COMP 4632
Practicing Cybersecurity: Attacks and Counter-measures

Week 4 Lab Exercise

Topic: *Network Vulnerability Scanning*

Lab Objective

In this lab, you will try to perform information gathering via vulnerability scanning on servers. We would also set up firewall rules and IDS to protect your server and detect potential attacks. The whole setup will include the following components:

- Vulnerability Scanning Initiation
- Scanning Procedure and Result Walkthrough
- Configure Firewall
- Configure Intrusion Detection System

Task 1 – Configure and Use Nessus

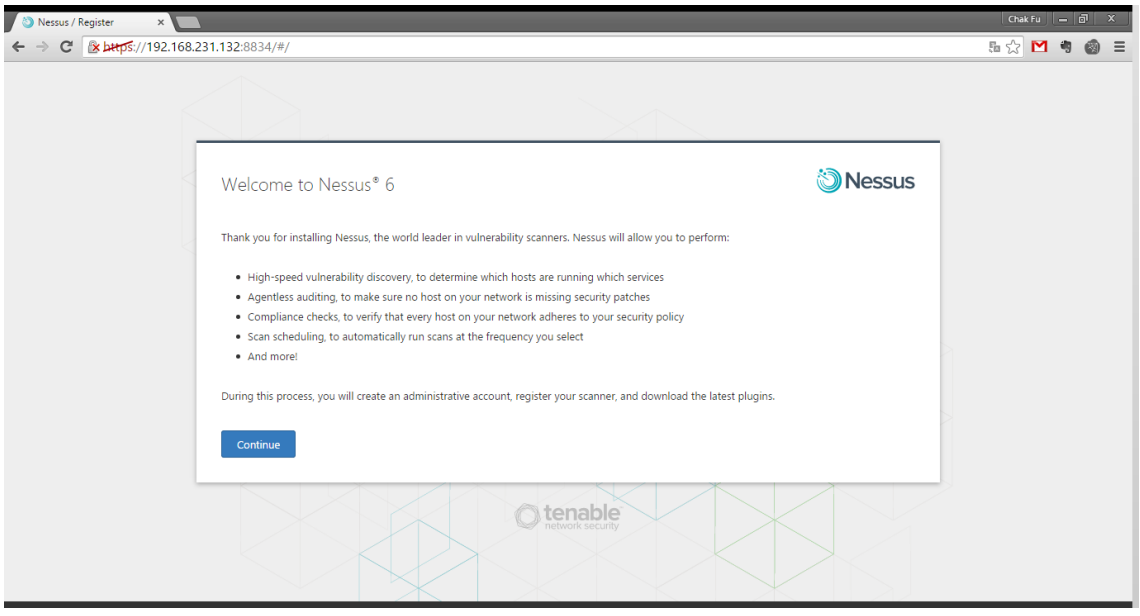
We are positioned as security assessor and perform an internal vulnerability scanning on the servers for audit purpose. This task will let you initiate a vulnerability scan on database server before and after its configuration.

Task 1.1 Install and activate Nessus 6.4.0 (Should be completed in Week 3)

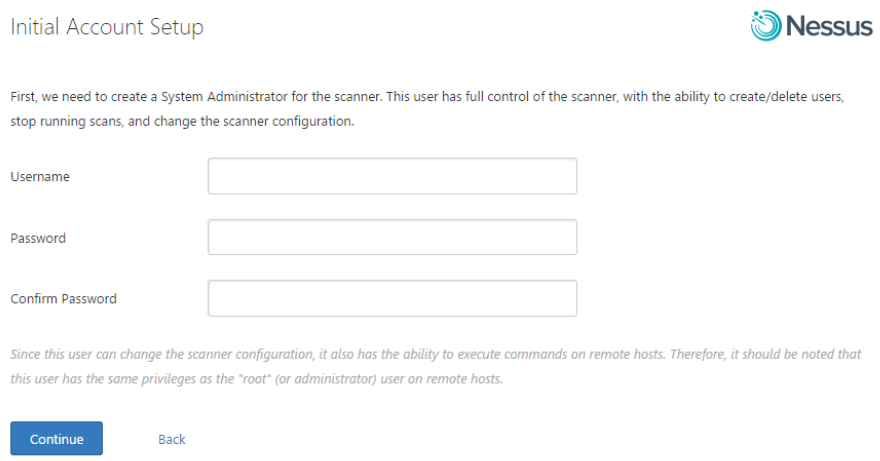
- Drag and drop the Nessus-6.4.0-debian6_amd64.deb file into Kali Linux
- Install Nessus 6.4.0 using the following command

```
dpkg -i Nessus-6.4.0-debian6_amd64.deb
/etc/init.d/nessusd start
```

