Lab

9

Preventing Accidental Deletion of Resources Using Azure Resource Locking

Azure Resource Locking helps to protect the resources deployed in Azure from accidental deletion. Resource Locking can be applied to individual resources or resource groups.

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Scenario

Azure RBAC (Role-Based Access Control) is used to prevent access to the resources in Microsoft Azure. Implementing RBAC restricts access, but it is not that effective in all situations. If a user with full access accidentally deletes any critical resources, it will affect the cloud operations. To prevent accidental deletion, malicious changes, or modification to the critical resources, you should lock the Azure resource group or resource. In this lab, you will learn how to lock the Azure Resource Group to protect it from accidental deletion.

Lab Objectives

In this lab, you will learn how to create a Resource Group and Resource, lock the Resource Group, lock the Resource, try to delete the resource after the lock is enabled, delete Lock on the Resource Group, and then delete the Resource using Azure PowerShell.

In this lab, you will:

- Create a Resource Group and Resource in the form of a Virtual Machine
- Lock the Resource Group and Resource
- Unlock Resource Group and try to delete the Resource

Lab Environment

To perform this lab, you need the following:

- Admin Machine VM
- Registered Microsoft Azure account.

Lab Duration

Time: 15 minutes

Overview of Azure Resource Locking

Azure Resource Locking adds an extra layer of security by preventing the accidental deletion of the resources. Azure Resource Lock can be applied to resources or resource groups. The two types of Azure Resource Locks are

- **CanNotDelete**: In this type of lock, modifications can be performed on the resources, but the resources cannot be deleted.
- **ReadOnly**: In this type of resource lock, both modification and deletion of resources cannot be done.

To safeguard the resource group or resource from accidental deletion or modification in Azure, the cloud security engineer should lock the resource group or resource. The lock that is applied on the parent scope will be inherited to all the resources that are within that scope. If you add any resource later, then the resource will inherit the lock from the parent. You can apply the lock at the resource group level or the resource level. Once the lock is applied to the resources or resource groups, it can only be deleted once the resource lock is removed, thus preventing accidental deletion. As a part of the Azure Governance Strategy, you can use the resource lock to safeguard your cloud resources. It is recommended to apply the lock at the resource group level.

Lab Tasks

TASK 1

Creating a Resource Group and Resource

Note: Web applications using cloud environments may undergo frequent updates. For this lab, since we are working on a cloud-based environment (i.e., Azure), the application interface may be updated with time. Hence, in case you happen to work on an updated version of Azure, the user interface you see on the application might differ from what you see in the lab. Consequently, the steps and screenshots demonstrated in this lab might also differ.

Note: Before starting this lab, you should create an Azure Free Account using the following link: https://azure.microsoft.com/free, in case you have already

not created it for the previous module. Once the registration is complete, perform the following tasks:

Note: You can also use any existing Azure account but be aware that it may incur significant charges to your account.

