Course: Cloud and Network Security
Name: Neville Ngothe Iregi
Student No.: CS-CNS10-25054
Date: Tuesday, 14th October 2025

Week 7 Assignment 1: Role-Based Access Control

Introduction

Role-Based Access Control (RBAC) is a system that restricts system access to authorized users by assigning permissions to roles rather than to individual users. Users are then assigned to these roles, which are typically based on their job titles and responsibilities, streamlining the management of access rights and enhancing security. For example, a security analyst can configure a firewall but can't view customer data, while a sales rep can see customer accounts but can't touch firewall settings.

Lab scenario

I was asked to create a proof of concept showing how Azure users and groups are created. Also, how role-based access control is used to assign roles to groups. Specifically, I needed to:

- Create a Senior Admins group containing the user account of Joseph Price as its member.
- Create a Junior Admins group containing the user account of Isabel Garcia as its member.
- Create a Service Desk group containing the user account of Dylan Williams as its member.
- Assign the Virtual Machine Contributor role to the Service Desk group.

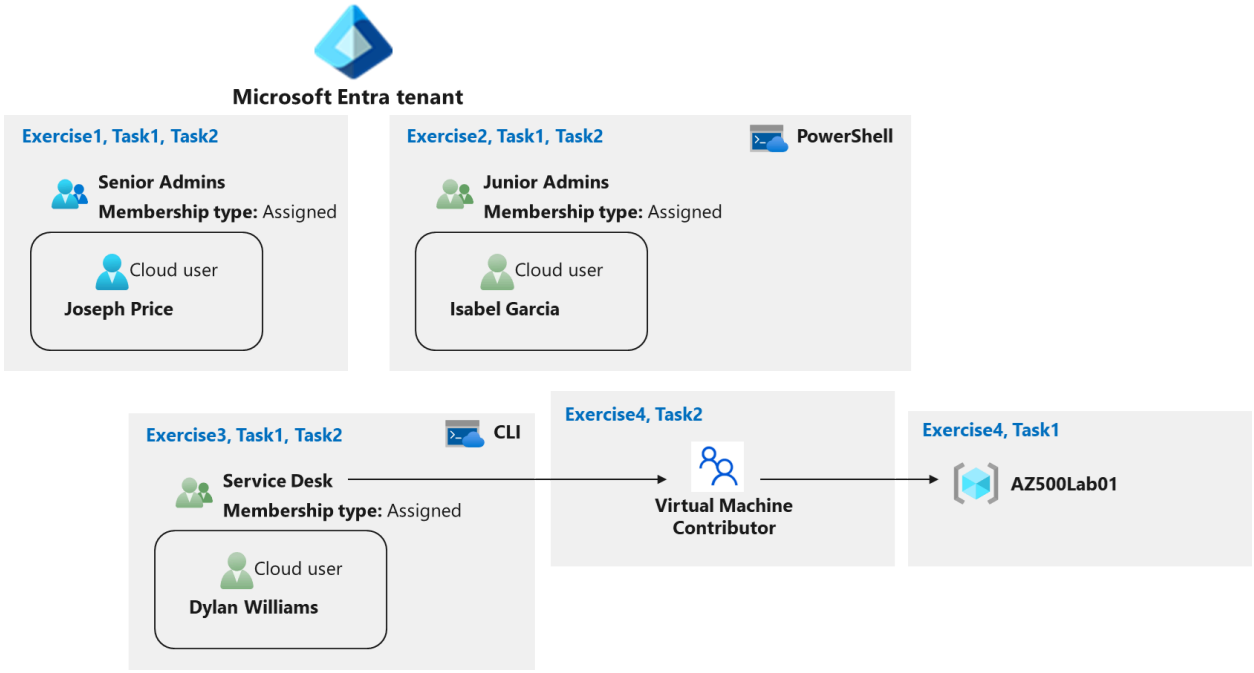
Lab objectives

In this lab, I completed the following exercises:

- Exercise 1: Create the Senior Admins group with the user account Joseph Price as its member (the Azure portal).
- Exercise 2: Create the Junior Admins group with the user account Isabel Garcia as its member (PowerShell).
- Exercise 3: Create the Service Desk group with the user Dylan Williams as its member (Azure CLI).

- Exercise 4: Assign the Virtual Machine Contributor role (an Azure Role-Based Access Control (RBAC) role that allows users to manage virtual machines, but not the networks or storage accounts they are connected to) to the Service Desk group.

Role-Based Access Control architecture diagram



Instructions

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

In this exercise, you will complete the following tasks:

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

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

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

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

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

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

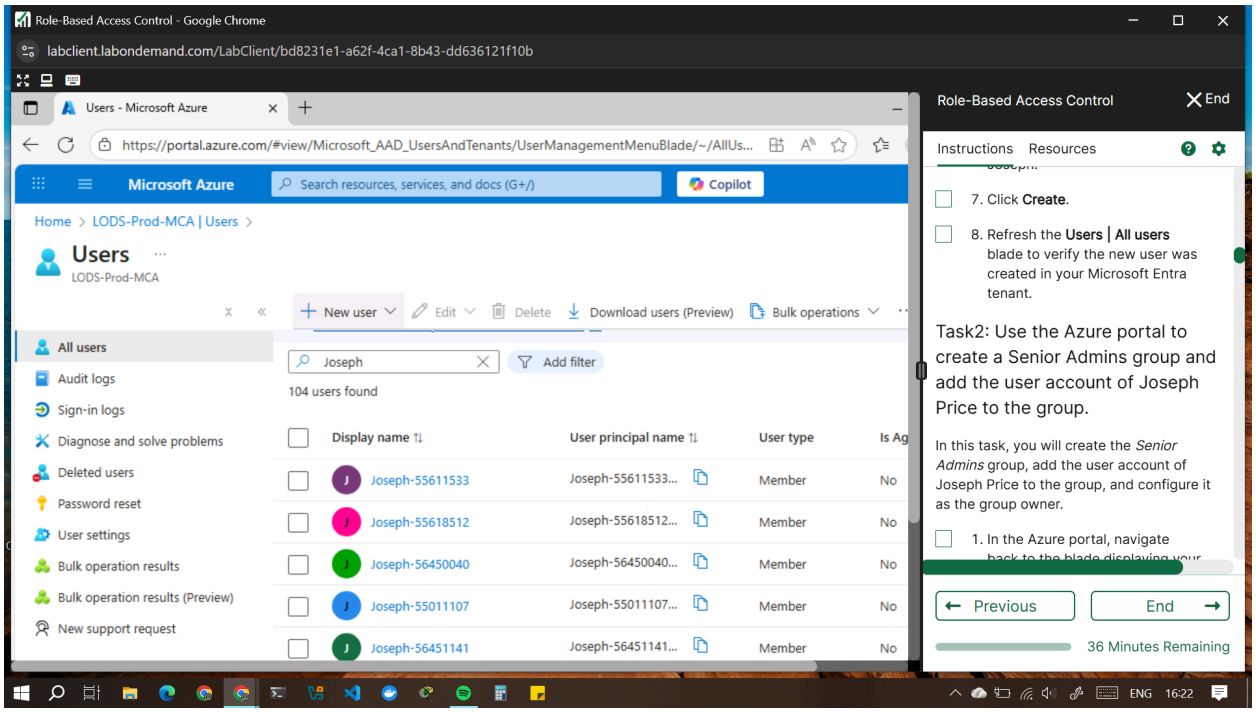
2. In the Search resources, services, and docs text box at the top of the Azure portal page, type Microsoft Entra ID and press the Enter key.
3. On the Overview blade of the Microsoft Entra ID tenant, in the Manage section, select Users, and then select + New user.

Steps 3-8 are for reference only, as this user has already been created for you. If you wish, you may review them but the New User button will be greyed out. This is expected in this Cloudslice lab and you may proceed to Task 2.

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

Setting	***Value***
User name	*Joseph*
Name	*Joseph Price*

5. Click on the copy icon next to the User name to copy the full user.
6. Ensure that the Auto-generate password is selected, select the Show password checkbox to identify the automatically generated password. You would need to provide this password, along with the user name to Joseph.
7. Click Create.
8. Refresh the Users | All users blade to verify the new user was created in your Microsoft Entra tenant.



