Lab #2: **Configure Microsoft SQL Server**

Purpose:

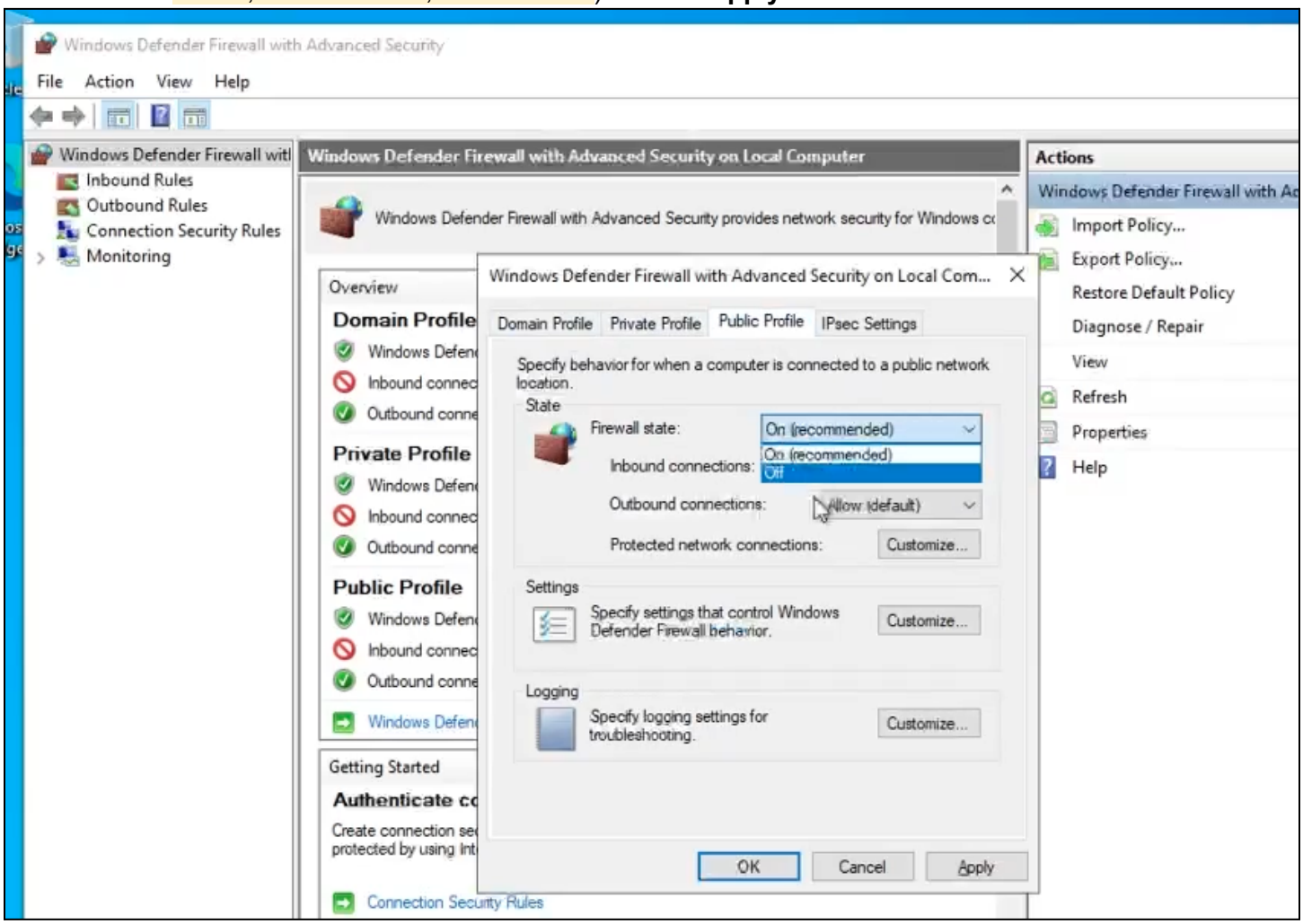
- We'll install the Microsoft SQL server inside the Windows "tester" VM. This server will eventually be used to test logging and monitoring.

Tasks:

1. **Disable the Windows Firewall in Windows "Tester" VM**
2. **Install SQL Server and create vulnerabilities**
3. **Test the Linux "Tester" VM (ping and SSH)**

Task 1: **Disable the Windows Firewall in Windows "Tester" VM**

1. Open windows-vm > open **Windows Defender Firewall** (**wf.msc**).
    a. **Windows Defender Firewall Properties** > update **Firewall State** to **OFF** (in 3 tabs: Domain Profile, Private Profile, Public Profile) > select **Apply** > **OK**.