1. Launch the **Admin Machine** VM. Log in with the following credentials: user **Admin** and password **admin@123.**

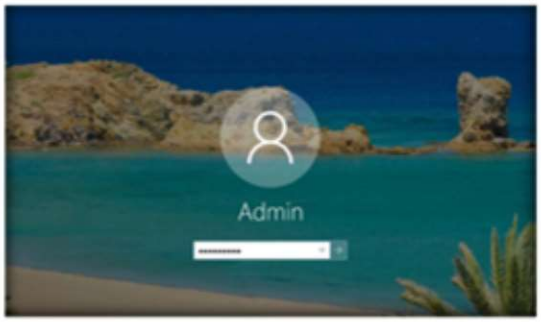

FIGURE 4.9.1: Launch Admin Machine and Log in

2. To open the browser, double-click on the **Google Chrome** icon on the desktop.

FIGURE 4.9.2: Navigating to the Chrome Browser from Taskbar

3. Go to the address bar, type **https://azure.microsoft.com/en-in/account/**, and press **Enter**.

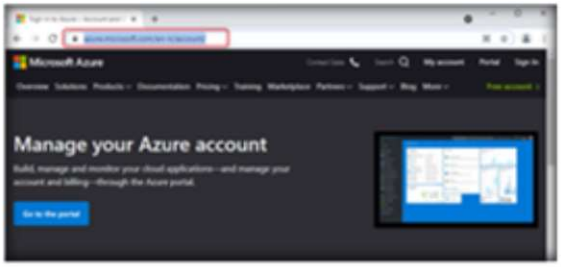

FIGURE 4.9.3 Entering the URL of Microsoft Azure

4. The **Microsoft Azure** page will appear. Click on **Portal.**

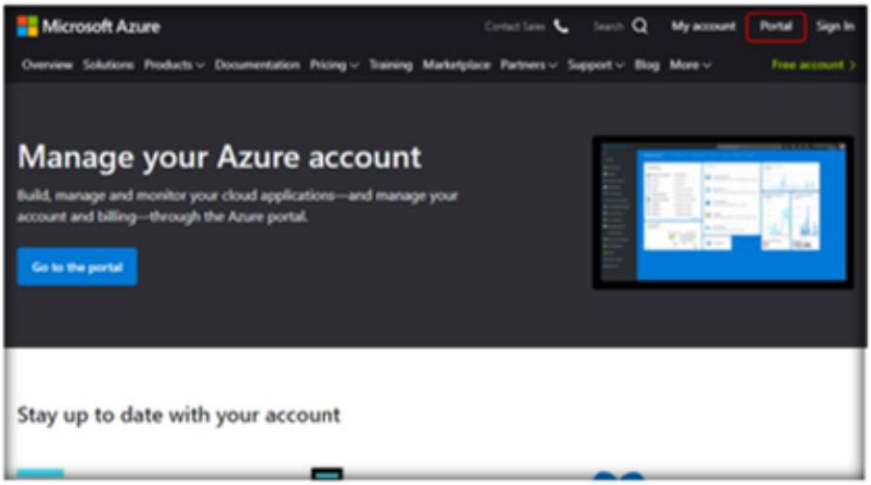

FIGURE 4.9.4: Sign in to Azure Portal

5. On the Sign-in page, enter the **Account ID** and click on **Next.**

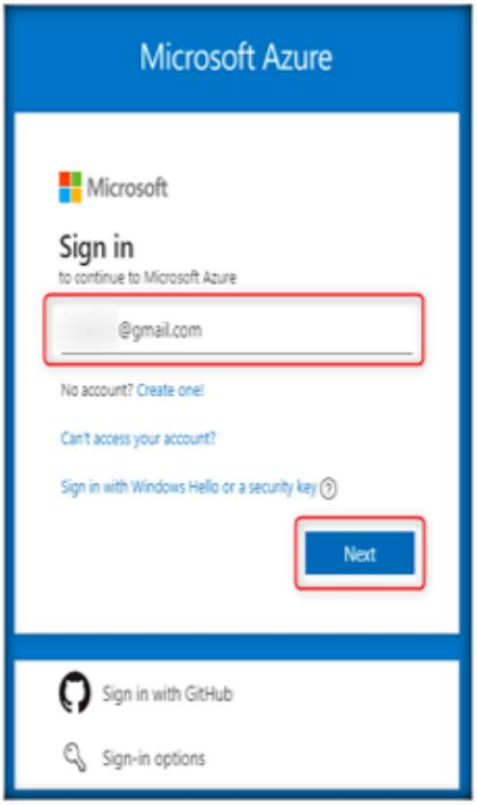

FIGURE 4.9.5: Entering Account ID to continue

6. In the next window, enter the **password** and click on **Sign in**.

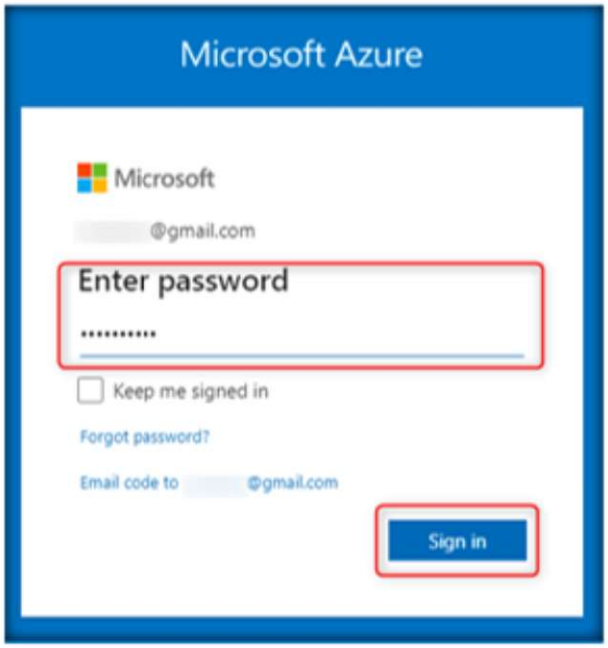

FIGURE 4.9.6: Entering the login Password

7. In the Azure portal, click on the **Cloud Shell** button in the top navigation bar.

