ngrep

方便的数据包匹配和显示工具

补充说明

ngrep命令 是grep命令的网络版，他力求更多的grep特征，用于搜寻指定的数据包。正由于安装ngrep需用到libpcap库， 所以支持大量的操作系统和网络协议。能识别TCP、UDP和ICMP包，理解bpf的过滤机制。

安装

ngrep命令的下载地址：http://ngrep.sourceforge.net/，libpcap下载地址：http://www.tcpdump.org/，先用`yum install libpcap`完全安装libpcap，注意有时候用libpcap安装包安装的不完整会影响ngrep的使用。

如果yum无法安装就用以下步骤安装libpcap，

```
wget http://www.tcpdump.org/release/libpcap-1.3.0.tar.gz
tar -zxf libpcap-1.3.0.tar.gz
cd libpcap-1.3.0
./configure
make && make install
```

ngrep的安装就是 configure/make/make install 三部曲。

注，configure时是遇到 please wipe out all unused pcap installations，添加以下选项：

```
./configure --with-pcap-includes=/usr/local/include/pcap
```

在安装后输入ngrep来验证下安装是否成功。

语法

```
ngrep <-LhNXViwqpevxlDtTRM> <-IO pcap_dump> <-n num> <-d dev> <-A num>
<-s snaplen> <-S limitlen> <-w normal|byline|single|none> <-c cols>
<-P char> <-F file> <match expression> <bpf filter>
```

选项

```
-e # 显示空数据包
-i # 忽略大小写
-v # 反转匹配
-R # don't do privilege revocation logic
-x # 以16进制格式显示
-X # 以16进制格式匹配
-w # 整字匹配
-p # 不使用混杂模式
-l # make stdout line buffered
-D # replay pcap_dumps with their recorded time intervals
-t # 在每个匹配的包之前显示时间戳
-T # 显示上一个匹配的数据包之间的时间间隔
-M # 仅进行单行匹配
```

```
-I # 从文件中读取数据进行匹配
-O # 将匹配的数据保存到文件
-n # 仅捕获指定数目的数据包进行查看
-A # 匹配到数据包后dump随后的指定数目的数据包
-s # set the bpf caplen
-S # set the limitlen on matched packets
-W # 设置显示格式byline将解析包中的换行符
-c # 强制显示列的宽度
-P # set the non-printable display char to what is specified
-F # 使用文件中定义的bpf(Berkeley Packet Filter)
-N # 显示由IANA定义的子协议号
-d # 使用哪个网卡，可以用-L选项查询
-L # 查询网卡接口
```

实例

捕捉cloudian□18080端口的request和response□-W byline用来解析包中的换行符，否则包里的所有数据都是连续的，可读性差。-d lo是监听本地网卡：

```
ngrep -W byline -d lo port 18080
```

捕捉amazon□80端口的request和response□-d eth0是用来监听对外的网卡：

```
ngrep -W byline -d eth0 port 80
```

可以用-d any来捕捉所有的包：

```
ngrep '[a-zA-Z]' -t -W byline -d any tcp port 18080
```

捕获字符串.flv，比如要查看在Web Flash 视频中的.flv文件的下载地址：

```
ngrep -d3 -N -q \.flv
interface: \Device\TNT_40_1_{670F6B50-0A13-4BAB-9D9E-994A833F5BA9}
(10.132.0.0/2
55.255.192.0)
match: \.flv
```

打开一个视频页面：

```
T(6) 10.132.34.23:24860 -> 61.142.208.154:80 [AP]
GET /f59.c31.56.com/flvdownload/12/19/ggyg7741@56.com_56flv_zhajm_119556973
97.flv HTTP/1.1..accept: */*..Referer: http://www.56.com/flashApp/v_player_
site.swf..x-flash-version: 9,0,45,0..UA-CPU: x86..Accept-Encoding: gzip, de
flate..User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET
CLR 2.0.50727; .NET CLR 3.0.04506.30)..host: f59.r.56.com..Connection: Keep
-Alive..Cookie: whistoryview=23423759-23635627-23423344-23171935-23058374-2
3081156-23207350-22395727-; geoip=.....; wl_all_s=y....
```

OK□地址已经找到了,就是http://f59.c31.56.com/flvdownload/12/19/ggyg7741@56.com_56flv_zhajm_11955697397.flv

加个-W byline参数后,将解析包中的换行符：

```
T(6) 2007/11/25 15:56:12.192619 10.132.34.23:26365 -> 59.151.21.101:80 [AP]
GET /aa.flv HTTP/1.1.
Accept: */*.
Accept-Language: zh-cn.
UA-CPU: x86.
Accept-Encoding: gzip, deflate.
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; .NET CLR
2.0.5072
7; .NET CLR 3.0.04506.30).
Host: www.google.cn.
Connection: Keep-Alive.
Cookie:
PREF=id=a0b2932c336477e9:TB=4:NW=1:TM=1187877372:LM=1187956074:S=Y1Fzndp
rT3vFo7ac;
SID=DQAAAHcAAABJCEXeOVLHu2rIfb5BfKP3GG9PbhJDEkXsLTV8y0f_lvSd2Y46Q0FPt
83CnEs9rxA1xBDM9mLR8-
ckWeScyOQA8PyYnX5u5OjFvjfRbDg_FDZfwxhRzqS9KPZv26pjnsUxs0FDM
1xpJ5AgDn38pXtlCdkksJ0-cbiIWoA61oHWMg;
NID=7=AvJxn5B6YOLLxoYz4LLzhIbNsQUQiulRS6U
JGxdBniQBmXm99y7L-
NBNORN82N3unmZSGHFPfePVHnLK2MjYjglyXZhU9x7ETXNBnY3NurNijHDhJ7K
yi7E53UBOcv4V.
```