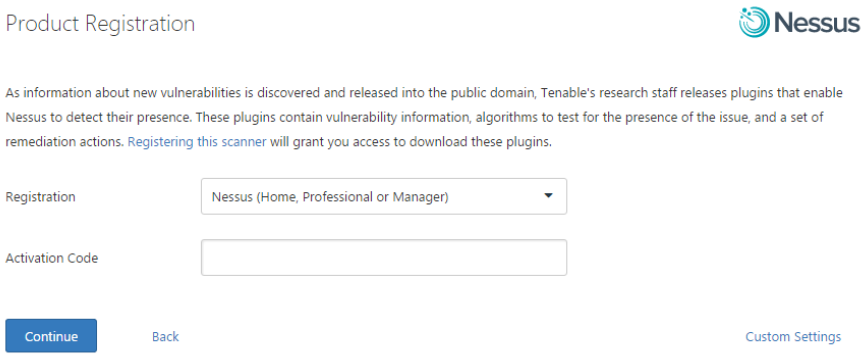
- Access the Nessus web interface via https://<kali_linux_ip>:8834



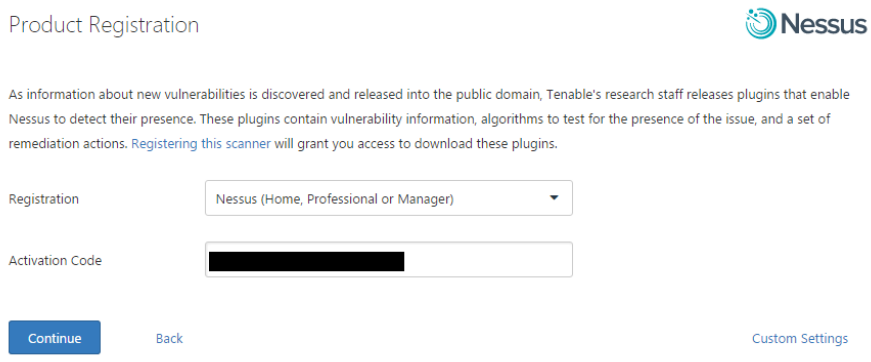
- Create a Nessus user account:
  - Username: comp4632
  - Password: pass4632



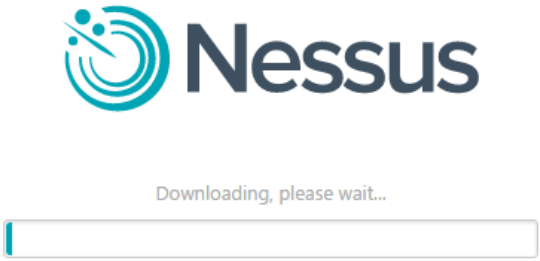
- An activation code would be required for the product registration



- Register for an activation code in the Nessus website
    - https://www.tenable.com/products/nessus-home
- An email containing the activation code would be sent to your email address
- Select “Nessus (Home, Professional or Manager)” and enter the activation code in the Nessus web interface