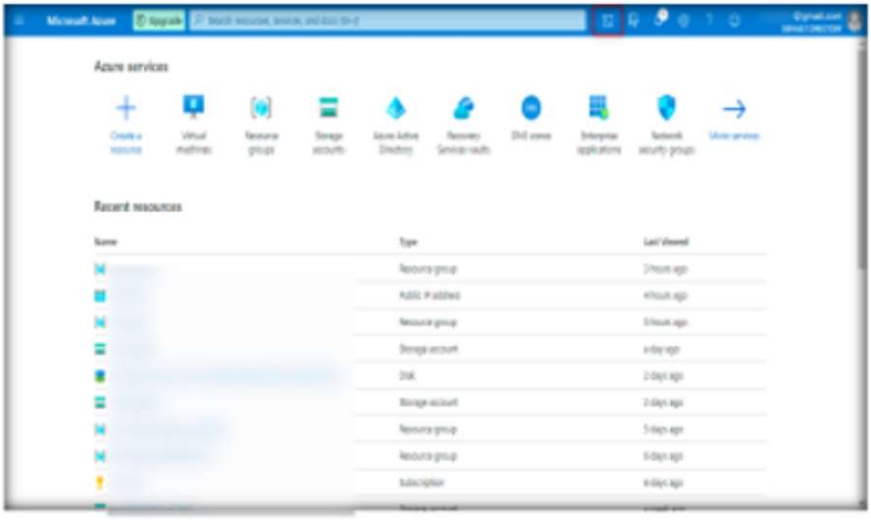

FIGURE 4.9.7: Selecting Cloud Shell in Azure Portal

8. Select the **PowerShell** environment from the dropdown in the top left side of the cloud shell.

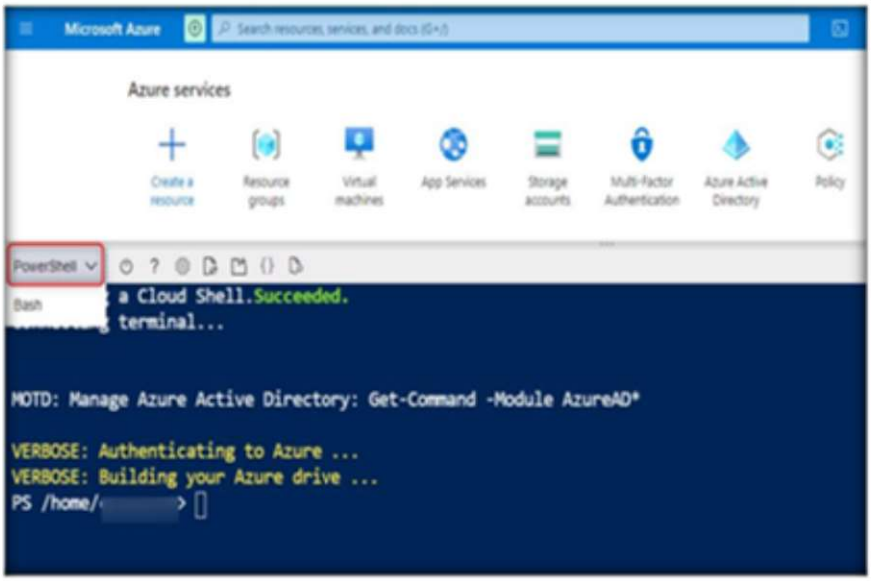

FIGURE 4.9.8: Choosing PowerShell Environment

9. Run the following command to create a new resource group.

```
New-AzResourceGroup -Name "myResourceGroup2556"
-Location "EastUS"
```

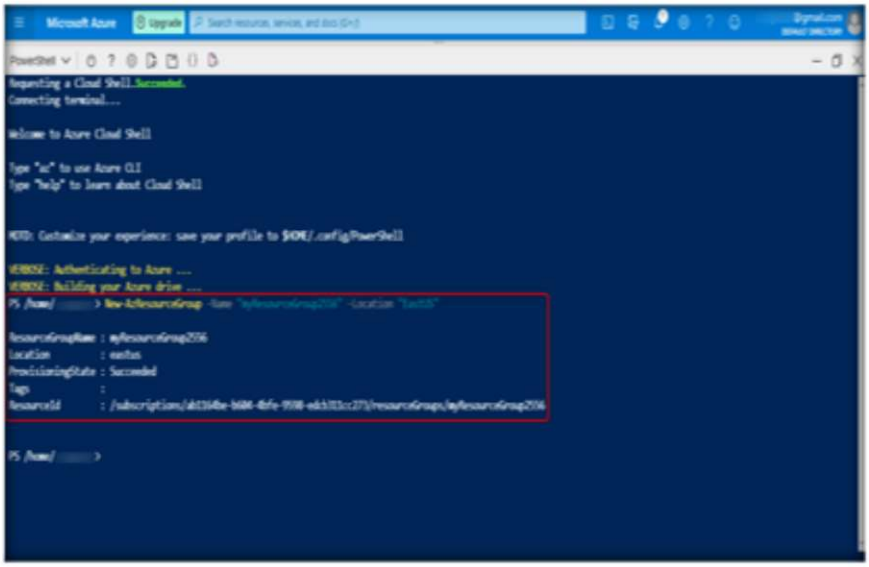

FIGURE 4.9.9: Creating a Resource Group

10. To create a resource i.e., virtual machine, enter the following command.

```
New-AzVm `
    -ResourceGroupName "myResourceGroup2556" `
    -Name "myVM2556" `
    -Location "East US" `
    -VirtualNetworkName "myVnet" `
    -SubnetName "mySubnet" `
    -SecurityGroupName "myNetworkSecurityGroup" `
    -PublicIpAddressName "myPublicIpAddress" `
-OpenPorts 80,3389
```

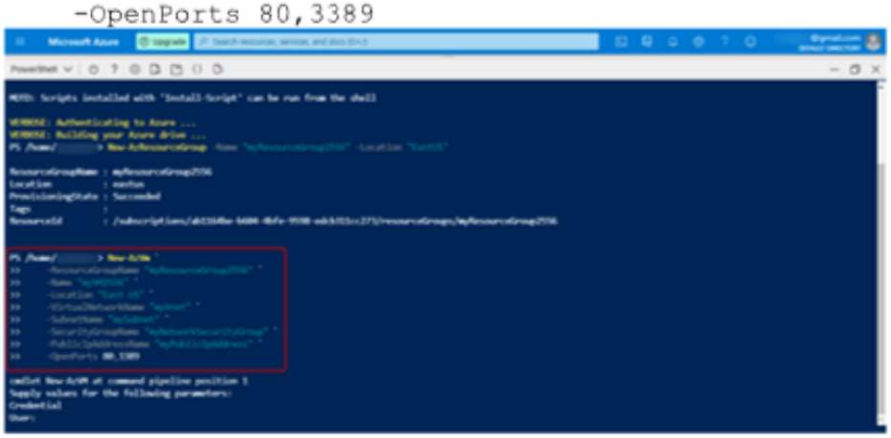

FIGURE 4.9.10: Creating a VM

11. You will have to enter a user name and password for the VM. Enter the User as **ccseuser** and a **Password** of your choice and press **Enter**.