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

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

1. In the Azure portal, navigate back to the blade displaying your Microsoft Entra ID tenant.
2. In the Manage section, click Groups, and then select + New group.
3. On the New Group blade, specify the following settings (leave others with their default values):

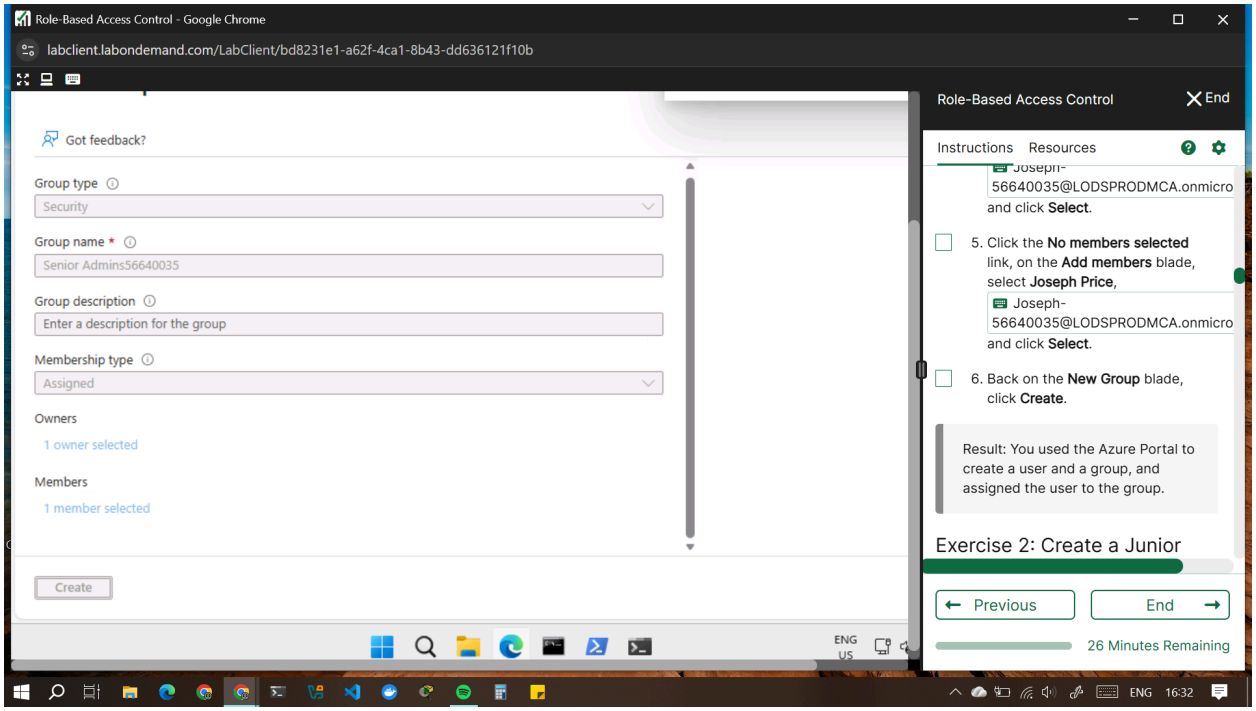
SettingValue

Group type Security

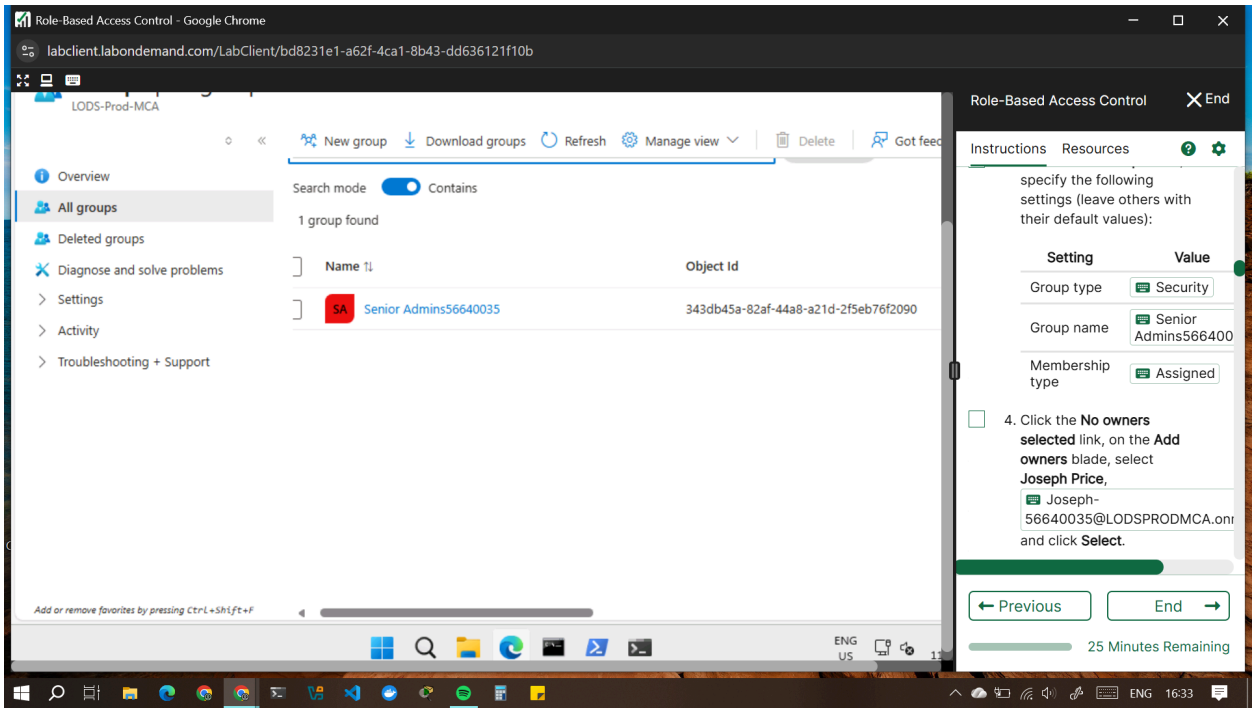
Group name Senior Admins56640035

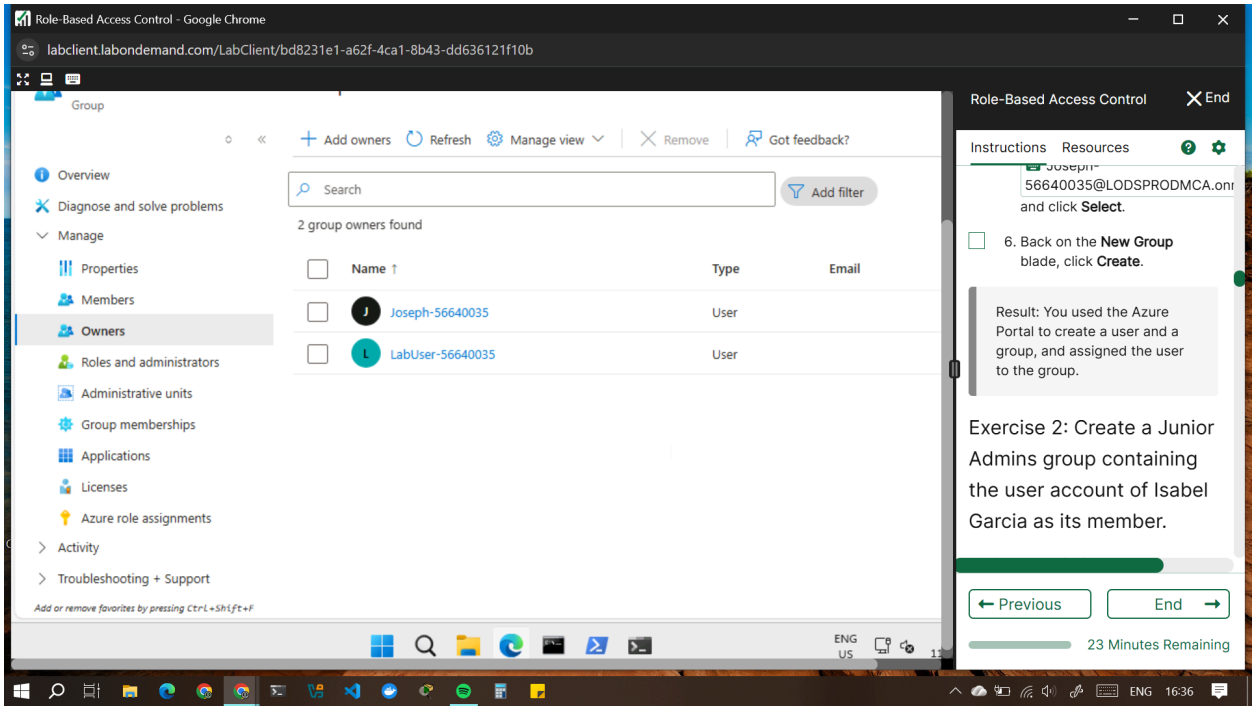
Membership type Assigned

4. Click the No owners selected link, on the Add owners blade, select Joseph Price, Joseph-56640035@LODSPRODMCA.onmicrosoft.com and click Select.
5. Click the No members selected link, on the Add members blade, select Joseph Price, Joseph-56640035@LODSPRODMCA.onmicrosoft.com and click Select.
6. Back on the New Group blade, click Create.



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





It can be seen that the user(Joseph) and group(Senior Admins56640035) has been created.

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

In this exercise, you will complete the following tasks:

- Task 1: Use PowerShell to create a user account for Isabel Garcia.
- Task 2: Use PowerShell to create the Junior Admins group and add the user account of Isabel Garcia to the group.