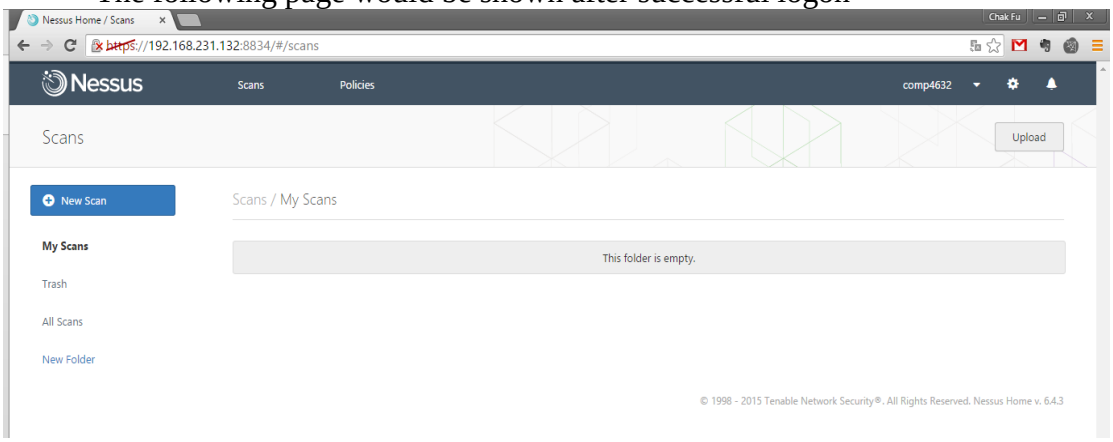


- Nessus will automatically update the plugin and perform initialization

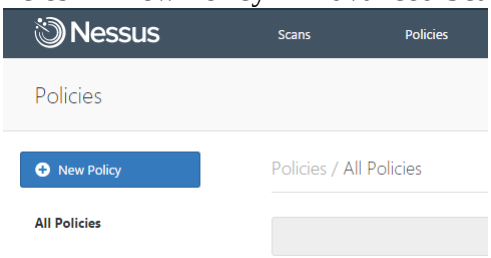


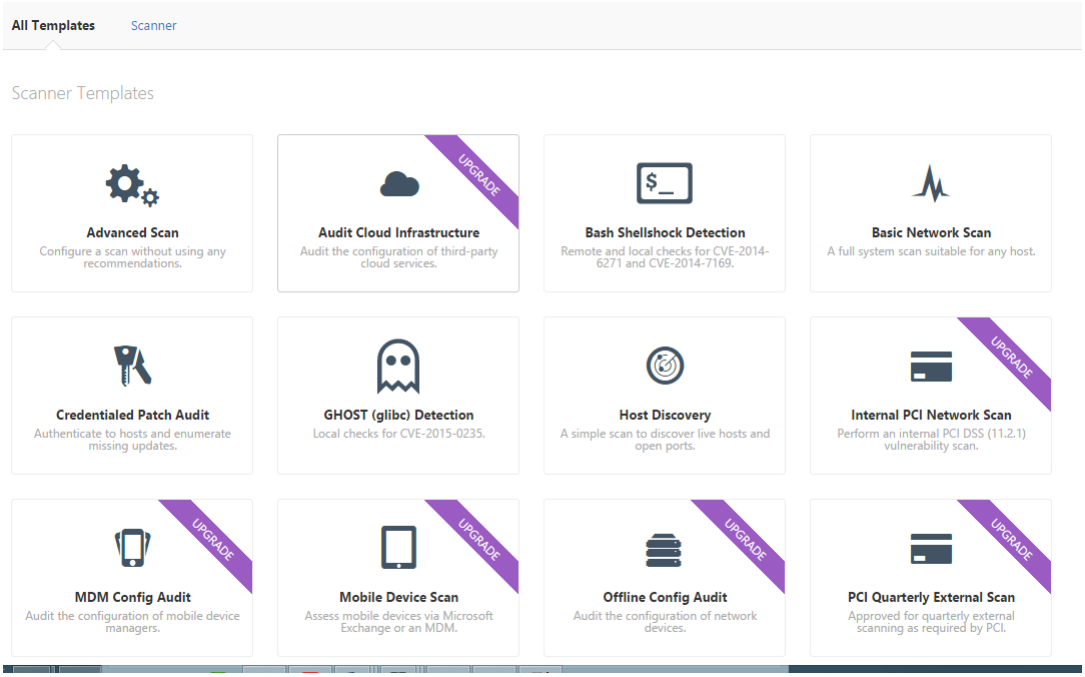
Task 1.2 Create New Nessus Policy

- Power on the Kali Linux virtual machine
- Access the Nessus web interface via https://<kali_linux_ip>:8834
- Login Nessus with the credential created previously
  - Username: comp4632
  - Password : pass4632
- The following page would be shown after successful logon