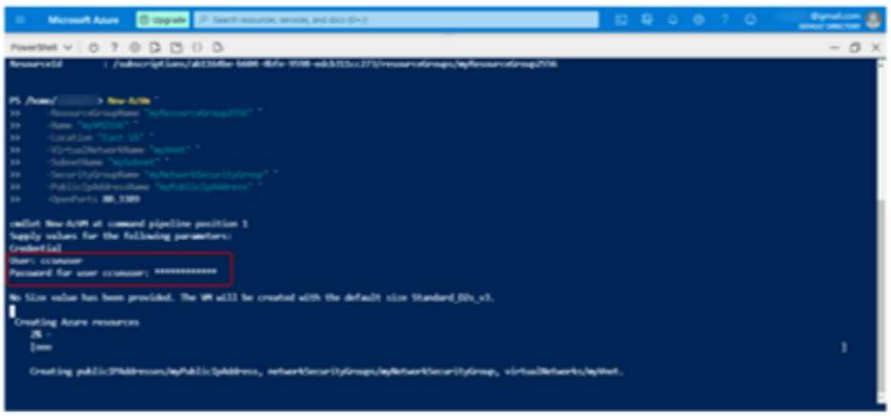

FIGURE 4.9.11: Entering the User and Password

12. After few minutes, your VM will get deployed successfully.

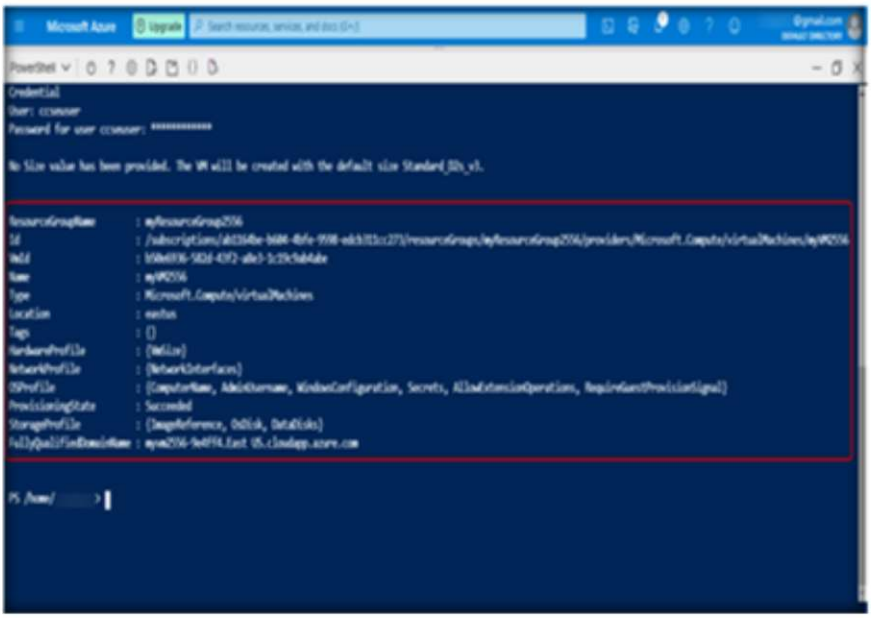

FIGURE 4.9.12: Successful Deployment of VM

13. Now, run the following command to determine the type of resource.

```
Get-AzResource
```

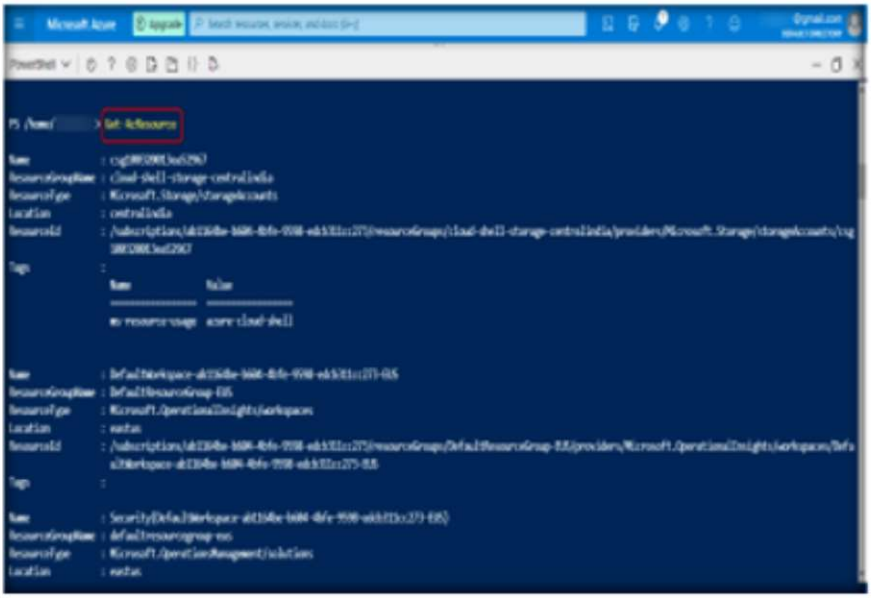

FIGURE 4.9.13: Fetching all the Resources in Azure

14. Scroll down and find the **Name** of the resource you have just created (**myVM2566**) and the **Resource Type** as **Microsoft.Compute/virtualMachines**, as shown in the screenshot. Copy the Name of the resource (**myVM2556**),ResourceGroupName (**myResourceGroup2556**), and ResourceType (**Microsoft.Compute/virtualMachines**) and paste in notepad.