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

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

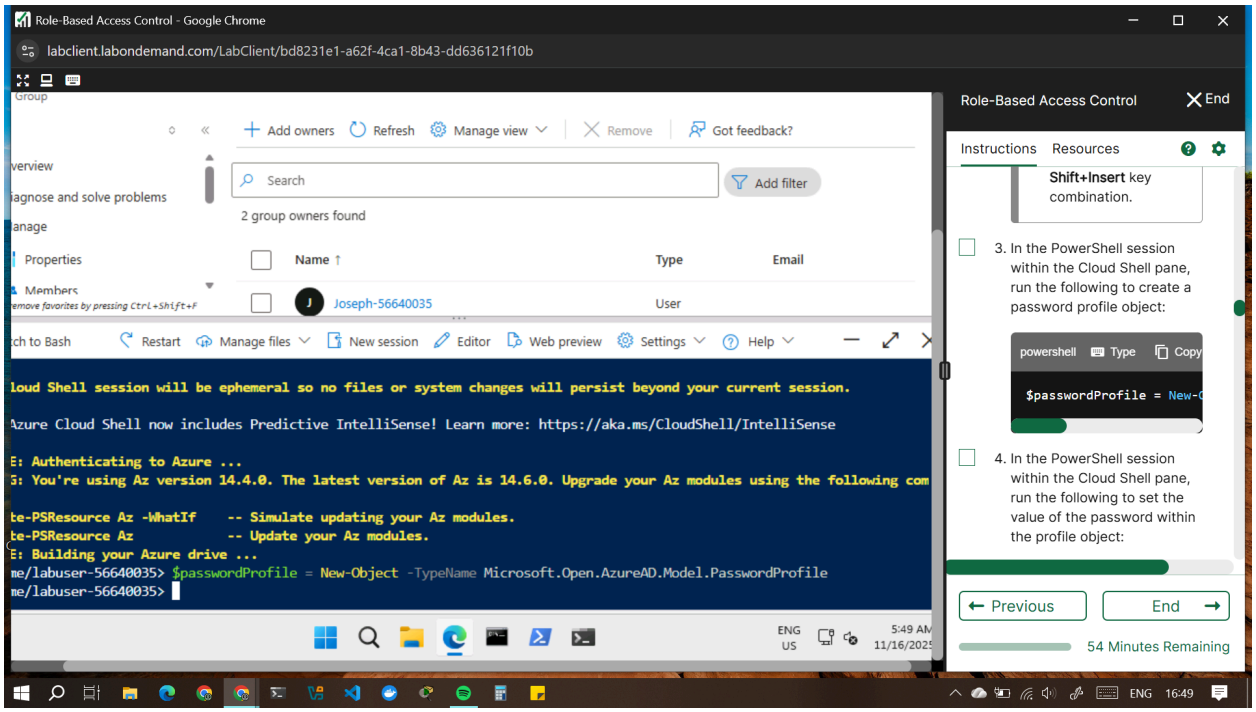
1. Open the Cloud Shell by clicking the first icon in the top right of the Azure Portal. If prompted, select PowerShell and Create storage.
2. Ensure PowerShell is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

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

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

`powershell`

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



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

powershell

$passwordProfile.Password = "Pa55w.rd1234"

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

powershell

Connect-AzureAD

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

powershell

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

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

Step 7 is for reference only, and can be skipped as this user has already been created for you. If you wish, you may run the command but you will receive the error New-AzureADUser: Error occurred while executing NewUser. This is expected in this Cloudslice lab and you may proceed to the next Step.

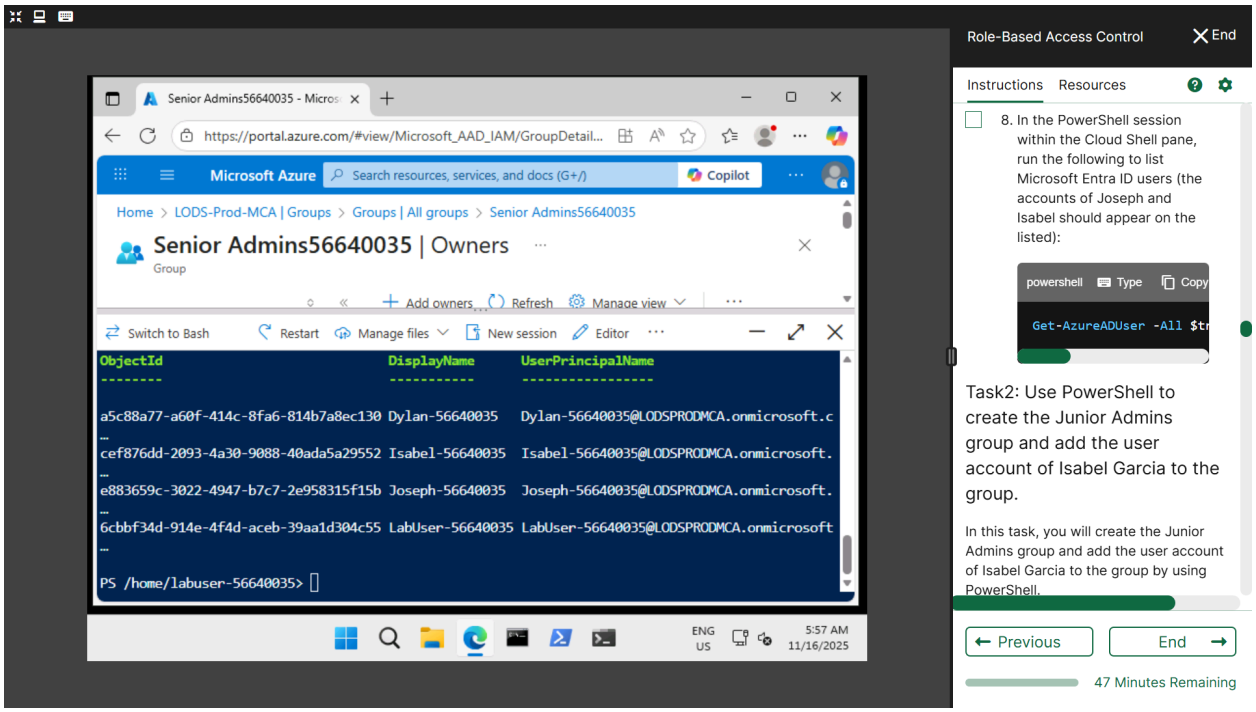
powershell

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

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

powershell

Get-AzureADUser -All $true | Where-Object {$_.UserPrincipalName -like "*56640035@LOD*"}



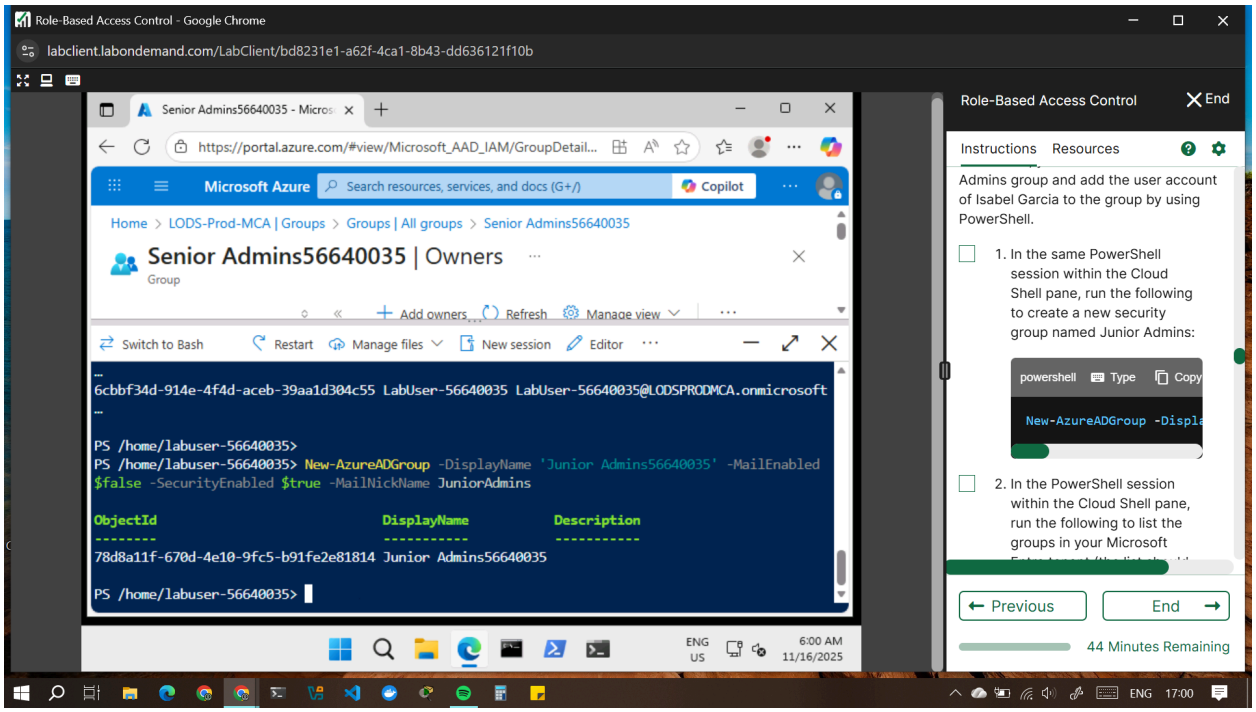
Task2: Use PowerShell to create the Junior Admins group and add the user account of Isabel Garcia to the group.