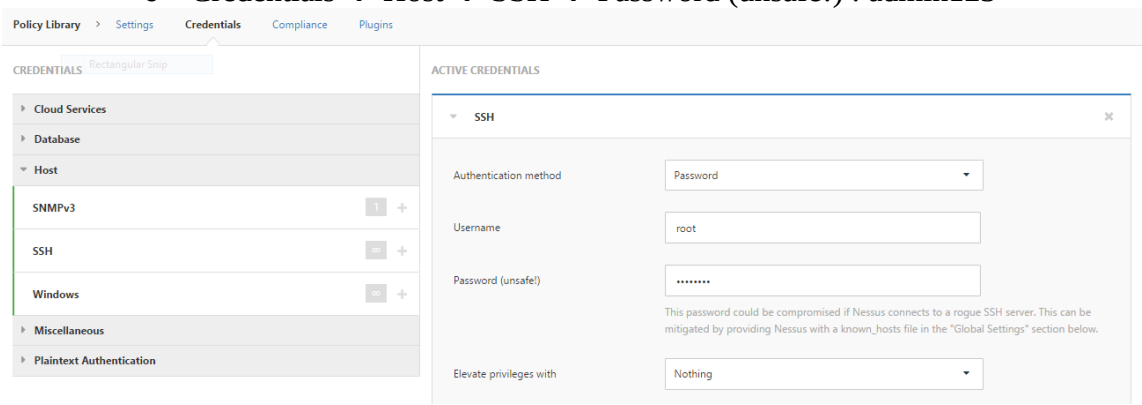


- Create a new scanning policy
  - Select Policies → New Policy→ Advanced Scan

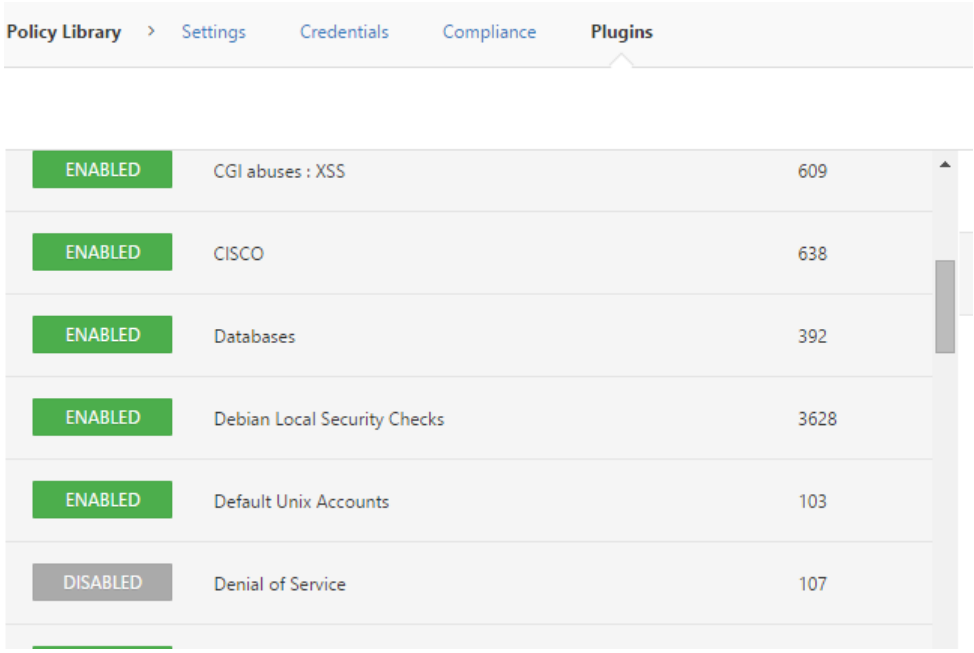




- In Settings tab, enter the following information
  - Basic → General → Name : Comp4632_InternalScan
  - Assessment → General → Override normal accuracy → Show potential false alarms
  - Assessment → General → Perform thorough tests (may disrupt your network or impact scan speed)
  - Assessment → Brute Force → Only user credentials provided by the user (uncheck)
- In Credentials tab, enter the following information
  - Credentials → Host → SSH → Authentication Method: Password
  - Credentials → Host → SSH → Username : root
  - Credentials → Host → SSH → Password (unsafe!) : admin123