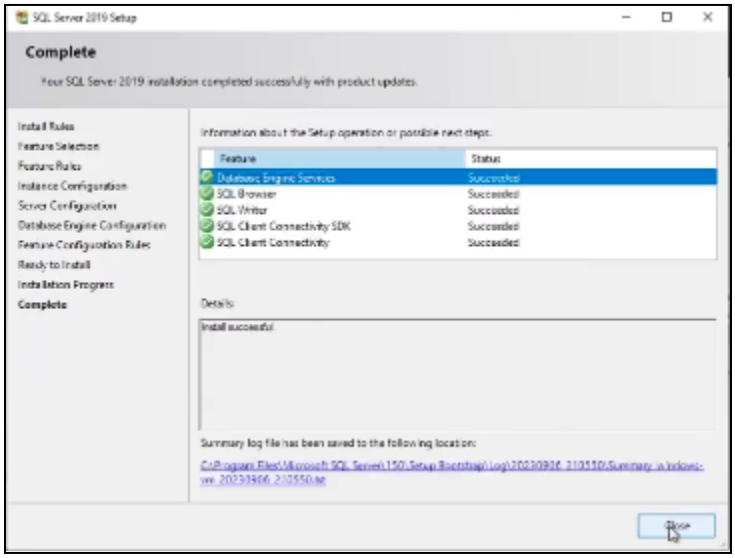
2. From your local PC, you should be able to ping the public IP of windows-vm.

Task 2: Install SQL Server and create vulnerabilities

1. In windows-vm, install **SQL Server Evaluation**:

   ***Note**: This will be another thing for attackers to attempt to attack.*

   a. In windows-vm, open **Edge** > go to Google.com > search for **Download SQL Server Evaluation**.
   b. Select the first option (should be **SQL Server 2019**) > select **Download the EXE** > Enter information, and select **Download Now** > select **EXE download** (64-bit edition).
   c. Open the downloaded file to finish the install > select **Download Media** > Update **Download Location** to **Desktop** > select **Download**.
   d. Once the download completes, select **Open Folder** (you'll see the Desktop folder and the new ISO file) > right-click the ISO file, and select **Mount** > double-click **setup** [to start the installation of SQL Server] > select **Installation**, and select **New SQL Server stand-alone…** > select **Next** (3x) > in **Feature Selection**, select the **Database Engine Services** checkbox and select **Next** > select **Next** (2x) > select **Mixed Mode** (for authentication mode) > enter password > select **Add Current User** (wait 10s), then select **Next** > (select **Finish** on the last page).
   e. The install process will take 10-20min to complete.



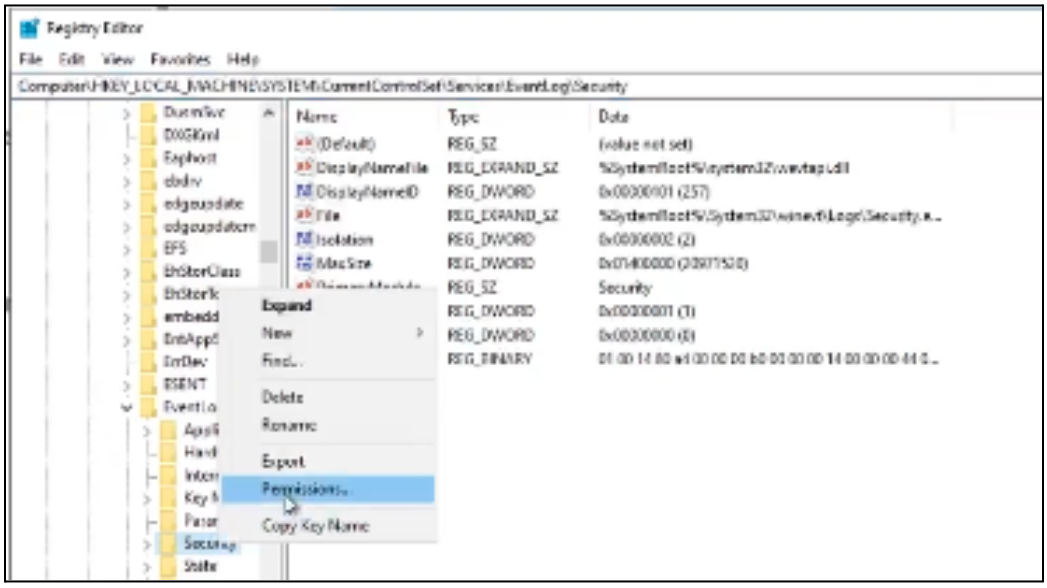
2. In windows-vm, also install **SQL Server Management Studio**:

   ***Note**: This app lets us log into SQL Server and visualize data. We'll soon see logon attempts from attackers in Windows Event Viewer.*

   a. Go back to **Edge** > go to Google.com, and search for **Download SSMS** > Under the **Download SSMS** section, select **Free Download…SSMS…19.1**.
   b. Open the downloaded file > select **Install** (it'll take 10-20min).
   c. Now, you'll need to restart your VM.

