Lab #1: **Create Azure Account and Lab Environment**

Purpose:

- We'll set up a Microsoft Azure account/tenant subscription, which will be utilized frequently in the subsequent cybersecurity labs. We'll also configure the two "tester" VMs (Windows and Linux).
- A backup "Break-Glass" Global Admin account will be created. If we get locked out of our admin account, we can utilize our break-glass account to access our Azure portal.

Tasks:

1. **Create Azure Environment and Virtual Machines (VMs)**
   - Create Azure account (tenant) and subscription
   - Create "Tester" VMs (Windows 10 Pro, Linux Ubuntu)
   - Configure the Network Security Groups (NSG) for "Tester" VMs
2. **Set up Microsoft Remote Desktop on local PC**
3. **Create a backup "Break-Glass" Global Admin account in Entra ID**
   - Create the new admin user account
   - Log into it once and update the password

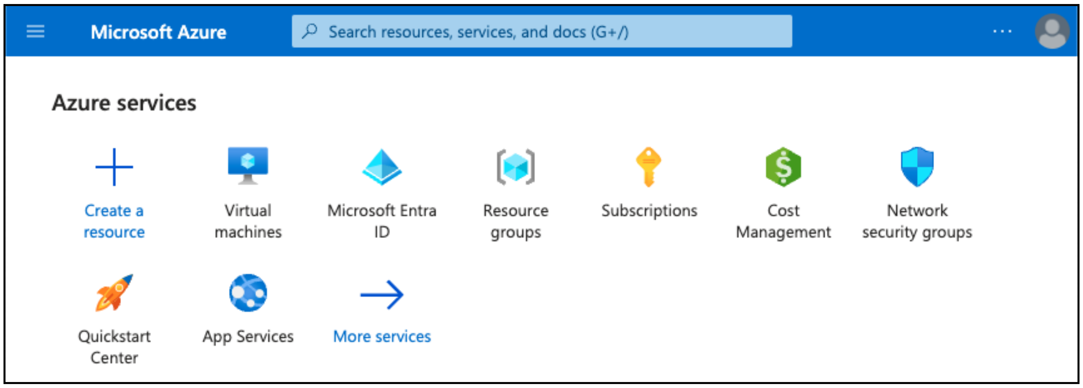
Task 1: **Create Azure Environment and Virtual Machines (VMs)**

Create Azure account (tenant) and subscription:

***Note**: You can make a free account, but it's recommended that you choose a "pay as you go" subscription as it provides more features and resources (may be required for the subsequent labs).*

1. Go to https://azure.microsoft.com/en-us/free/ to create an Azure account.
   a. Set up MFA.
2. Afterward, create the Azure subscription.

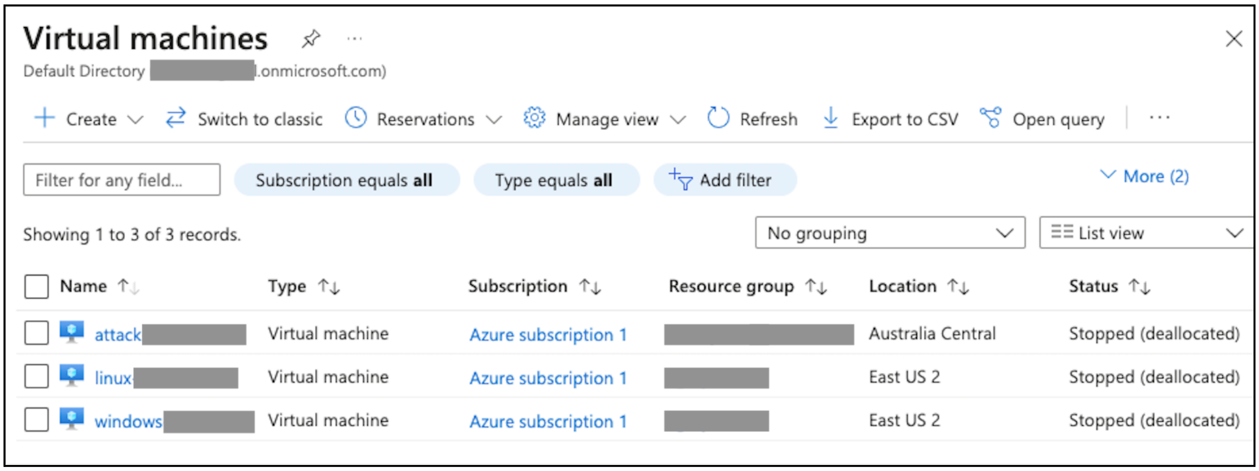
Create"Tester" VMs (Windows 10 Pro, Linux Ubuntu):



1. Steps for creating the Windows "Tester" VM:
   a. **Azure** account > **+Create a resource** > **Virtual Machines** > **Create**.
   b. Assign the subscription, resource group, VM name ("windows-vm"), region, image (Windows 10 Pro), admin credentials, and CPU usage.
   c. Create a new **Virtual Network**.
2. Steps for creating the Linux "Tester" VM:
   a. **Azure** account > **+Create a resource** > **Virtual Machines** > **Create**.
   b. Assign the subscription, resource group, VM name ("linux-vm"), region, image (Ubuntu Server), admin credentials, CPU usage.

c. Select the same **Virtual Network** as the Windows “Tester” VM.

