**PS：**[tcpdump](http://www.ha97.com/tag/tcpdump" \o "tcpdump)**是一个用于截取网络分组，并输出分组内容的工具，简单说就是数据包抓包工具。tcpdump凭借强大的功能和灵活的截取策略，使其成为**[Linux](http://www.ha97.com/category/linux" \o "Linux)**系统下用于网络分析和问题排查的首选工具。**

tcpdump提供了源代码，公开了接口，因此具备很强的可扩展性，对于网络维护和入侵者都是非常有用的工具。tcpdump存在于基本的[Linux](http://www.ha97.com/tag/linux" \o "Linux)系统中，由于它需要将网络界面设置为混杂模式，普通用户不能正常执行，但具备[root](http://www.ha97.com/tag/root" \o "root)权限的用户可以直接执行它来获取网络上的信息。因此系统中存在网络分析工具主要不是对本机安全的威胁，而是对网络上的其他计算机的安全存在威胁。

**一、概述**  
顾名思义，tcpdump可以将网络中传送的数据包的“头”完全截获下来提供分析。它支持针对网络层、协议、主机、网络或端口的过滤，并提供and、or、not等逻辑语句来帮助你去掉无用的信息。

# tcpdump -vv  
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes  
11:53:21.444591 IP (tos 0x10, ttl  64, id 19324, offset 0, flags [DF], proto 6, length: 92) asptest.localdomain.ssh > 192.168.228.244.1858: P 3962132600:3962132652(52) ack 2726525936 win 1266  
asptest.localdomain.1077 > 192.168.228.153.domain: [bad udp cksum 166e!]  325+ PTR? 244.228.168.192.in-addr.arpa. (46)  
11:53:21.446929 IP (tos 0x0, ttl  64, id 42911, offset 0, flags [DF], proto 17, length: 151) 192.168.228.153.domain > asptest.localdomain.1077:  325 NXDomain q: PTR? 244.228.168.192.in-addr.arpa. 0/1/0 ns: 168.192.in-addr.arpa. (123)  
11:53:21.447408 IP (tos 0x10, ttl  64, id 19328, offset 0, flags [DF], proto 6, length: 172) asptest.localdomain.ssh > 192.168.228.244.1858: P 168:300(132) ack 1 win 1266  
347 packets captured  
1474 packets received by filter  
745 packets dropped by kernel

不带参数的tcpdump会收集网络中所有的信息包头，数据量巨大，必须过滤。

**二、选项介绍**

-A 以ASCII格式打印出所有分组，并将链路层的头最小化。

-c 在收到指定的数量的分组后，tcpdump就会停止。

-C 在将一个原始分组写入文件之前，检查文件当前的大小是否超过了参数file\_size 中指定的大小。如果超过了指定大小，则关闭当前文件，然后在打开一个新的文件。参数 file\_size 的单位是兆字节（是1,000,000字节，而不是1,048,576字节）。

-d 将匹配信息包的代码以人们能够理解的汇编格式给出。

-dd 将匹配信息包的代码以c语言程序段的格式给出。

-ddd 将匹配信息包的代码以十进制的形式给出。

-D 打印出系统中所有可以用tcpdump截包的网络接口。

-e 在输出行打印出数据链路层的头部信息。

-E 用spi@ipaddr algo:secret解密那些以addr作为地址，并且包含了安全参数索引值spi的[IPsec](http://www.ha97.com/tag/ipsec" \o "IPsec) ESP分组。