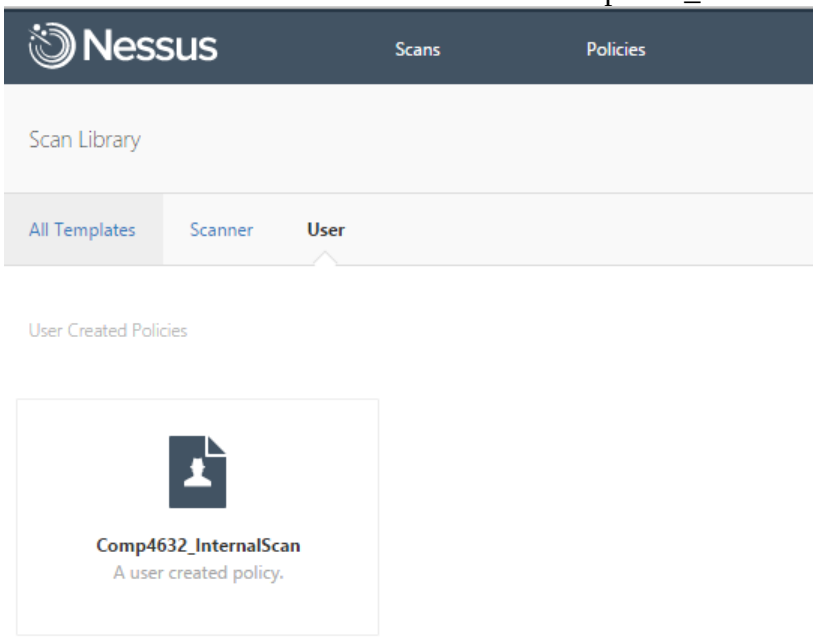
- In Plugin tab, disable the following plugin
  - Denial of Service



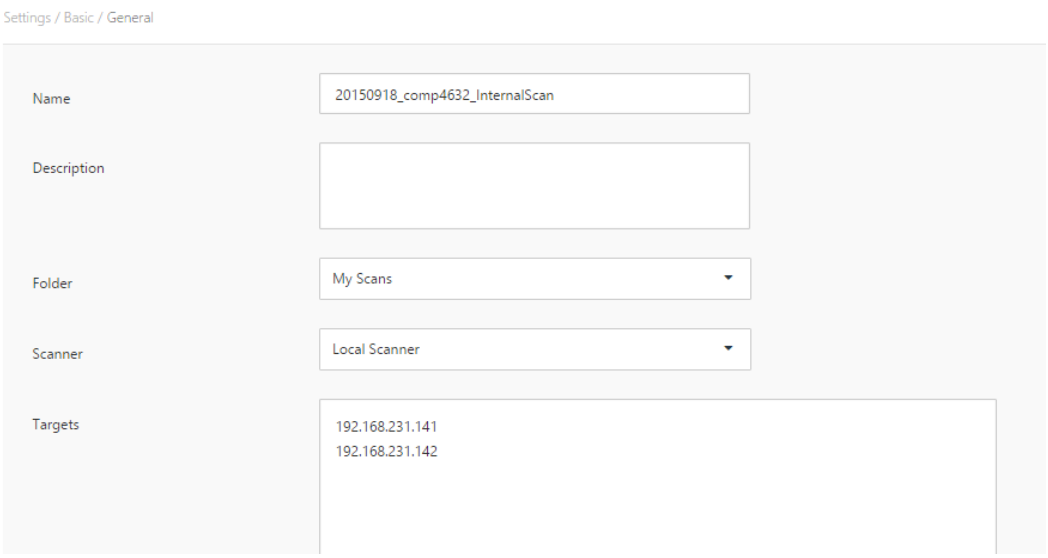
- Save the policy

Task 1.3 Create New Nessus Scan

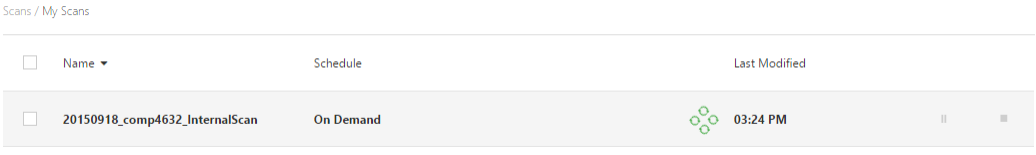
- Select Scans → New Scan → User → Comp4632_InternalScan