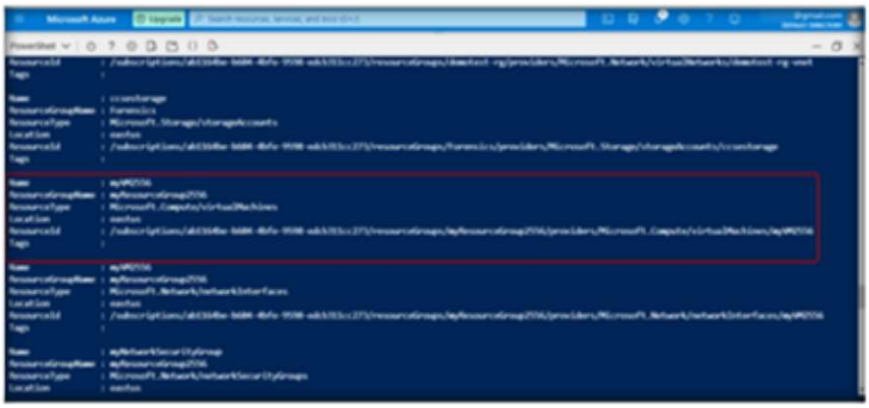

FIGURE 4.9.14: Determining the Type of Resource

TASK 2

Lock the Resource Group and Resource

15. Now, to lock the resource, you need the resource's name, resource type, and resource group that you have copied to notepad in the previous step. First, lock the resource with the **CanNotDelete** lock level. Run the following command to lock the resource.

```
New-AzResourceLock -LockLevel CanNotDelete -
LockName LockSite -ResourceName myVM2556 -
ResourceType
Microsoft.Compute/virtualMachines -
ResourceGroupName myResourceGroup2556
```

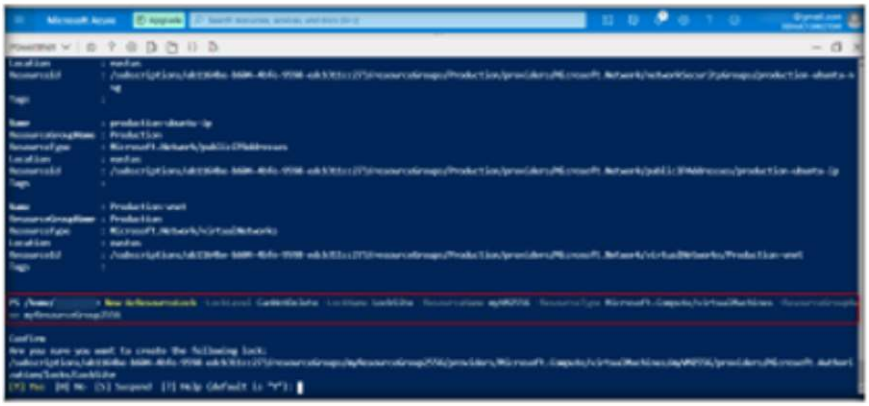

FIGURE 4.9.15: Locking the Resource

16. Type **Y** and press the **enter** button. The resource gets successfully locked.

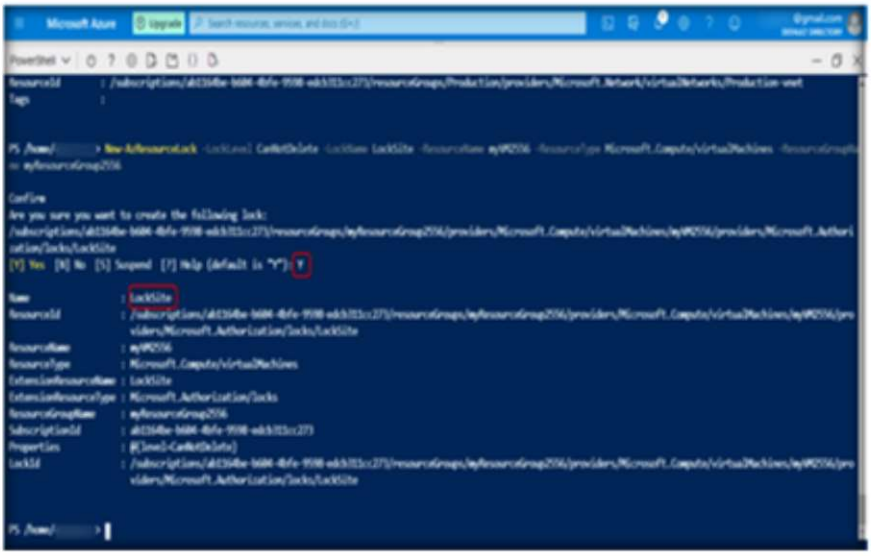

FIGURE 4.9.16: Resource is Successfully Locked

17. Next, you need to lock the resource group. Run the following command to lock the resource group with **CanNotDelete** lock level.

```
New-AzResourceLock -LockName LockGroup -
LockLevel CanNotDelete -ResourceGroupName
myResourceGroup2556
```