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

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

powershell

New-AzureADGroup -DisplayName 'Junior Admins56640035' -MailEnabled $false -SecurityEnabled $true -MailNickName JuniorAdmins



2. In the PowerShell session within the Cloud Shell pane, run the following to list the groups in your Microsoft Entra tenant (the list should include the Senior Admins56640035 and Junior Admins56640035 groups):

powershell

Get-AzureADGroup

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

powershell

$user = Get-AzureADUser -Filter "UserPrincipalName eq 'Isabel-56640035@LODSPRODMCA.onmicrosoft.com'"

4. In the PowerShell session within the Cloud Shell pane, run the following to add the user account of Isabel to the Junior Admins56640035 group:

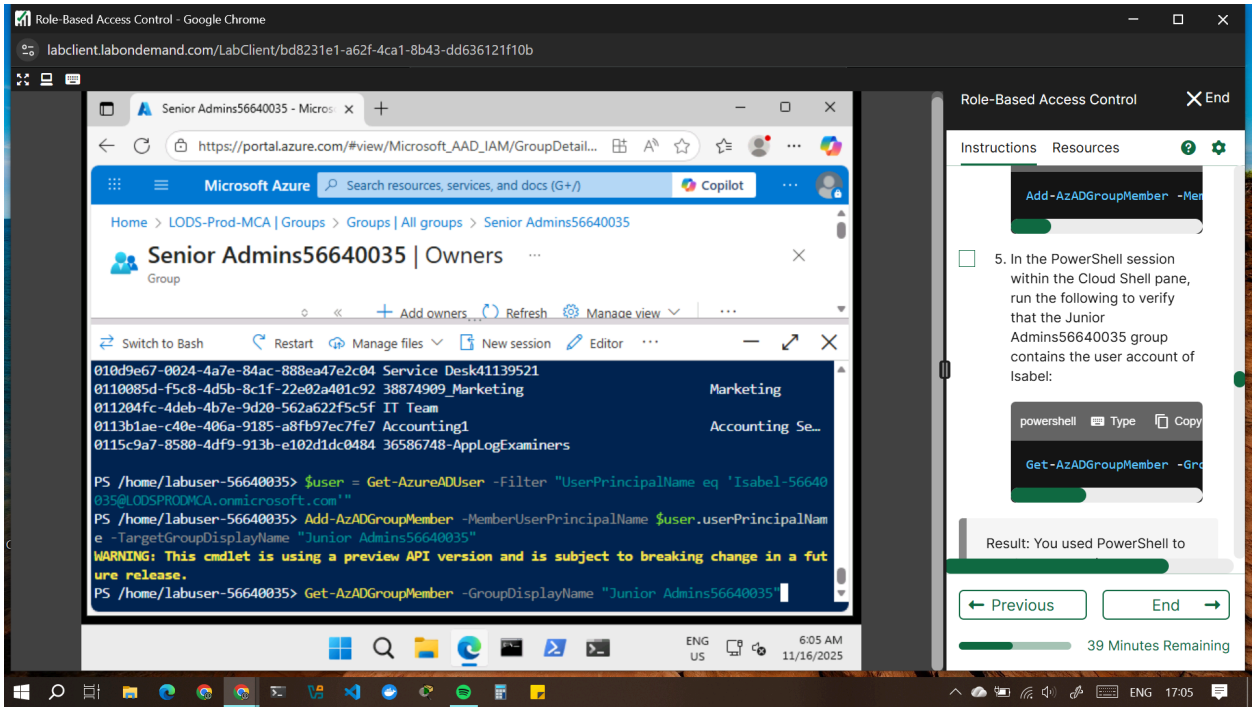
powershell

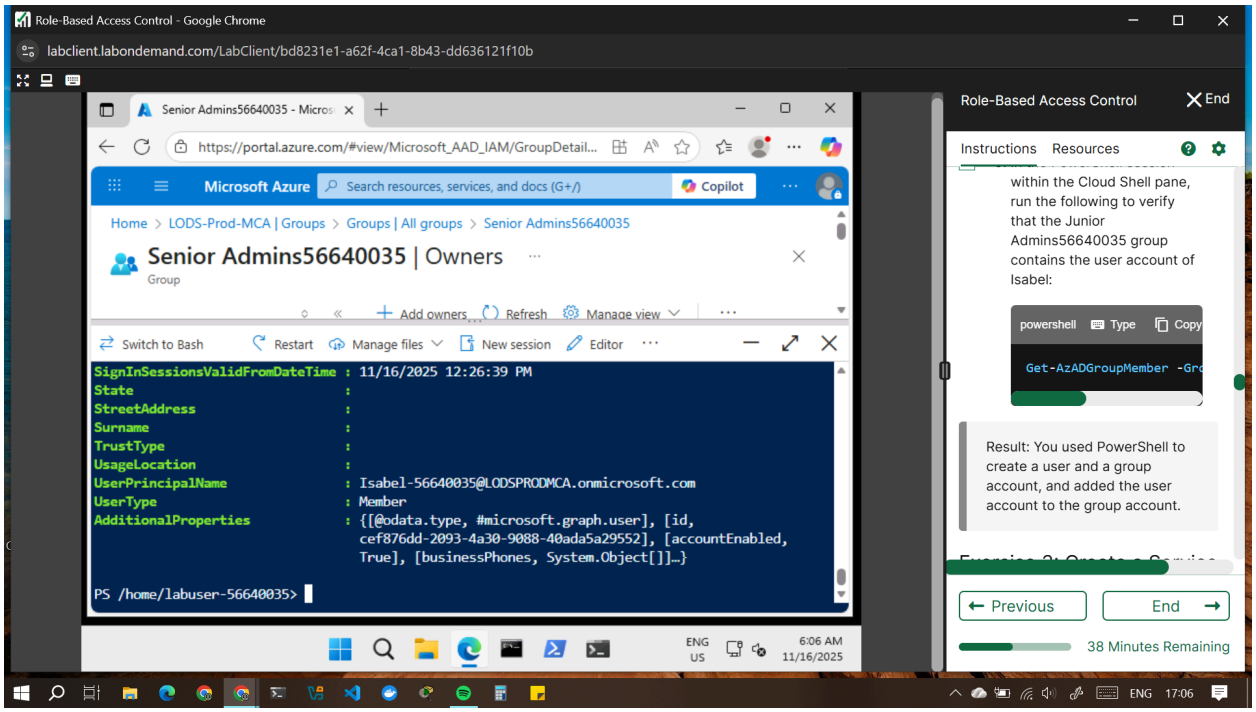
Add-AzADGroupMember -MemberUserPrincipalName $user.userPrincipalName -TargetGroupDisplayName "Junior Admins56640035"

5. In the PowerShell session within the Cloud Shell pane, run the following to verify that the Junior Admins56640035 group contains the user account of Isabel:

powershell

Get-AzADGroupMember -GroupDisplayName "Junior Admins56640035"





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

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

In this exercise, you will complete the following tasks:

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

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

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

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

1. In the drop-down menu in the upper-left corner of the Cloud Shell pane, select Bash, and, when prompted, click Confirm.
2. In the Bash session within the Cloud Shell pane, run the following to identify the name of your Microsoft Entra tenant:

cli

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

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

Step 3 is for reference only, as this user has already been created for you. If you wish, you may run the command but you will receive the error Insufficient privileges to complete the operation. This is expected in this Cloudslice lab and you may proceed to the next Step.