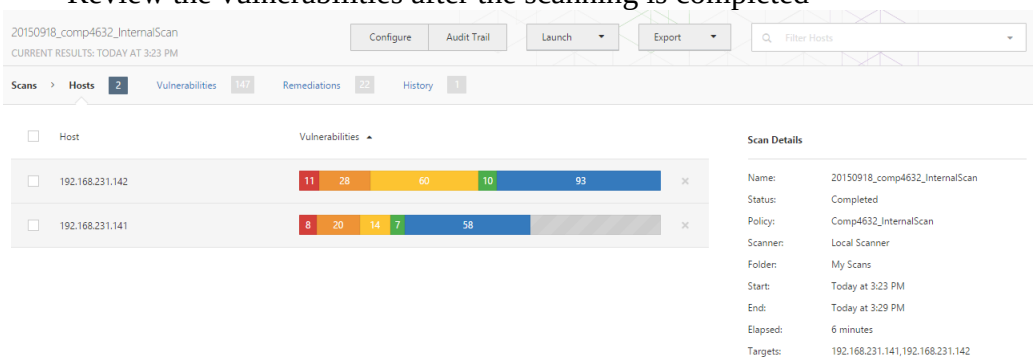
- Enter the following information in Basic → General
  - Name: <Name of the Scan>
  - Description: <Description for the Scan>
  - Targets: <IP address of the webserver and the dbserver>

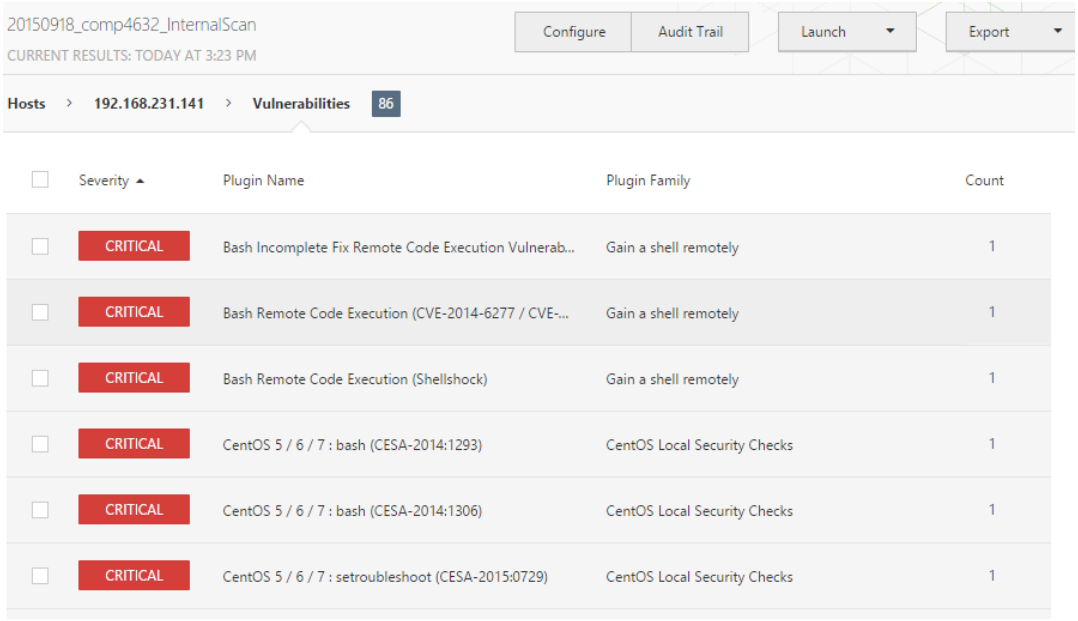


- Click Save and start the scan



- Review the vulnerabilities after the scanning is completed





Bonus Question 1: Please list out what MYSQL related vulnerabilities were found. (1 mark)

