Hands-On Excersise3-System Security

Introduction:

In this session, we'll focus on securing our systems within a GNS3 environment. This includes setting up a firewall using iptables, configuring SSH for secure remote access, and using Fail2Ban to protect against brute-force attacks. Finally, we capture and analyze the packets to assess the effectiveness of our system's security measures!

Prerequisites:

Ensure you have GNS3 installed and configured as described in the previous sessions. You should also have Docker installed on your Linux system.

1. Setting Up iptables:

- **iptables**: A command-line firewall utility for Linux that allows you to configure rules for incoming and outgoing traffic.
- **Purpose**: To secure your system by controlling network traffic and blocking unauthorized access.

1.1. Install iptables:

```
# Update package lists to get the latest version information
sudo apt update
# Install iptables package
sudo apt install iptables
```

1.2. Basic iptables Configuration:

```
# Allow all traffic on the loopback interface (localhost)
sudo iptables -A INPUT -i lo -j ACCEPT
# Allow established and related connections
sudo iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
# Allow SSH connections on port 22
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
# Allow HTTP connections on port 80
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
# Drop all other incoming traffic
sudo iptables -A INPUT -j DROP
```

1.3. Save iptables Rules:

```
# Save the current iptables rules to a file
sudo sh -c "iptables-save > /etc/iptables.rules"
#Access the file using
nano /etc/iptables.rules
```

1.4. Load iptables Rules on Boot:

Create a new file /etc/network/if-pre-up.d/iptables:

```
# Open the file in a text editor
sudo nano /etc/network/if-pre-up.d/iptables
```

Add the following content:

```
#!/bin/sh
# Restore iptables rules from the saved file
iptables-restore < /etc/iptables.rules
```

Make the script executable:

```
# Make the script executable
sudo chmod +x /etc/network/if-pre-up.d/iptables
```

2. Configuring SSH:

- **OpenSSH Server**: A suite of secure networking utilities based on the Secure Shell (SSH) protocol.
- **Purpose**: To provide secure remote login and other secure network services over an insecure network.

2.1. Install OpenSSH Server:

```
# Update package lists to get the latest version information
sudo apt update
# Install OpenSSH server package
sudo apt install openssh-server
```

2.2. Configure SSH:

Edit the SSH configuration file:

```
# Open the SSH configuration file in a text editor
sudo nano /etc/ssh/sshd_config
```

Make the following changes:

```
# Disallow root login via SSH
PermitRootLogin no
# Allow password authentication
PasswordAuthentication yes
# Restrict SSH access to specific users
AllowUsers user_name
```

Restart the SSH service:

```
# Restart SSH service to apply changes
sudo systemctl restart ssh
```

3. Setting Up Fail2Ban:

- **Fail2Ban**: An intrusion prevention software framework that protects servers from brute-force attacks.
- **Purpose**: To automatically block IP addresses that show signs of malicious activity, such as repeated failed login attempts.

3.1. Install Fail2Ban:

```
# Update package lists to get the latest version information
sudo apt update
# Install Fail2Ban package
sudo apt install fail2ban
```

3.2. Configure Fail2Ban:

Create a new file /etc/fail2ban/jail.local:

```
# Open the file in a text editor
sudo nano /etc/fail2ban/jail.local
```

Add the following content:

```
# Default configuration
[DEFAULT]
# Ban time in seconds
bantime = 600
# Time frame for detecting multiple failed attempts
findtime = 600
# Maximum retries before banning
maxretry = 3
# SSH configuration
[sshd]
# Enable SSH protection
enabled = true
```

Restart Fail2Ban:

```
# Restart Fail2Ban service to apply changes
sudo systemctl restart fail2ban
```

Connecting GNS3 to Linux Systems(We discussed it in session2):

- **Cloud Node in GNS3**: Represents a bridge between the virtual GNS3 environment and your real network.
- **Purpose**: To connect GNS3 virtual devices to your actual network, allowing interaction and testing in a more realistic setting.

4. Introduction to Wireshark

Wireshark is a powerful network protocol analyzer that lets you capture and interactively browse the traffic running on a computer network. It is widely used for network troubleshooting, analysis, and protocol development.