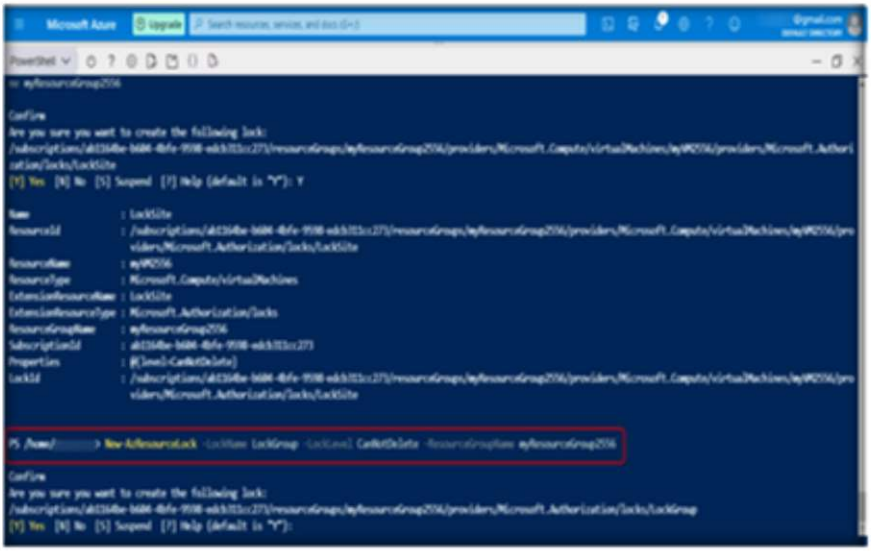

FIGURE 4.9.17: Locking the Resource Group

18. Type **Y** and press the **enter** button. Your resource group gets successfully locked.

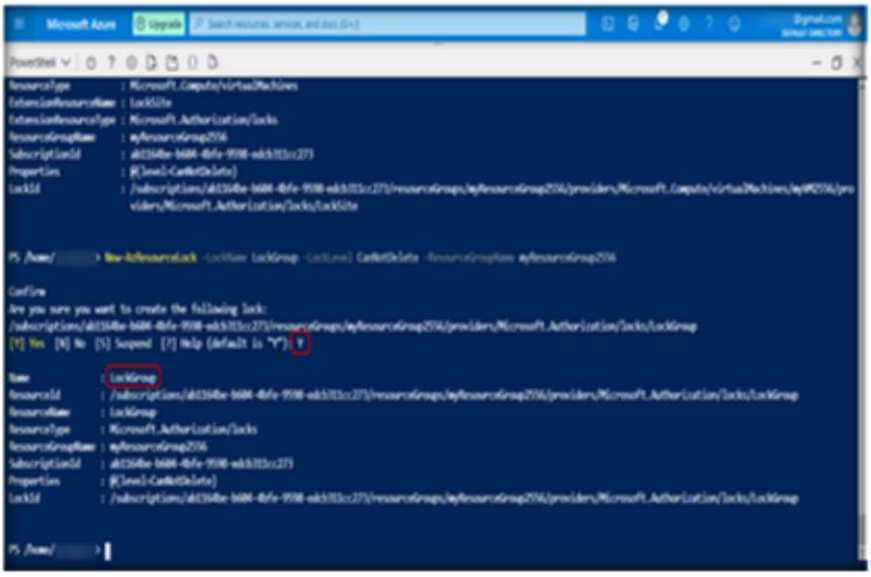

FIGURE 4.9.18: Resource Group is Successfully Locked

19. Run the following command to get information about all the locks in your subscription.

```
Get-AzResourceLock
```

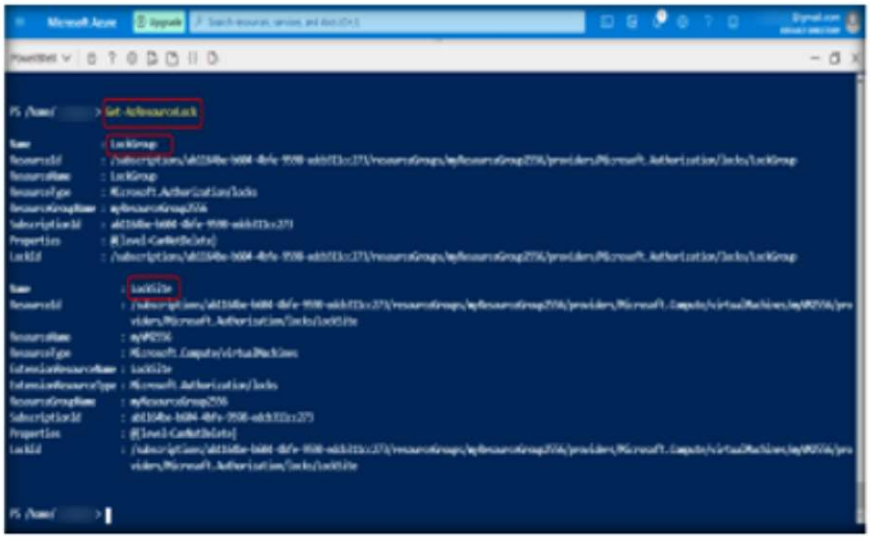

FIGURE 4.9.19: Fetching Information about all the Locks in the Subscription

20. Now you can try to delete the VM which you have locked to confirm whether the deployed resource and resource group lock is functioning. Run the following command to delete the VM that we have locked.

```
Remove-AzVM -ResourceGroupName
"myResourceGroup2556" -Name "myVM2556"
```