Task 1.4 Post-installation and Re-scanning

- Type the following command to harden the MySQL database (https://dev.mysql.com/doc/refman/5.0/en/mysql-secure-installation.html)

```
mysql_secure_installation
```

Question 1: Please state what security features would be changed by mysql secure installation? (1 mark)

- Set “admin123” as MySQL root password and answer Y for all questions
  - Change the root password
  - Remove anonymous users
  - Disallow root login remotely
  - Remove test database and access to it
  - Reload privilege tables
- Add a MySQL user “comp4632” with password “pass4632” for web server to connect

```
mysql -uroot -p
admin123
SHOW DATABASES;
```

```
GRANT ALL ON eightwin.* TO 'comp4632'@'%' IDENTIFIED
BY 'pass4632';
FLUSH PRIVILEGES;
\q
```

- Re-scan the DbServer by Nessus and see how results are different from the previous scan

Bonus Question 2: What is the difference between two scan results? Use one to two sentences to describe the reason. (0.5 mark)

Task 2 – Firewall Configuration

Firewall is an important network security system that monitor and controls the incoming and outgoing traffic based on pre-defined rules. Well defined firewall rules could minimize the attacking surface. In this task, we would go through some basic firewall rules set up and perform some access control testing.

Task 2.1 Reset iptables rules in Kali Linux

- List the current rules set using the following command

```
root@kali:~# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

- Flush all the rules except policy using the following command

```
iptables -F
```

- Currently the INPUT, FORWARD and OUTPUT policies are ACCEPT
- Access the Nessus via *https://<kali_linux_ip>:8834* in the host machine to ensure it could be accessed

Task 2.2 Configure iptables in Kali Linux

- The following command set the default policy to be DROP for all incoming traffic

```
iptables -P INPUT DROP
```

- Check the current rules using the following command and verify that INPUT policy is changed to DROP

```
root@kali:~# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

- Access the Nessus via *https://<kali_linux_ip>:8834* in the host machine again, you should not be able to access it now
- Permit the access to Kali Linux port 8834 explicitly in iptables using the following command

```
iptables -A INPUT -p tcp --dport 8834 -j ACCEPT
```

Options	Parameter	Meaning
-A	INPUT	Append the rule to incoming rules
-p	tcp	Specify the protocol to be tcp
--dport	8834	Specify the destination port to be 8834
-j	ACCEPT	If the packet matches the rule, accept the packet

- Verify if the rule has been added and access the Nessus again via *https://<kali_linux_ip>:8834*

```
root@kali:~# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:8834

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Question 2: What is the response when the host ping the Kali Linux? What rule should be added explicitly to allow this traffic? (0.5 mark)

Question 3: After configuring the rule in previous question, if the default policy for OUTPUT is set to DROP, will you receive response when the host ping the Kali Linux? Please explain the behavior. (0.5 mark)

Task 2.3 Configure iptables in Web Server and Database Server

- Verify you could connect to MySQL in Database Server using the following command in Web Server and Kali Linux