4.1. Prerequisites

Wireshark Installation: Ensure that Wireshark is installed on your Kali Linux system. You can install it using the following command:

```
sudo apt update
sudo apt install wireshark
```

Administrative Privileges: You may need administrative privileges to capture network traffic. To allow non-root users to capture packets, run:

```
sudo dpkg-reconfigure wireshark-common
sudo usermod -aG wireshark $USER
newgrp wireshark
```

4.2. Step-by-Step Guide

Step 1: Open Wireshark

Launch Wireshark from the application menu or by typing wireshark in the terminal.
You will see a list of available network interfaces.

Step 2: Select the Network Interface

Choose the network interface that you want to monitor. This is usually the interface connected to the internet (e.g., eth0, wlan0).
Double-click on the interface to start capturing packets.

Step 3: Capture Network Traffic

Once you start the capture, you will see packets being listed in real-time.
Perform the network activity you want to monitor. For example, visit a website, download a file, or run a network-based application.

Step 4: Apply Filters

To make the analysis easier, you can apply filters. Wireshark uses a powerful filtering language.
Some common filters:
HTTP traffic: http
TCP traffic: tcp
IP address: ip.addr == 192.168.1.1
Port: tcp.port == 80

Step 5: Analyze Captured Traffic

Click on a packet to view its details. Wireshark provides a breakdown of each packet, including the protocol used, source and destination addresses, and other details. Use the bottom pane to view the raw packet data.

Step 6: Save the Capture

To save the captured traffic, go to File > Save As and choose a location and file name.
You can also export specific packets or summaries.

Step 7: Stop the Capture

To stop the capture, click the red square button on the toolbar.
You can now analyze the captured data offline.

4.3. Tips for Effective Monitoring

- ❖ Use specific filters to narrow down the traffic to what is relevant for your analysis.
- ❖ Save your captures regularly, especially if monitoring for long periods.
- ❖ Use Wireshark’s built-in statistics tools (found under the Statistics menu) to get a high-level view of the traffic.

Practice 3: Capturing Traffic from GNS3 Systems to Kali Linux

In this practice session, you will apply the knowledge gained from the previous sections to capture and analyze network traffic between your GNS3 environment and your Kali Linux system. Follow the steps below:

Objective

- Capture and analyze network traffic from GNS3 systems to your Kali Linux machine.
- Determine what types of traffic (e.g., HTTP, TCP, other protocols) are passing through.

Steps

1. **Set Up Wireshark on Kali Linux**
   - Ensure Wireshark is installed and you have the necessary administrative privileges to capture network traffic:

```
sudo apt update
sudo apt install wireshark
sudo dpkg-reconfigure wireshark-common
sudo usermod -aG wireshark $USER
newgrp wireshark
```

2. **Start Wireshark**
   - Launch Wireshark from the application menu or by typing `wireshark` in the terminal.
   - Select the appropriate network interface (e.g., `eth0`, `wlan0`) to monitor.
3. **Capture Network Traffic**
   - Begin capturing packets by double-clicking on the chosen network interface.
   - Generate network traffic from your GNS3 systems directed to your Kali Linux machine. This could include activities like:
     - Ping commands from GNS3 systems to your Kali Linux IP.
     - Accessing a web server running on your Kali Linux machine from a browser on GNS3 systems.
     - Transferring files using protocols such as FTP or SCP.
4. **Apply Filters in Wireshark**
   - Use Wireshark filters to narrow down the captured traffic:
     - To see HTTP traffic: `http`
     - To see TCP traffic: `tcp`
     - To filter by IP address: `ip.addr == <Your Kali Linux IP>`
     - To filter by port: `tcp.port == 80` (for HTTP) or any other relevant port.
5. **Analyze Captured Traffic**
   - Click on individual packets to view their details.
   - Examine the protocols used, source and destination addresses, and other relevant details.
   - Use the bottom pane to view raw packet data and understand the communication between GNS3 systems and your Kali Linux machine.
6. **Save and Review the Capture**
   - Save the captured traffic for future reference:
     - Go to `File` > `Save As` and choose a location and file name.
   - Review the captured data offline if necessary.

Observations and Analysis

- Identify the types of traffic (e.g., HTTP, TCP) captured during the session.
- Determine if all expected traffic is passing through and identify any unexpected traffic types.
- Record your findings and consider how the captured traffic aligns with the security configurations set up earlier (e.g., iptables rules).

By completing this practice session, you will enhance your ability to use Wireshark for monitoring and analyzing network traffic in a real-world scenario, integrating knowledge from both system security and network analysis perspectives.