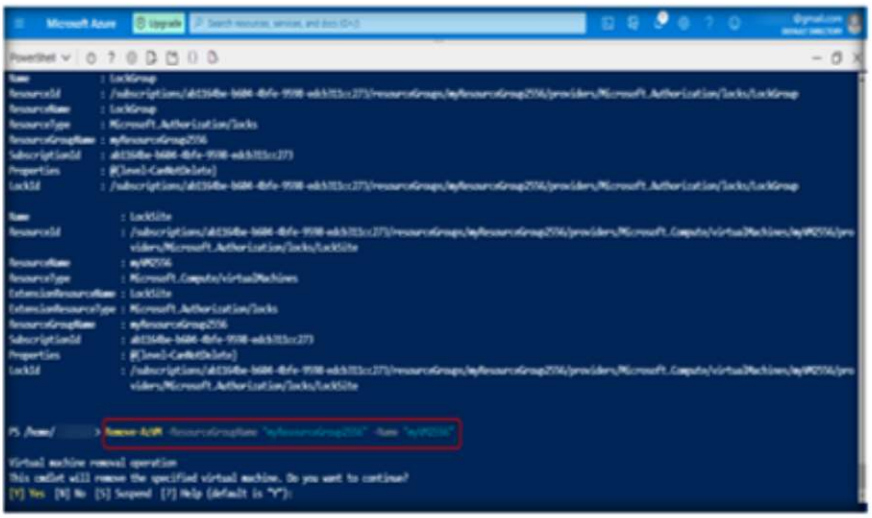

FIGURE 4.9.20: Deleting the VM

21. Type **Y** to confirm the removal of the virtual machine. You will observe an **error** in removal.

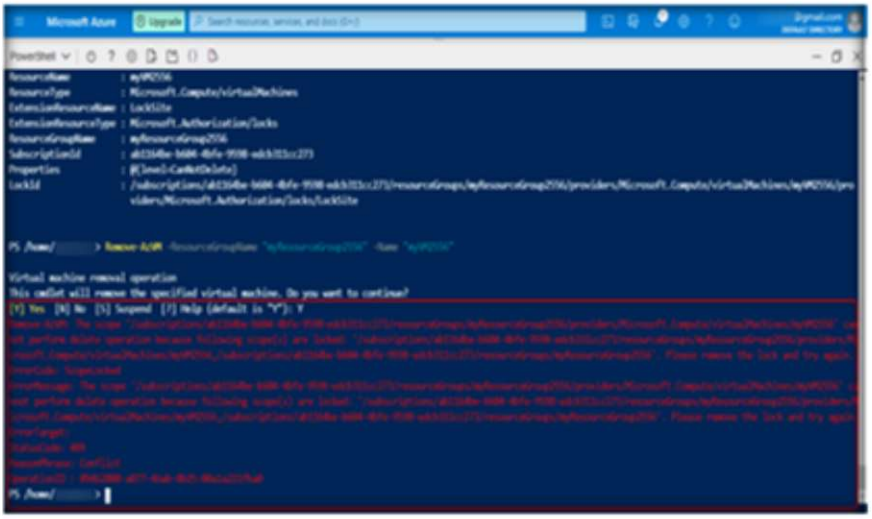

FIGURE 4.9.21: Error in VM Deletion

22. Thus, the resource group and resources cannot be deleted since they are locked, even by the administrator.

TASK 3

Unlock the Resource Group and Delete the Resource

23. Now, you can delete the lock on the resource group to remove the lock on all the resources in that resource group. Run the following commands to delete the lock.

```
$rgName = "myResourceGroup2556"

Get-AzResourceLock | Where-Object
ResourceGroupName -eq $rgName | Remove-
AzResourceLock -Force
```

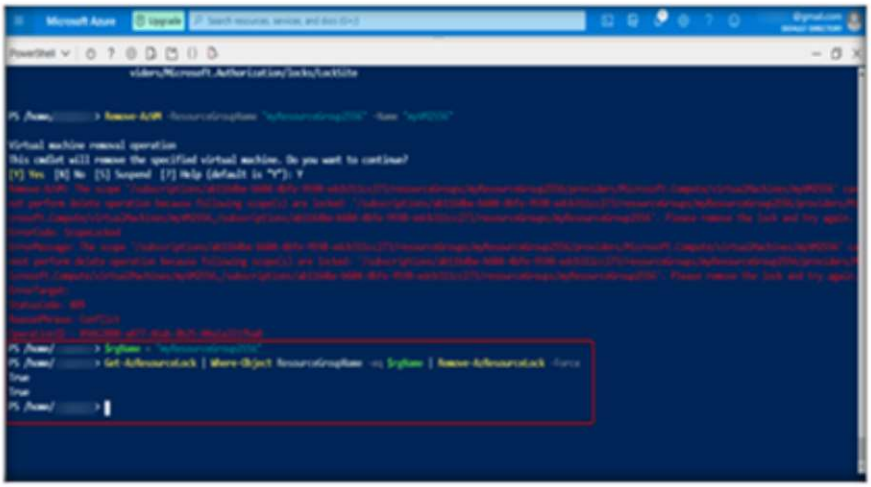

FIGURE 4.9.22: Deleting Lock of Resource Group

24. After removing the lock on the resource group, you can remove the VM, which was earlier not removed, since it was locked. Run the following command to delete the VM.

```
Remove-AzVM -ResourceGroupName
"myResourceGroup2556" -Name "myVM2556"
```