```
mysql -u comp4632 -h <database_server_ip> -p
pass4632
```

- **How should the iptables in Web Server and Database Server be configured to achieve the following target?**
  - In Web Server
    - Allow HTTP access from any host
    - Allow FTP access from any host
    - Allow DNS access from any host
    - Allow Ping from any host
    - Block all other incoming traffic
    - Allow all outgoing traffic
  - In Database Server
    - Allow Ping from any host
    - Allow MySQL connection from Web Server
    - Block any other incoming traffic
    - Allow all outgoing traffic

HINTS:

- http://linux.die.net/man/8/iptables
- **conntrack** module

Web Server:

Database Server:

Question 4: What would happen to the rules if you restart the guest machine now? And how should you fix this? (0.5 mark)

Question 5: How do you verify only the Web Server was able to access the MySQL Database? (0.5 mark)

Task 3 – Snort IDS configuration

In this task, we will walk through the configuration and usage of snort, an intrusion detection system. You are expected to understand the basic concept of the IDS and how to identify potential attacks.

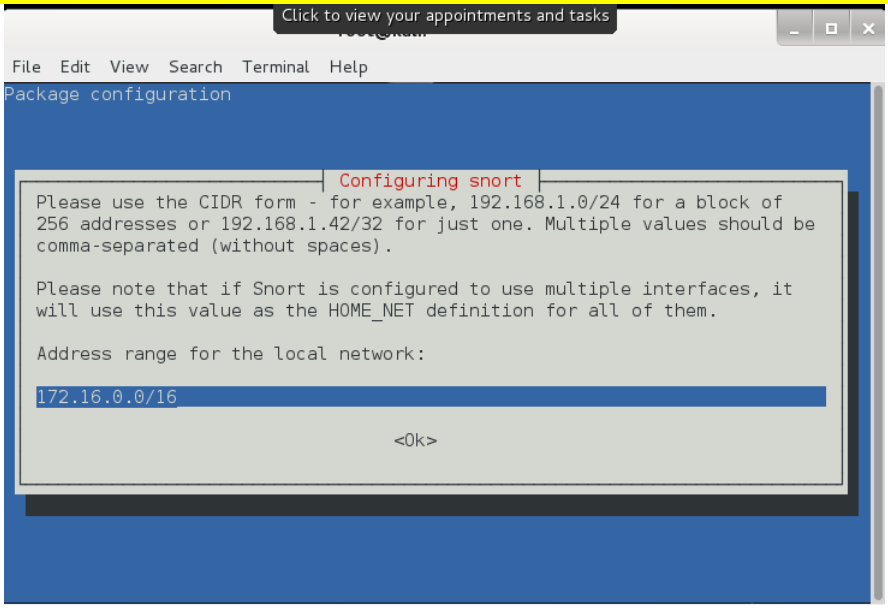
Task 3.1 Configure Snort IDS settings

- Install Snort in Kali Linux using the following command

```
apt-get update
apt-get install snort
```

- Set the local network interface range

YOU NEED TO SPECIFY YOUR OWN LOCAL NETWORK RANGE

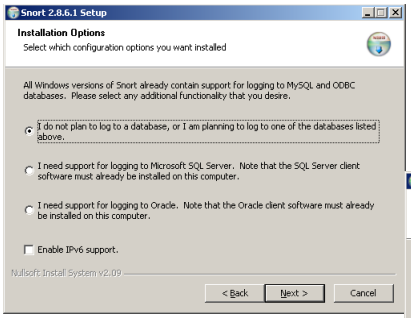


- After the installation, you could check the version of the snort being installed using the following command.

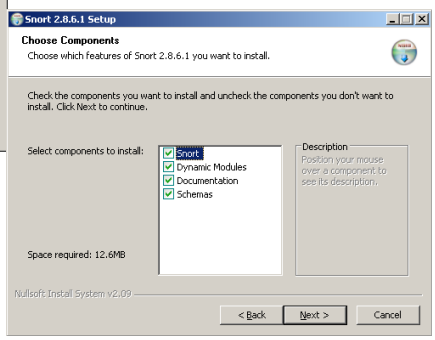
```
snort --version
```

Task 3.2 Configure Snort IDS rules

For Windows



Installation with default



For Mac OSX

- Use the following command to configure snort IDS rules

```
tar -xzvf snortrules-snapshot-xxxx.tar.gz
sudo mv ./etc /etc/snort
sudo mv ./preproc_rules /etc/snort/preproc_rules
sudo mv ./rules /etc/snort/rules
sudo mv ./so_rules /etc/snort/so_rules
sudo chown -R root:wheel /etc/snort
sudo touch /etc/snort/rules/white_list.rules
sudo touch /etc/snort/rules/black_list.rules
```

For Kali Linux

- No reconfiguration is needed

Task 3.3 Launch Snort

- Launch the snort with snort.conf file. (hints: check web page)
- Read the network packet from a previous captured file
- Output logs into a separate alert file

Bonus Question 3: What is the parameter used for launching snort? (0.5 marks)

Task 3.4 Determine the attack method

- Test the provided file and determine what the identified attack was.

Bonus Question 4: What is the attack method being used for gaining the privilege of the system? (1 mark)

End of Lab