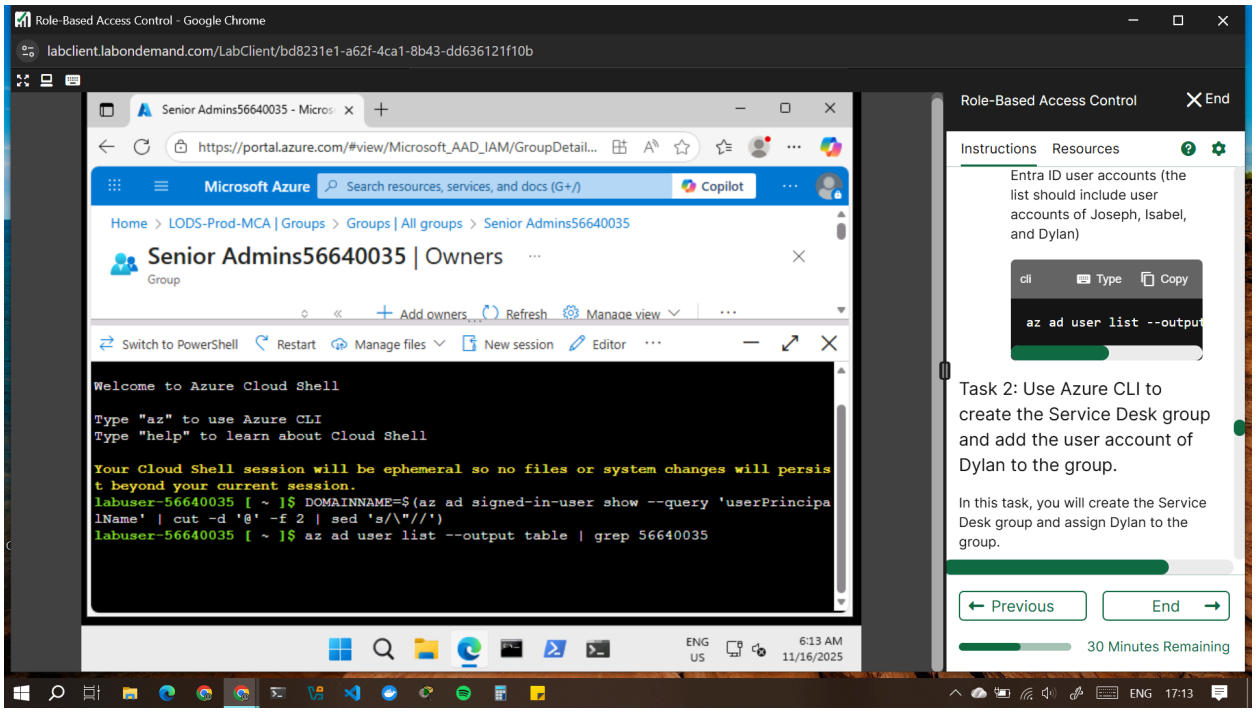
cli

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

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

cli

az ad user list --output table | grep 56640035



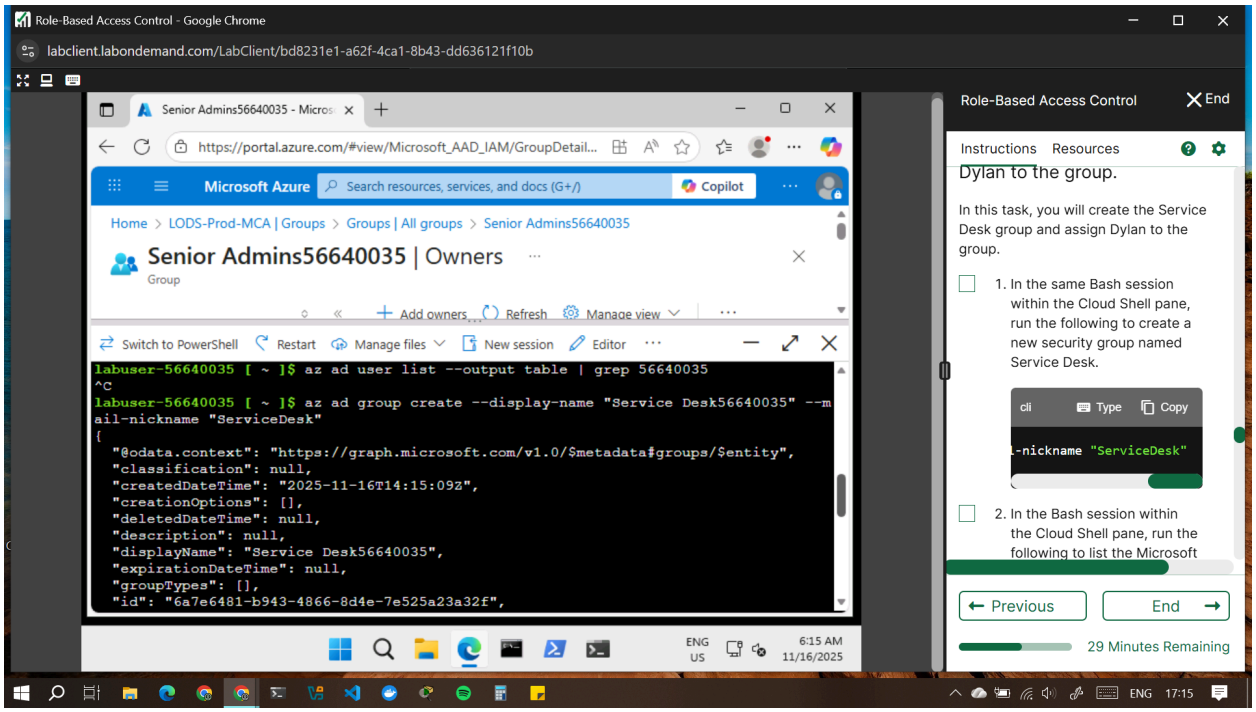
Task 2: Use Azure CLI to create the Service Desk group and add the user account of Dylan to the group.

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

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

cli

az ad group create --display-name "Service Desk56640035" --mail-nickname "ServiceDesk"



2. In the Bash session within the Cloud Shell pane, run the following to list the Microsoft Entra ID groups (the list should include Service Desk56640035, Senior Admins56640035, and Junior Admins56640035 groups):

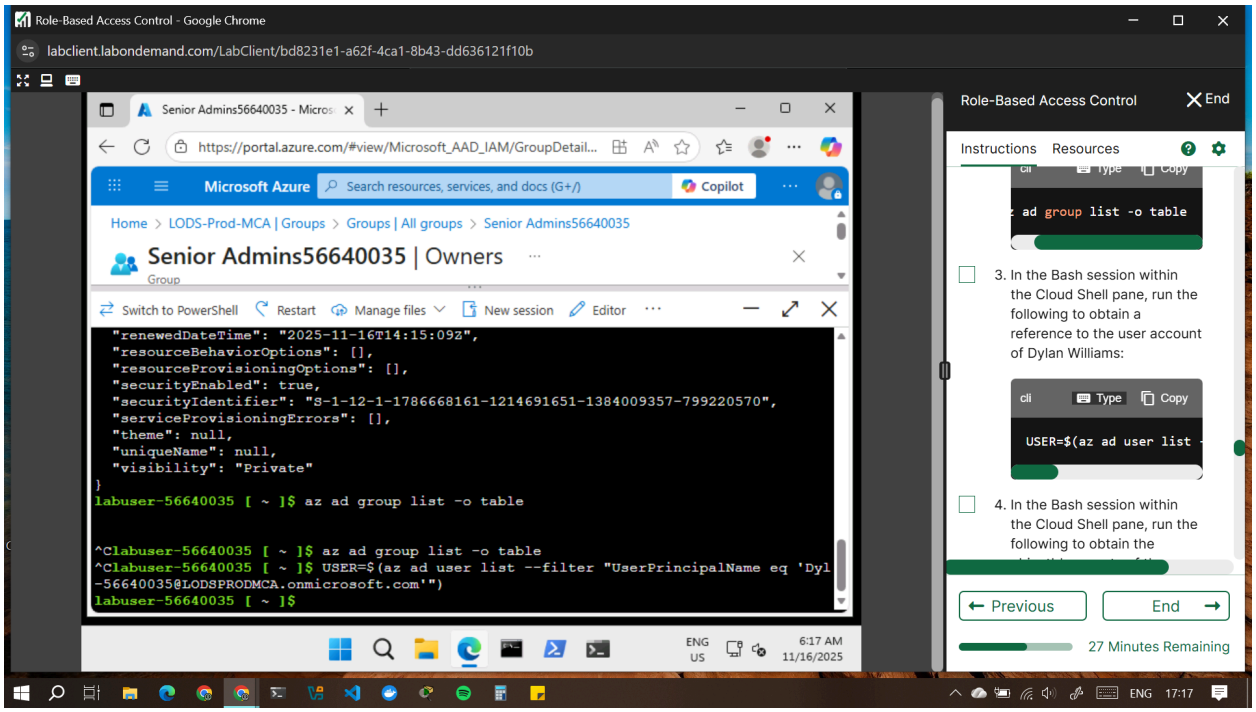
cli

az ad group list -o table

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

cli

USER=$(az ad user list --filter "UserPrincipalName eq 'Dylan-56640035@LODSPRODMCA.onmicrosoft.com'")



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

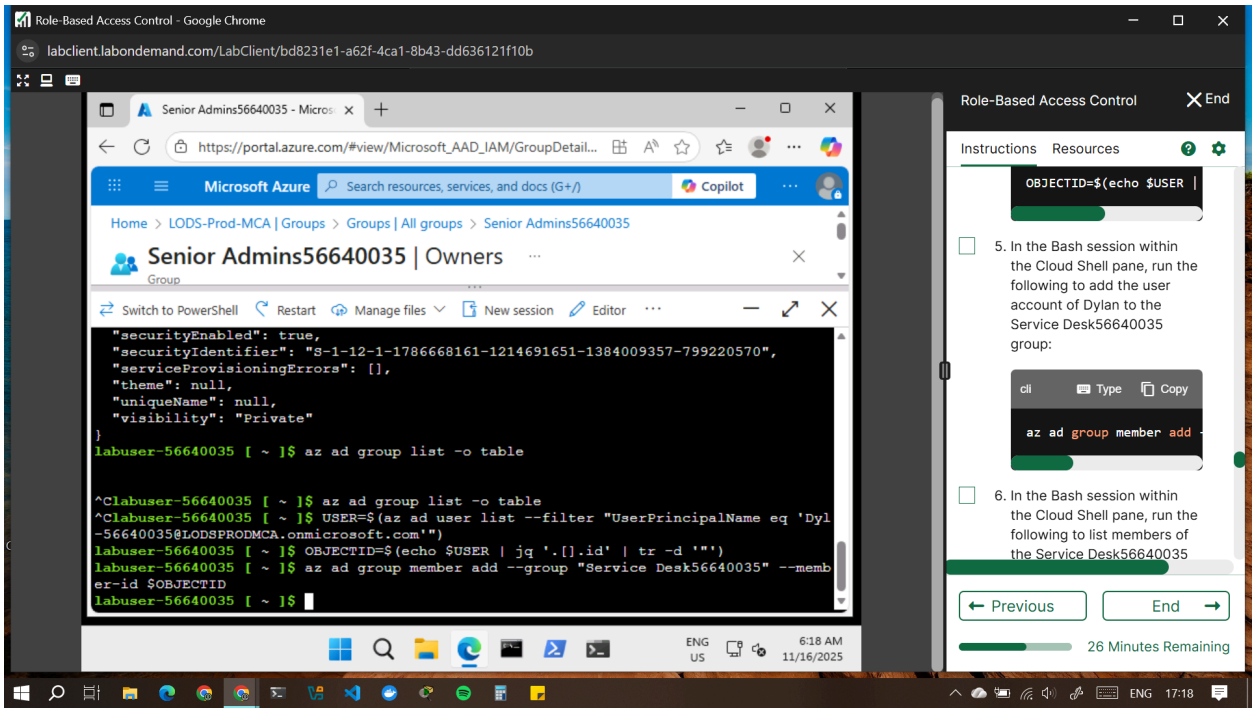
cli

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

5. In the Bash session within the Cloud Shell pane, run the following to add the user account of Dylan to the Service Desk56640035 group:

cli

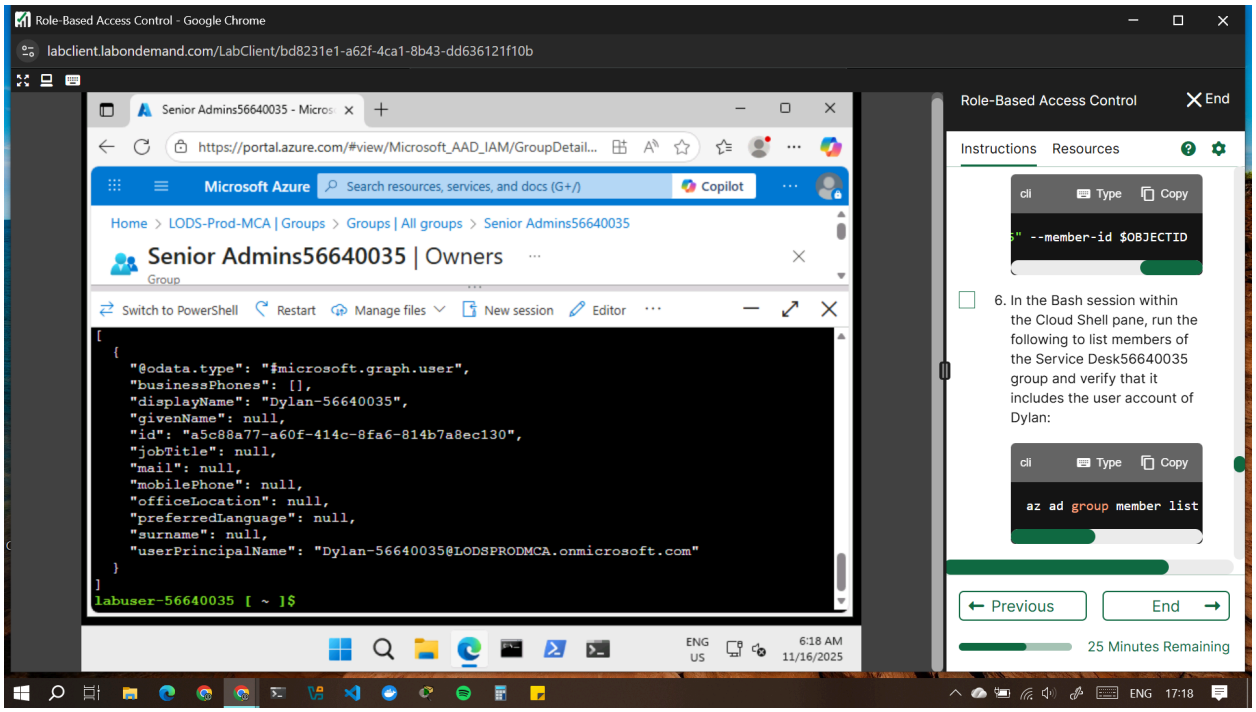
az ad group member add --group "Service Desk56640035" --member-id $OBJECTID



6. In the Bash session within the Cloud Shell pane, run the following to list members of the Service Desk56640035 group and verify that it includes the user account of Dylan:

cli

az ad group member list --group "Service Desk56640035"



7. Close the Cloud Shell pane.

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

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

In this exercise, you will complete the following tasks:

Task 1: Create a resource group.

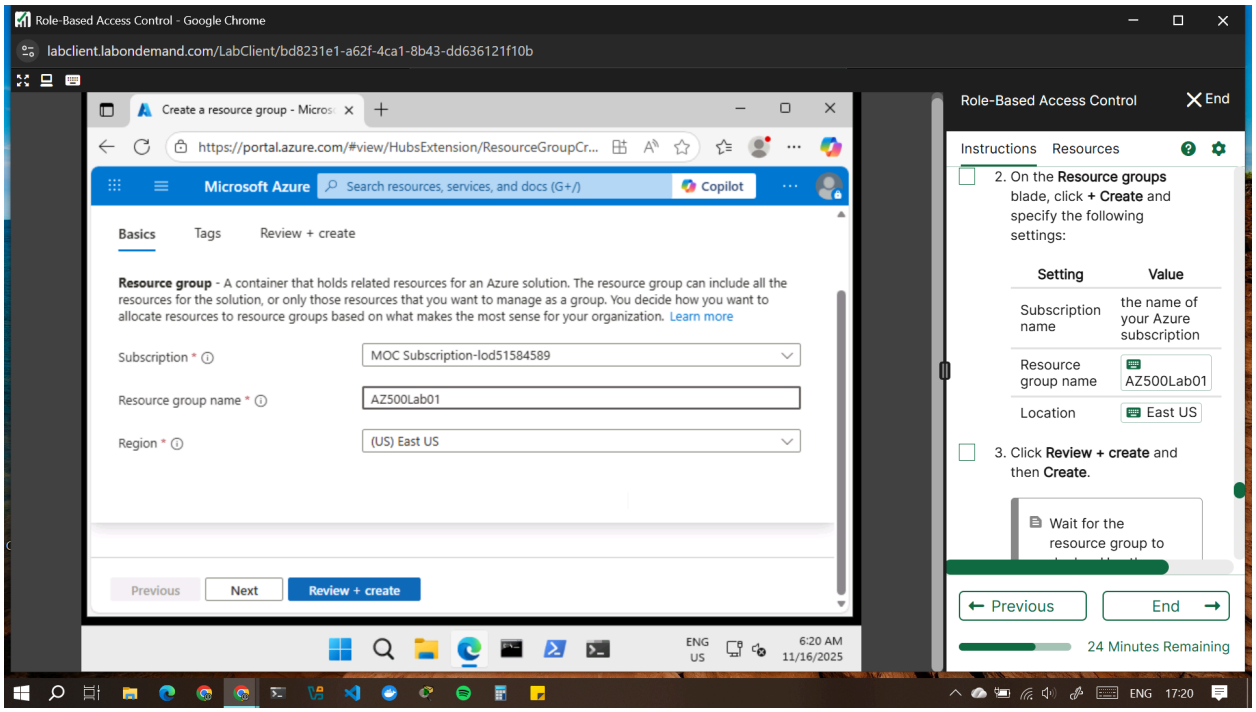
Task 2: Assign the Service Desk Virtual Machine Contributor permissions to the resource group.