-f 将外部的[Internet](http://www.ha97.com/tag/internet" \o "Internet)地址以数字的形式打印出来。

-F 从指定的文件中读取表达式，忽略命令行中给出的表达式。

-i 指定监听的网络接口。

-l 使标准输出变为缓冲行形式，可以把数据导出到文件。

-L 列出网络接口的已知数据链路。

-m 从文件module中导入SMI MIB模块定义。该参数可以被使用多次，以导入多个MIB模块。

-M 如果tcp报文中存在TCP-MD5选项，则需要用secret作为共享的验证码用于验证TCP-MD5选选项摘要（详情可参考RFC 2385）。

-b 在数据-链路层上选择协议，包括ip、arp、rarp、ipx都是这一层的。

-n 不把网络地址转换成名字。

-nn 不进行端口名称的转换。

-N 不输出主机名中的域名部分。例如，‘nic.ddn.mil‘只输出’nic‘。

-t 在输出的每一行不打印时间戳。

-O 不运行分组分组匹配（packet-matching）代码优化程序。

-P 不将网络接口设置成混杂模式。

-q 快速输出。只输出较少的协议信息。

-r 从指定的文件中读取包(这些包一般通过-w选项产生)。

-S 将tcp的序列号以绝对值形式输出，而不是相对值。

-s 从每个分组中读取最开始的snaplen个字节，而不是默认的68个字节。

-T 将监听到的包直接解释为指定的类型的报文，常见的类型有rpc远程过程调用）和snmp（简单网络管理协议；）。

-t 不在每一行中输出时间戳。

-tt 在每一行中输出非格式化的时间戳。

-ttt 输出本行和前面一行之间的时间差。

-tttt 在每一行中输出由date处理的默认格式的时间戳。

-u 输出未解码的NFS句柄。

-v 输出一个稍微详细的信息，例如在ip包中可以包括ttl和服务类型的信息。

-vv 输出详细的报文信息。

-w 直接将分组写入文件中，而不是不分析并打印出来。

**三、tcpdump的表达式介绍**

表达式是一个正则表达式，tcpdump利用它作为过滤报文的条件，如果一个报文满足表 达式的条件，则这个报文将会被捕获。如果没有给出任何条件，则网络上所有的信息包 将会被截获。

在表达式中一般如下几种类型的关键字：

**第一种**是关于类型的关键字，主要包括host，net，port，例如 host 210.27.48.2， 指明 210.27.48.2是一台主机，net 202.0.0.0指明202.0.0.0是一个网络地址，port 23 指明端口号是23。如果没有指定类型，缺省的类型是host。

**第二种**是确定传输方向的关键字，主要包括src，dst，dst or src，dst and src， 这些关键字指明了传输的方向。举例说明，src 210.27.48.2 ，指明ip包中源地址是 210.27.48.2 ， dst net 202.0.0.0 指明目的网络地址是202.0.0.0。如果没有指明 方向关键字，则缺省是src or dst关键字。

**第三种**是协议的关键字，主要包括fddi，ip，arp，rarp，tcp，udp等类型。Fddi指明是在FDDI (分布式光纤数据接口网络)上的特定的网络协议，实际上它是”ether”的别名，fddi和ether 具有类似的源地址和目的地址，所以可以将fddi协议包当作ether的包进行处理和分析。 其他的几个关键字就是指明了监听的包的协议内容。如果没有指定任何协议，则tcpdump 将会 监听所有协议的信息包。

除了这三种类型的关键字之外，其他重要的关键字如下：gateway， broadcast，less， greater， 还有三种逻辑运算，取非运算是 ‘not ‘ ‘! ‘， 与运算是’and’，’&&’;或运算是’or’ ，’||’； 这些关键字可以组合起来构成强大的组合条件来满足人们的需要。

**四、输出结果介绍**

下面我们介绍几种典型的tcpdump命令的输出信息

**(1) 数据链路层头信息**  
使用命令：  
#tcpdump --e host ICE  
ICE 是一台装有linux的主机。它的MAC地址是0：90：27：58：AF：1A H219是一台装有[Solaris](http://www.ha97.com/category/unix/solaris" \o "Solaris)的SUN工作站。它的MAC地址是8：0：20：79：5B：46； 上一条命令的输出结果如下所示：

21:50:12.847509 eth0 < 8:0:20:79:5b:46 0:90:27:58:af:1a ip 60: h219.33357 > ICE.  telne t 0:0(0) ack 22535 win 8760 (DF)