***Note**: The screenshot below displays the 2 lab VMs we created, as well as a third “Attacker” VM that we’ll create soon in another lab.*



Default Directory [redacted].onmicrosoft.com)

Showing 1 to 3 of 3 records.

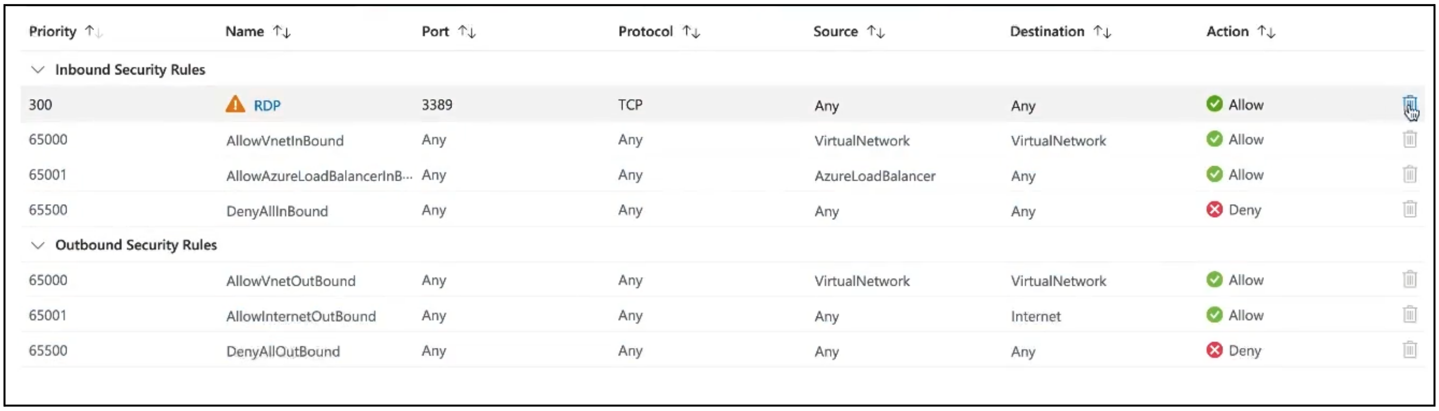
Name	Type	Subscription	Resource group	Location	Status
attack[redacted]	Virtual machine	Azure subscription 1	[redacted]	Australia Central	Stopped (deallocated)
linux[redacted]	Virtual machine	Azure subscription 1	[redacted]	East US 2	Stopped (deallocated)
windows[redacted]	Virtual machine	Azure subscription 1	[redacted]	East US 2	Stopped (deallocated)

Configure the Network Security Groups (NSG) for “Tester” VMs:

1. **Azure** portal > **Network Security Groups** (NSG) > there should be 2 options (Windows, Linux).
2. Select the Windows NSG:

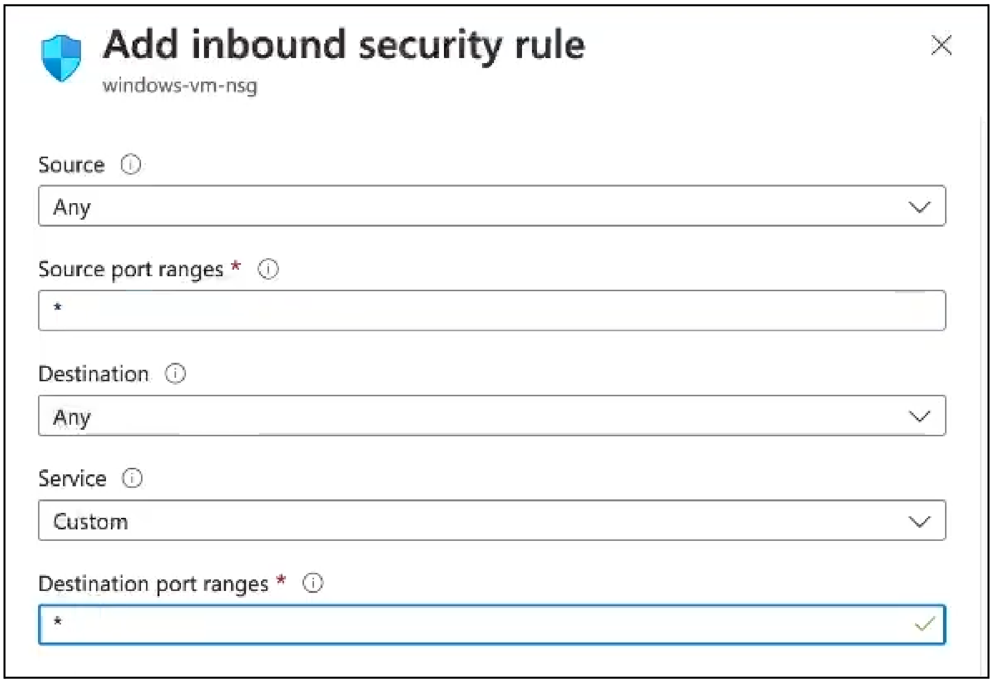
***Note**: We see both inbound and outbound security rules*

