Create**

    Note: Quota increase requests are typically completed during the same business day.