3. Enable logging for the SQL server (to receive logs for sign-in attempts):
   a. Follow the steps in this Microsoft article: Write SQL Server Audit events to the Security log
   b. Open the **Registry** (regedit.msc) > go to HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Security

c. > right-click it and select **Permissions** > (System > **Add**) > type **Network Service** (select **Check Name**), and select **OK** > select **Full Control** checkbox, **Apply**, and **OK**.

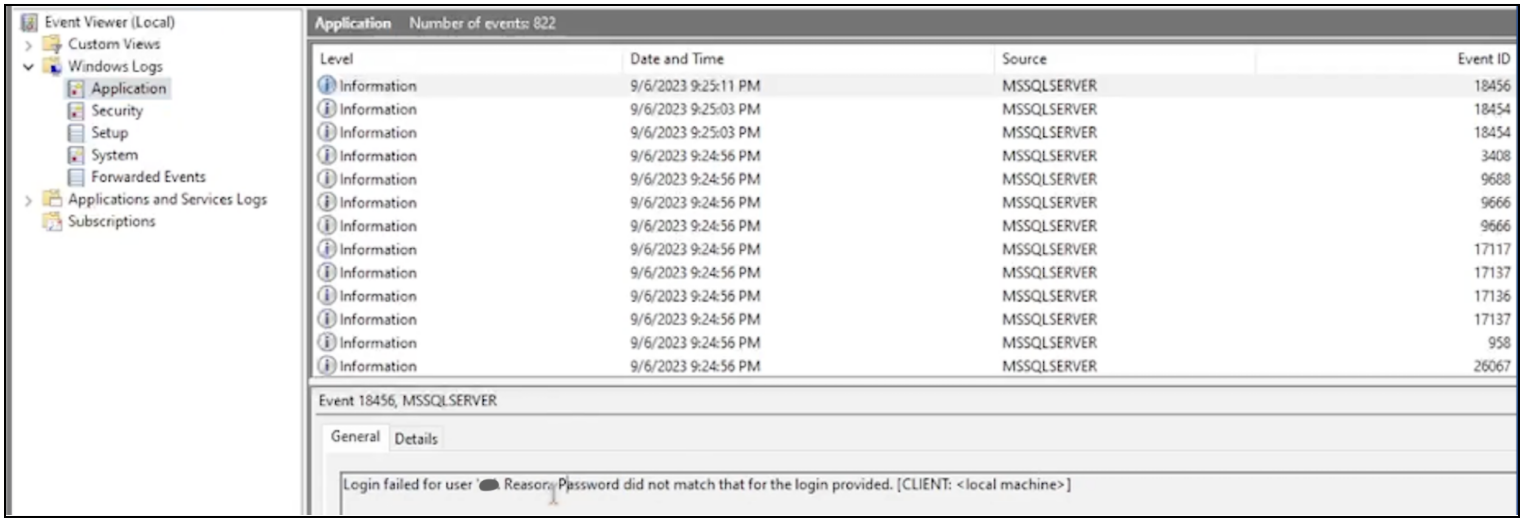


d. Copy the **Windows Command Prompt** prompt (from the above article).
   i. In Command Prompt (run as admin) run the prompt > done (close any open windows).

4. Open the **SSMS** app and enable auditing:
   a. **Properties** > **Security** > Select the **Both failed and successful logins** radio button > **OK**.
   b. Disconnect and reconnect the SSMS server > intentionally attempt to log in using invalid credentials.

> ***Note***: *This generated a few "Login Failed" alerts.*

5. View the failed SQL login attempt:
   a. Open **Event Viewer** > **Application** section (to view SQL login attempts) > we see the failed login attempts.



> ***Note***: *Now, anybody can attempt to connect to this VM and SQL database. We're exposing all of this to the internet with the intent to analyze the logs and practice incident response later. We're setting up our environment, making it look enticing to hackers.*

Task 3: Test the Linux “Tester” VM (ping and SSH)

1. **Azure** portal > **Virtual Machines** > start **linux-vm**.
2. Open terminal/PowerShell on our local PC.
    a. Test that we can **ping** linux-vm: **ping <the public IP>**
    b. Test that we can **SSH** (login) into linux-vm: **ssh <username>@<VM public IP>**
        i. “*Are you sure…*”: **yes**
        ii. “*Password*”: **<VM password>**
        iii. Disconnect from the SSH session: **exit**

    ***Note**: Successfully confirmed that we can ping and SSH to linux-vm.*
3. Stop all of the running VMs.

End:

- In windows-vm, we disabled the internal firewall and installed SQL Server.
- Also we logged into linux-vm via SSH.