21：50：12是显示的时间， 847509是ID号，eth0 <表示从网络接口eth0接收该分组， eth0 >表示从网络接口设备发送分组， 8:0:20:79:5b:46是主机H219的MAC地址， 它表明是从源地址H219发来的分组. 0:90:27:58:af:1a是主机ICE的MAC地址， 表示该分组的目的地址是ICE。 ip 是表明该分组是IP分组，60 是分组的长度， h219.33357 > ICE. telnet 表明该分组是从主机H219的33357端口发往主机ICE的 TELNET(23)端口。 ack 22535 表明对序列号是222535的包进行响应。 win 8760表明发 送窗口的大小是8760。

**(2) ARP包的tcpdump输出信息**

使用命令：  
#tcpdump arp

得到的输出结果是：

22:32:42.802509 eth0 > arp who-has route tell ICE (0:90:27:58:af:1a)  
22:32:42.802902 eth0 < arp reply route is-at 0:90:27:12:10:66 (0:90:27:58:af:1a)

22:32:42是时间戳， 802509是ID号， eth0 >表明从主机发出该分组，arp表明是ARP请求包， who-has route tell ICE表明是主机ICE请求主机route的MAC地址。 0:90:27:58:af:1a是主机 ICE的MAC地址。

**(3) TCP包的输出信息**

用tcpdump捕获的TCP包的一般输出信息是：

src > dst: flags data-seqno ack window urgent options

src > dst:表明从源地址到目的地址， flags是TCP报文中的标志信息，S 是SYN标志， F (FIN)， P (PUSH) ， R (RST) “.” (没有标记); data-seqno是报文中的数据 的顺序号， ack是下次期望的顺序号， window是接收缓存的窗口大小， urgent表明 报文中是否有紧急指针。 Options是选项。

**(4) UDP包的输出信息**

用tcpdump捕获的UDP包的一般输出信息是：

route.port1 > ICE.port2: udp lenth

UDP十分简单，上面的输出行表明从主机route的port1端口发出的一个UDP报文 到主机ICE的port2端口，类型是UDP， 包的长度是lenth。

**五、举例**

(1) 想要截获所有210.27.48.1 的主机收到的和发出的所有的分组：  
#tcpdump host 210.27.48.1

(2) 想要截获主机210.27.48.1 和主机210.27.48.2或210.27.48.3的通信，使用命令（注意：括号前的反斜杠是必须的）：  
#tcpdump host 210.27.48.1 and (210.27.48.2 or 210.27.48.3 )

(3) 如果想要获取主机210.27.48.1除了和主机210.27.48.2之外所有主机通信的ip包，使用命令：  
#tcpdump ip host 210.27.48.1 and ! 210.27.48.2

(4) 如果想要获取主机192.168.228.246接收或发出的ssh包，并且不转换主机名使用如下命令：  
#tcpdump -nn -n src host 192.168.228.246 and port 22 and tcp

(5) 获取主机192.168.228.246接收或发出的ssh包，并把mac地址也一同显示：  
# tcpdump -e src host 192.168.228.246 and port 22 and tcp -n -nn

(6) 过滤的是源主机为192.168.0.1与目的网络为192.168.0.0的报头：  
tcpdump src host 192.168.0.1 and dst net 192.168.0.0/24

(7) 过滤源主机物理地址为XXX的报头：  
tcpdump ether src 00:50:04:BA:9B and dst……  
（为什么ether src后面没有host或者net？物理地址当然不可能有网络喽）。

(8) 过滤源主机192.168.0.1和目的端口不是telnet的报头，并导入到tes.t.txt文件中：  
Tcpdump src host 192.168.0.1 and dst port not telnet -l > test.txt

**ip icmp arp rarp 和 tcp、udp、icmp这些选项等都要放到第一个参数的位置，用来过滤数据报的类型。**

**例题：如何使用tcpdump监听来自eth0适配卡且通信协议为port 22，目标来源为192.168.1.100的数据包资料？**

答：tcpdump -i eth0 -nn port 22 and src host 192.168.1.100

**例题：如何使用tcpdump抓取访问eth0适配卡且访问端口为tcp 9080？**

答:tcpdump -i eth0 dst 172.168.70.35 and tcp port 9080

**例题：如何使用tcpdump抓取与主机192.168.43.23或着与主机192.168.43.24通信报文，并且显示在控制台上**

tcpdump -X -s 1024 -i eth0 host (192.168.43.23 or 192.168.43.24) and  host 172.16.70.35