a. Delete the top-most **inbound security rule** (RDP).



Priority	Name	Port	Protocol	Source	Destination	Action
Inbound Security Rules						
300	⚠ RDP	3389	TCP	Any	Any	✓ Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow
65001	AllowAzureLoadBalancerInB...	Any	Any	AzureLoadBalancer	Any	✓ Allow
65500	DenyAllInBound	Any	Any	Any	Any	✖ Deny
Outbound Security Rules						
65000	AllowVnetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	✓ Allow
65001	AllowInternetOutBound	Any	Any	Any	Internet	✓ Allow
65500	DenyAllOutBound	Any	Any	Any	Any	✖ Deny

b. Create a new inbound security rule: **Settings** > **Inbound Security Rules**.



Add inbound security rule
windows-vm-nsg

Source: Any

Source port ranges *: *

Destination: Any

Service: Custom

Destination port ranges *: *

i. **Source**: Any
ii. **Source port ranges**: *
iii. **Destination**: Any
iv. **Destination on port ranges**: *
v. **Priority**: 100
vi. **Name**: (ex. “DANGER_AllowAnyInbound”)

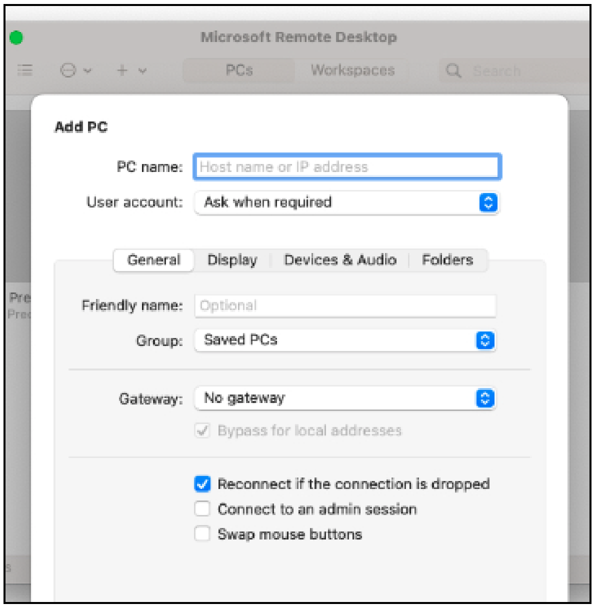
3. Now, select the Linux NSG:
   a. (perform the same steps as the Windows NSG)

***Note**: This is bad practice (it allows <u>all</u> inbound traffic!). We’re only performing this for testing purposes. This should encourage attackers to interact with the “Tester” VMs (we’ll analyze the security logs in future labs).*

Task 2: Set up Microsoft Remote Desktop on local PC

1. Open the **Microsoft Remote Desktop** app.

2. Add the “Tester” and “Attacker” VMs to the Microsoft Remote Desktop app:
   a. Locate the **public IPs** for each VM > select **Add PC**.



Add PC

PC name: Host name or IP address

User account: Ask when required

Friendly name: Optional

Group: Saved PCs

Gateway: No gateway

☑ Bypass for local addresses

☑ Reconnect if the connection is dropped

☐ Connect to an admin session

☐ Swap mouse buttons

Task 3: Create a backup “Break-Glass” Global Admin account

***Note**: In Entra ID we’ll create a backup account with Global Admin privileges (aka a “Break-Glass” account). This account allows us to still access our Azure tenant in case Microsoft locks our main admin account because of our security testing.*

Create the new admin user account:

1. **Azure** portal > **Entra ID** > **Users** >
   a. Assignments: (assign the **Global Administrator** role)
   b. Select **Review + Create**.
2. Copy the full username (email) of the new admin user.

Create the new admin user account:

1. Open a new incognito window > portal.azure.com > sign in using the new admin user’s credentials.
2. Update the password.

End:

- For future labs, we will want to keep these VMs on so we have logs and behaviors to ultimately analyze.