In this activity, you'll continue in your role as an SOC analyst for a large retailer, Better Buys, Inc.

- Since ecommerce is a large part of your organization's business infrastructure, Better Buys takes various forms of online payment, including credit cards, and must comply with the Payment Card Industry Data Security Standard (PCI DSS).

- The PCI DSS has 12 requirements that organizations must comply with. The first states that organizations must protect their systems with firewalls.

- In this activity, you'll configure uncomplicated firewall (UFW) to ensure that your organization is in compliance with PCI DSS.

- In additions to concerns about being in compliance, one of your junior analysts has detected numerous port scans that originated from public sources. Your organization requires the use of ports `110`, `143`, `587`, `80`, `443` to be open at all times as part of its daily operations.

- To mitigate against adversarial reconnaissance and network fingerprinting as well as being PCI DSS compliant, you've decided to harden your network by filtering all non-essential ports on your perimeter firewall.

- Feel free to use online resources and manual pages to help you through these exercises.

    - Use `ufw version` to determine what version of UFW is installed.

    - Use `sudo ufw reset` to reset all UFW rules to factory defaults.

    - Use `sudo ufw status` to check the current status of the firewall.

    - Use `sudo ufw enable` to start the firewall and update rules.

    - Use `sudo ufw default deny incoming` to block all incoming connections.

    - Use `sudo ufw default allow outgoing` to allow all outgoing connections.

    - Use `sudo ufw allow` to open specific ports.

    - Use `sudo ufw deny` to close specific ports.

- Use `sudo ufw delete` to delete rules.
- Use `sudo ufw disable` to shut down the firewall.
- Use `sudo ufw reload` to reload the UFW firewall.

Set Up

Complete the following steps before beginning the activity:

1. Log into your Ubuntu UFW virtual machine (VM) with the following credentials:
    - Username: `sysadmin`
    - Password: `cybersecurity`
2. Reset all the UFW rules to their factory defaults. This will allow us to customize UFW with our own rule sets.
    - Run: `sudo ufw reset`.
    - When prompted, enter the letter y and press `Enter`.
3. Make sure firewalld is not running. (We'll learn more about firewalld in the next section.)
    - Run: `sudo service firewalld stop`.
    - Run: `sudo service firewalld status`.

Activity Instructions

1. Test that you can SSH into the Ubuntu UFW using Ubuntu firewalld (Other VM). The firewalld VM will be used to test connections to the UFW VM.
    - Log into the firewalld VM with the following credentials:
        - Username: `sysadmin`
        - Password: `cybersecurity`
    - Attempt to use SSH to log into the UFW VM.
        - See if you can use SSH to gain access to your Ubuntu UFW. Make sure to use the IP address for the `Eth0` interface of the Ubuntu UFW.

- Type the command to SSH.

  ssh sysadmin@<ip>

  Type `exit` to terminate the connection.

- Now that you know you can log in using SSH, you'll use UFW to stop that from happening.

2. Switch back to the Ubuntu UFW. Enable UFW and set up default rules that block all incoming and allow all outgoing traffic.

3. Block all incoming SSH connections.

   Sudo ufw deny SSH

4. The protocol Telnet sends unencrypted traffic and is therefore a vulnerability. Configure the firewall to block Telnet.

   Sudo ufw deny telnet

5. Open the following ports to allow incoming information from web servers and mail applications.

   - `80`: For HTTP connections
   - `143`: For IMAP (Internet Message Access Protocol)
   - `587`: For SMTP (Simple Mail Transfer Protocol)
   - `110`: For POP (Post Office Protocol)
   - `443`: For HTTPS (HTTP over SSL)

6. Stop and restart UFW then verify the UFW status.

   - Check the UFW status to see if the rules are still there.

7. Use the SSH protocol to try to connect to the UFW VM from firewalld VM.
    - Type the SSH command to connect to the UFW machine.
    - Press Ctrl+c to stop your terminal from processing.
    - Does this confirm that the Ubuntu UFW VM will block all incoming SSH traffic?
8. Recall that you blocked Telnet but opened up port `80`. What will happen if you try to use Telnet on port `80`?
    - Type the command to `telnet` to port `80`.
    - What do you see? What happens if you type a letter and then hit enter?
    - You will see the web server contacted on port `80` and the `HTTP` headers retrieved.
        - Why is it able to make the connection with Telnet even though you blocked Telnet?
9. Switch back to the UFW VM. Delete your initial rule blocking SSH. It is no longer needed because you are actively blocking all incoming connections.
    - Type the command that deletes the first rule only: `sudo ufw delete #`. (It goes by the line the rule is on.)
        - If you did not block SSH as your first rule, enter `sudo ufw status` and determine the line number of your `deny 22` rule. Add that line number to the end of the command.
    - You will be prompted to delete your `deny 22` rule.

Bonus:

10. You have blocked all incoming traffic except for the protocols deemed essential. Try to SSH once more from the firewalld VM into UFW. What happens? (We'll learn more about firewalld in the next section.)
    - You'll notice you're still blocked even though we deleted the ssh rule, this is because we are still blocking all incoming traffic. We will now add a rule to just allow only the firewalld VM to SSH into UFW.

- Now you will allow the firewalld's IP address to access to your VM via SSH.
    - Type the command that allows traffic to `ufw vm` on port `22` from firewalld VMs IP address.

To recap, you now have a rule in place that blocks all incoming traffic and a rule allowing firewalld VM's IP address to connect on port `22` to UFW VM.

11. Switch back to the firewalld VM and test using SSH again:
    - From the firewalld VM, try to SSH back into the UFW VM.
    - Type the command to SSH into `UFW VM`.
    - Exit out of the active SSH connection.