------------------------------

抓包命令：  
tcpdump tcp -i eth0 -c 3000 -s 0 and src net 10.17.140.173 -w /opt/test/20151221.pcap

src net 10.17.178.61：抓取来此IP地址的TCP包

tail -f 10 XXXX

查看XXXX文件前10行  
可以查看日志  
tcpdump tcp -i eth0 -c 3000 -s 0 -w /opt/test/20151130.pcap

tcpdump -i eth0 -p udp port 1812 and host 10.17.178.25 -nne

tcpdump -i eth0 -c 3000 -s 0 -w /opt/test/20151221.pcap

-c 在收到指定的数量的分组后，tcpdump就会停止。  
-C 在将一个原始分组写入文件之前，检查文件当前的大小是否超过了参数file\_size 中指定的大小。如果超过了指定大小，则关闭当前文件，  
然后在打开一个新的文件。参数 file\_size 的单位是兆字节（是1,000,000字节，而不是1,048,576字节）。

tcpdump tcp -i eth1 -t -s 0 -c 100 and dst port ! 22 and src net 192.168.1.0/24 -w ./target.cap  
------------------------------------------------------  
可以用tcpdump -D查看一下网卡接口，找到lo的接口ID，例如是1,那么用命令tcpdump -i 1 tcp 9000 -w temp.pcap //结果存入temp.pcap，再用wiresharp之类的分析软件就可以分析啦。

--------------------------------------------------------

(1)tcp: ip icmp arp rarp 和 tcp、udp、icmp这些选项等都要放到第一个参数的位置，用来过滤数据报的类型  
(2)-i eth1 : 只抓经过接口eth1的包  
(3)-t : 不显示时间戳  
(4)-s 0 : 抓取数据包时默认抓取长度为68字节。加上-S 0 后可以抓到完整的数据包  
(5)-c 100 : 只抓取100个数据包  
(6)dst port ! 22 : 不抓取目标端口是22的数据包  
(7)src net 192.168.1.0/24 : 数据包的源网络地址为192.168.1.0/24  
(8)-w ./target.cap : 保存成cap文件，方便用ethereal(即wireshark)分析  
---------------------------------  
后台抓包， 控制台退出也不会影响：  
nohup tcpdump -i eth1 port 110 -w /tmp/xxx.cap &  
-----------------------------------

tcpdump 的抓包保存到文件的命令参数是-w xxx.cap  
抓eth1的包  
tcpdump -i eth1 -w /tmp/xxx.cap  
抓 192.168.1.123的包  
tcpdump -i eth1 host 192.168.1.123 -w /tmp/xxx.cap  
抓192.168.1.123的80端口的包  
tcpdump -i eth1 host 192.168.1.123 and port 80 -w /tmp/xxx.cap  
抓192.168.1.123的icmp的包  
tcpdump -i eth1 host 192.168.1.123 and icmp -w /tmp/xxx.cap  
抓192.168.1.123的80端口和110和25以外的其他端口的包  
tcpdump -i eth1 host 192.168.1.123 and ! port 80 and ! port 25 and ! port 110 -w /tmp/xxx.cap  
抓vlan 1的包  
tcpdump -i eth1 port 80 and vlan 1 -w /tmp/xxx.cap  
抓pppoe的密码  
tcpdump -i eth1 pppoes -w /tmp/xxx.cap  
以100m大小分割保存文件， 超过100m另开一个文件 -C 100m  
抓10000个包后退出 -c 10000  
后台抓包， 控制台退出也不会影响：  
nohup tcpdump -i eth1 port 110 -w /tmp/xxx.cap &  
抓下来的文件可以直接用ethereal 或者wireshark打开。 wireshark就是新版的ethereal，程序换名了