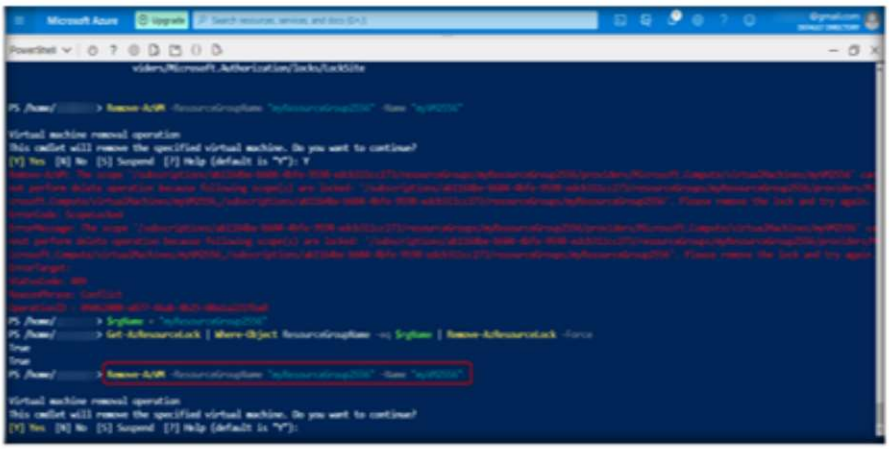

FIGURE 4.9.23: Removing the VM

25. Type **Y** and press **Enter** to confirm the removal of the virtual machine. You will see that the VM gets successfully removed.

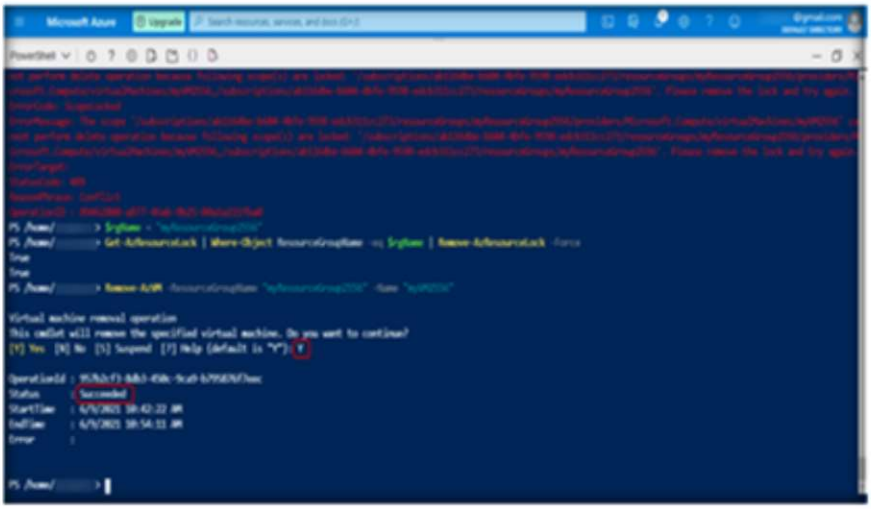

FIGURE 4.9.24: Successful Removal of the VM

26. Thus, a cloud security engineer can prevent the accidental deletion of critical resources in Azure by locking the resource and resource group.

Caution: Ensure you **delete, shut down, or terminate** all resources created and used in this lab to prevent their billing.

27. From the Azure portal, navigate to Resource groups and click on the name of the resource group (myResourceGroup2556). In the Overview window for the resource group, click on Delete resource group at the top.

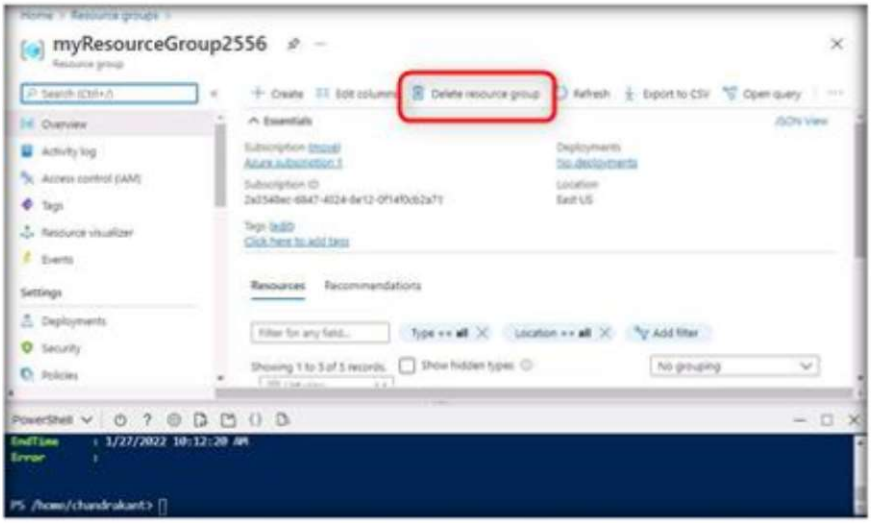

FIGURE 4.9.25: Deleting Resource Group

Lab Analysis

Analyze and document the results of this lab exercise. Provide your opinion on your target's security posture and exposure through free public information.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS ABOUT THIS LAB.