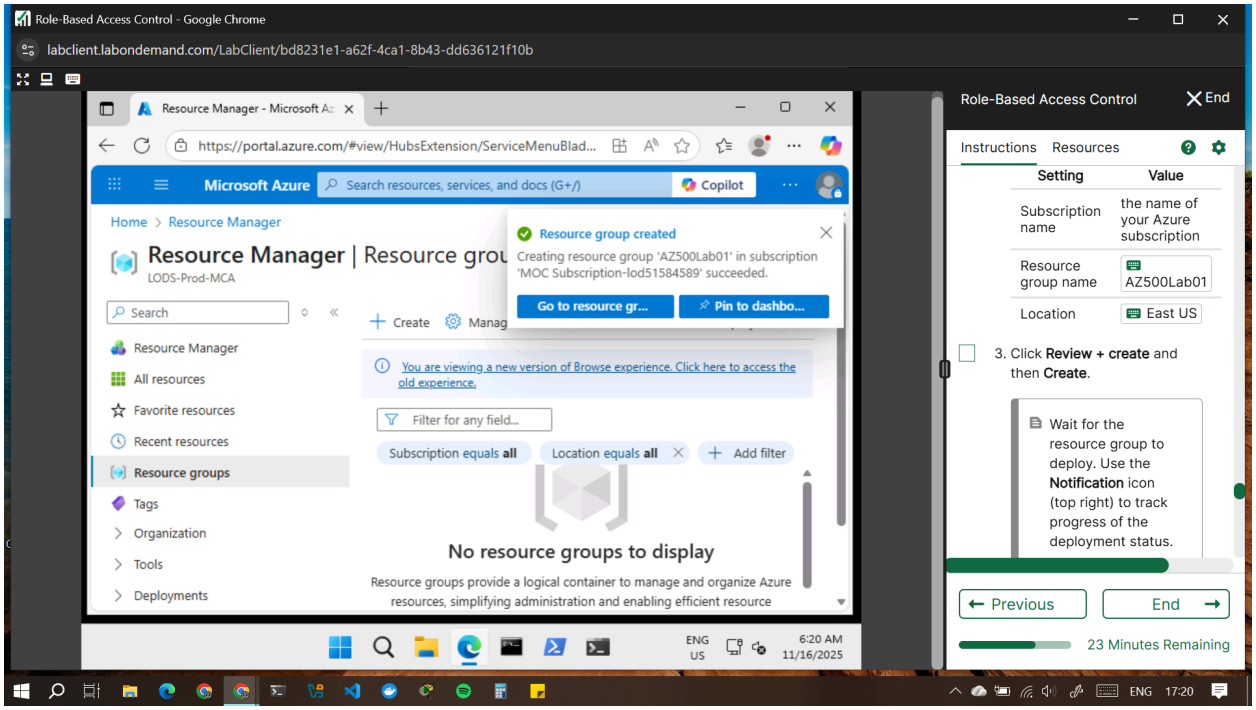
Task 1: Create a resource group

1. In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type Resource groups and press the Enter key.
2. On the Resource groups blade, click + Create and specify the following settings:

Setting		Value
Subscription name	-	the name of your Azure subscription
Resource group name	-	AZ500Lab01
Location	-	East US

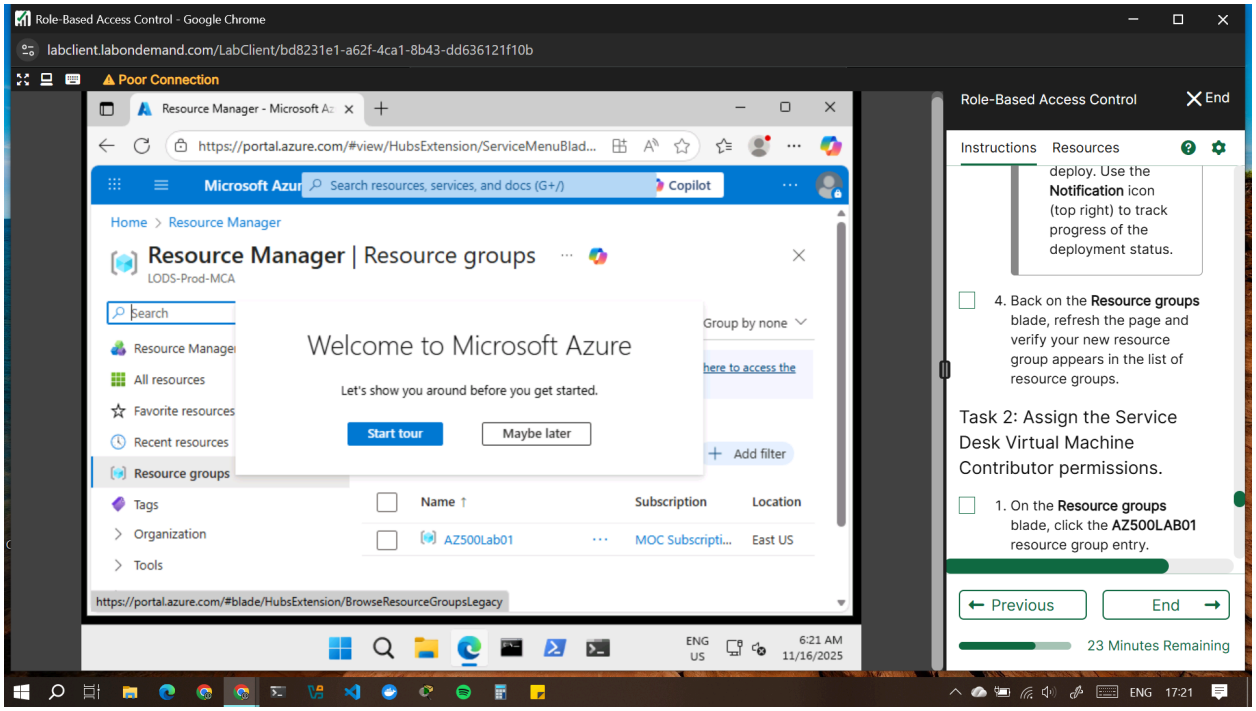
Click Review + create and then Create.





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

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



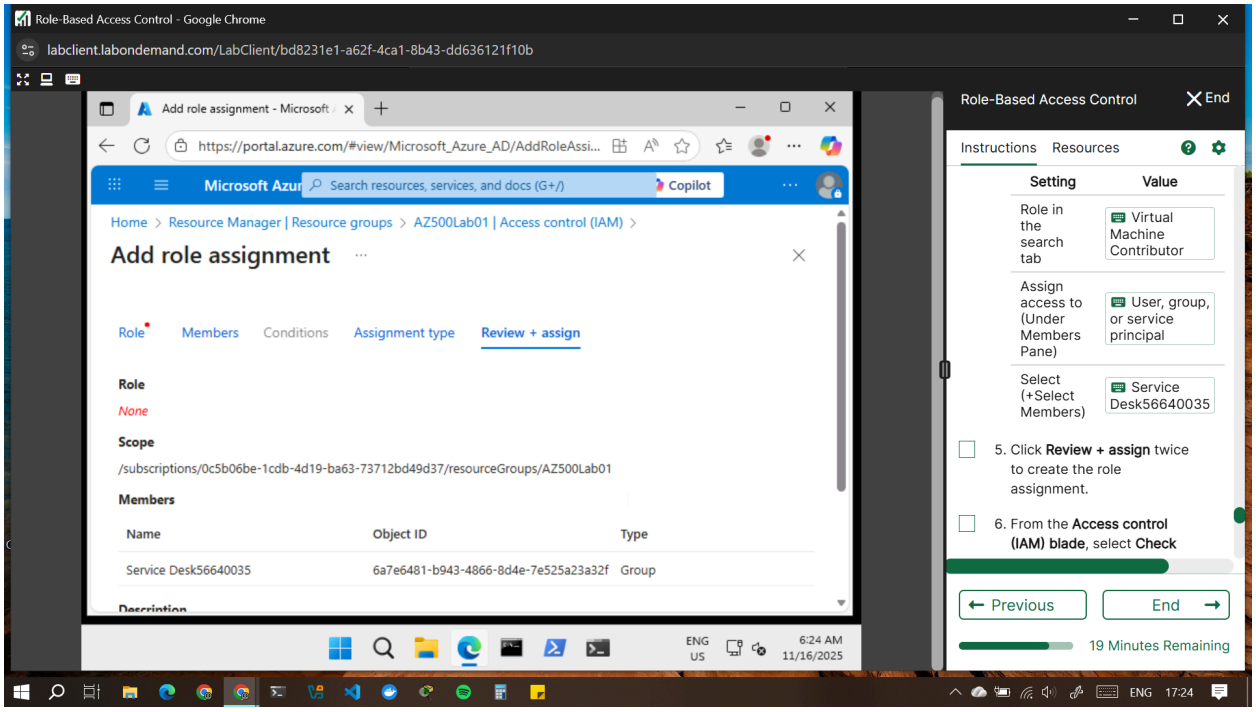
Task 2: Assign the Service Desk Virtual Machine Contributor permissions.

1. On the Resource groups blade, click the AZ500LAB01 resource group entry.
2. On the AZ500Lab01 blade, click Access control (IAM) in the middle pane.
3. On the AZ500Lab01 | Access control (IAM) blade, click + Add and then, in the drop-down menu, click Add role assignment.
4. On the Add role assignment blade, specify the following settings and click Next after each step:

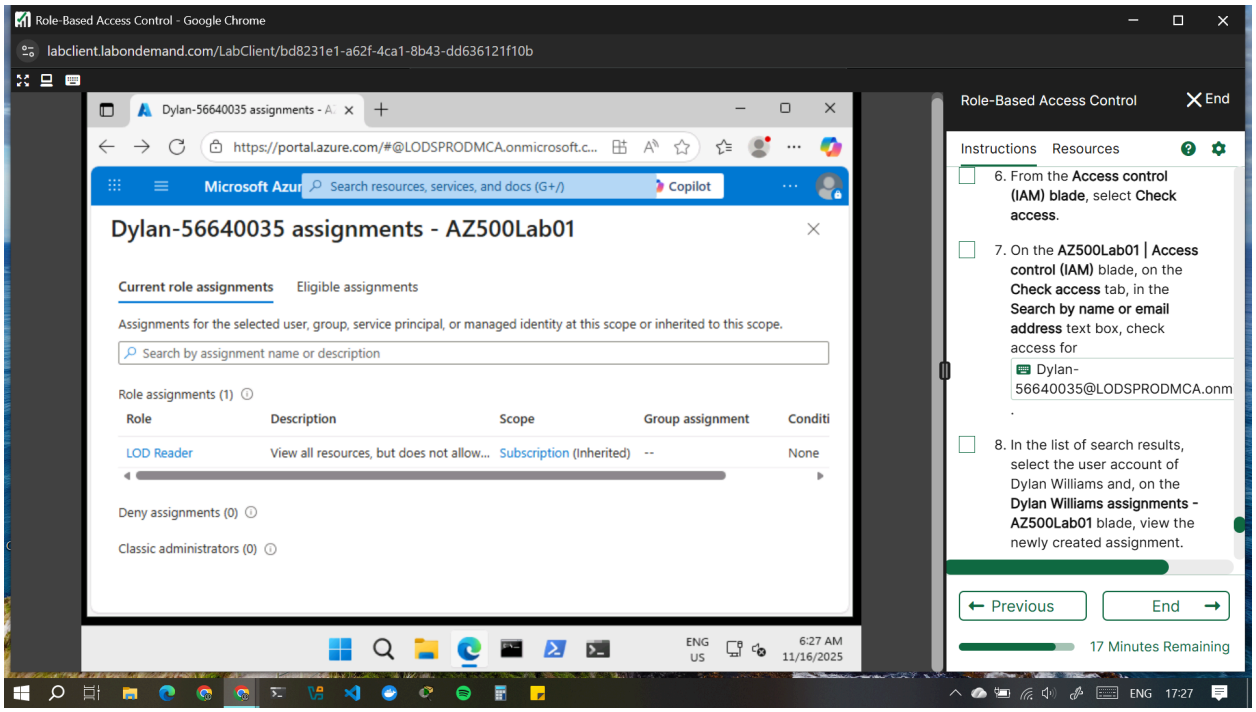
Setting	Value
Role in the search tab	Virtual Machine Contributor
Assign access to (Under Members Pane)	User, group, or service principal

Select (+Select Members) Service Desk56640035

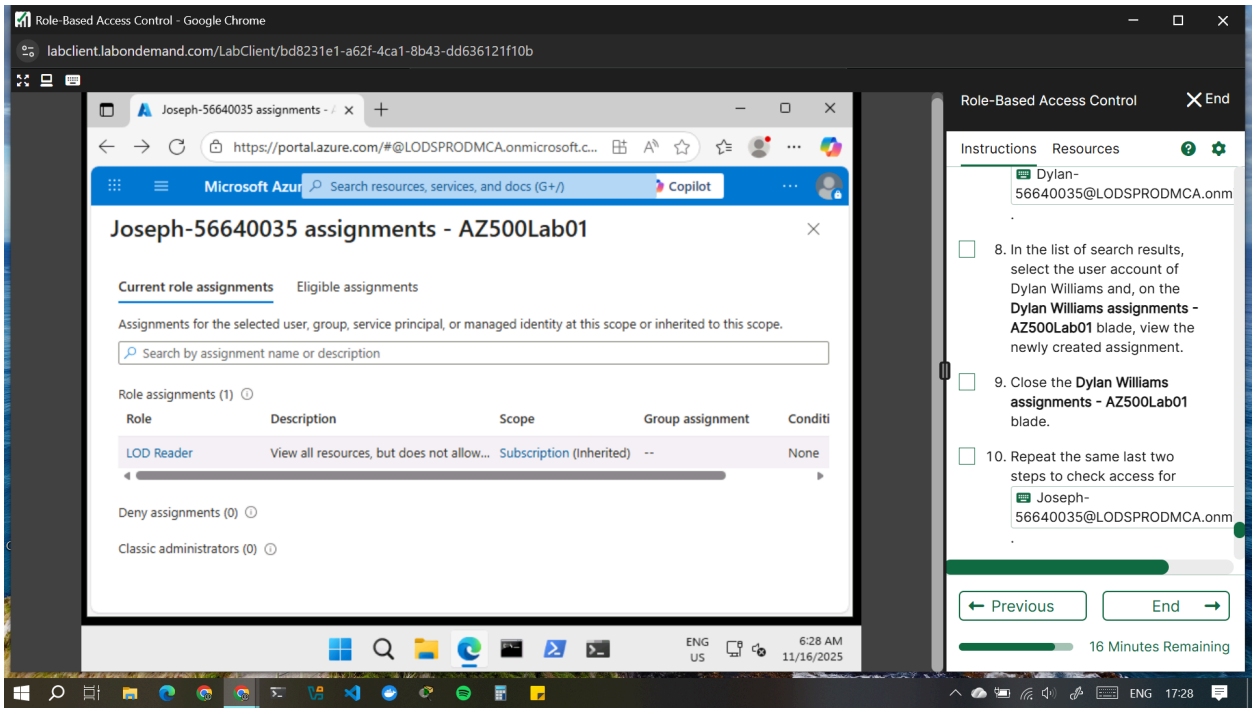
Click Review + assign twice to create the role assignment.



5. From the Access control (IAM) blade, select Check access.
6. On the AZ500Lab01 | Access control (IAM) blade, on the Check access tab, in the Search by name or email address text box, check access for Dylan-56640035@LODSPRODMCA.onmicrosoft.com.
7. In the list of search results, select the user account of Dylan Williams and, on the Dylan Williams assignments - AZ500Lab01 blade, view the newly created assignment.



8. Close the Dylan Williams assignments - AZ500Lab01 blade.
9. Repeat the same last two steps to check access for Joseph-56640035@LODSPRODMCA.onmicrosoft.com.



Result: You have assigned and checked RBAC permissions.

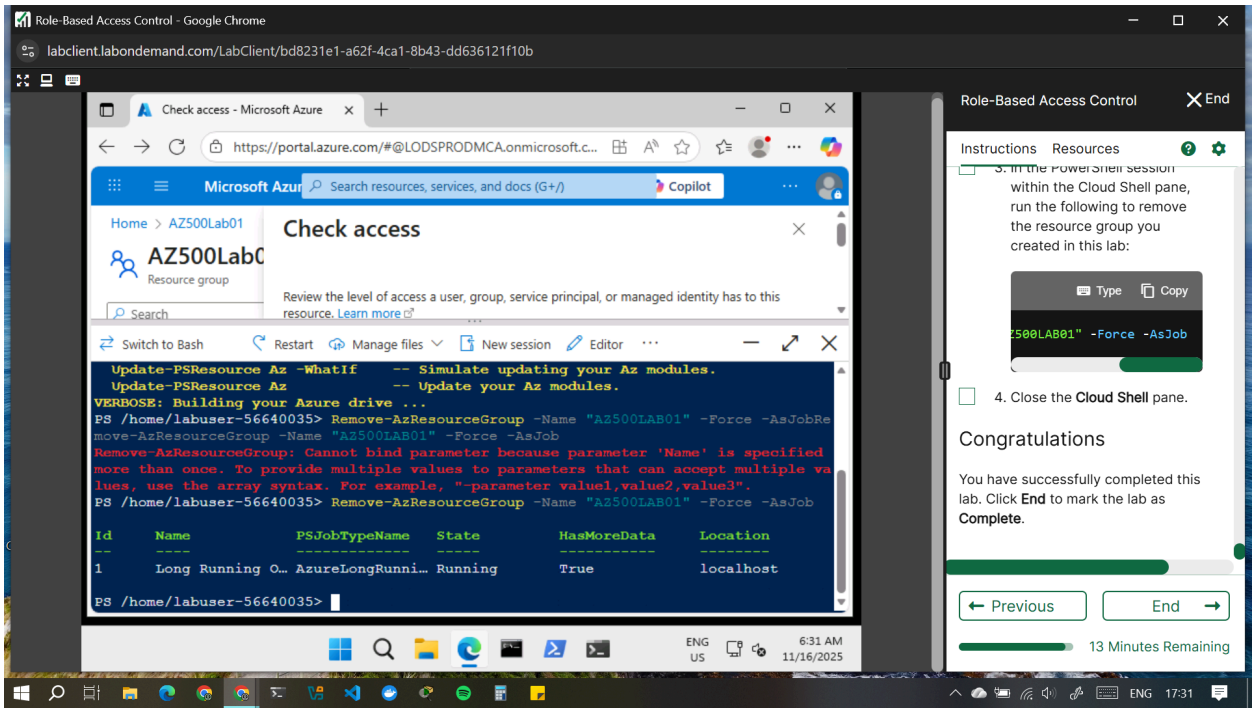
Clean up resources

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

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

TypeCopy

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



4. Close the Cloud Shell pane.

Conclusion

This lab provided practical, hands-on experience with managing identities, groups, and permissions in Microsoft Entra ID (Azure AD) using the Azure Portal, PowerShell, and the Azure CLI. By completing the exercises, I demonstrated how Role-Based Access Control (RBAC) streamlines permission management by assigning roles to groups rather than individual users, ensuring scalability, consistency, and improved security.

I successfully created three security groups—Senior Admins, Junior Admins, and Service Desk—each containing the appropriate user accounts created using different management tools. This reinforced the ability to work across multiple interfaces and scripting environments while achieving the same administrative outcomes. I also assigned the **Virtual Machine Contributor** role to the Service Desk group and verified the assignment, illustrating how RBAC controls access to Azure resources based on role definitions.

Overall, this lab enhanced my understanding of identity governance, group-based access control, and the practical implementation of RBAC in a cloud environment. These skills are essential for managing secure and efficient access in modern enterprise cloud infrastructures.