MC833 A - Programação de redes de computadores
Relatório - Tarefa 02

093175 - Victor Fernando Pompeo Barbosa - *victorfpb@gmail.com*

31 de março de 2016

Prof. Paulo Licio de Geus
IC – UNICAMP

Sumário

1 Introdução

Nesta tarefa estudaremos a ferramenta `tcpdump`, utilizada para análise de protocolos e pacotes de rede. A ferramenta permite ao usuário a exibição de pacotes transmitidos por uma interface de rede presente no computador.

2 Desenvolvimento

O exercício se baseia em um cenário no qual um programa transmite um arquivo da máquina *willow* para a máquina *maple* sobre uma conexão TCP. A ferramenta `tcpdump` foi executada no transmissor para registrar os pacotes enviados e os pacotes de reconhecimento recebidos.

O arquivo `tcpdump.dat` contém o log de todos os pacotes TCP do cenário descrito e foi obtido a partir de um link fornecido na página da disciplina. O primeiro passo envolveu a conversão do arquivo `tcpdump.dat` para o formato texto, por meio do comando `tcpdump -r tcpdump.dat > outfile.txt`.

3 Questões

1. As interfaces nas quais o `tcpdump` pode escutar/capturar dados podem ser obtidas por meio do comando `tcpdump -D --list-interfaces`. A saída obtida no terminal após a execução do comando pode ser verificada a seguir.

```
niko@ubuntu:~/Desktop/mc833/t2$ tcpdump -D --list-interfaces
1.eth0 [Up, Running]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.bluetooth-monitor (Bluetooth Linux Monitor)
5.nflog (Linux netfilter log (NFLOG) interface)
6.nfqueue (Linux netfilter queue (NFQUEUE) interface)
7.usbmon1 (USB bus number 1)
8.usbmon2 (USB bus number 2)
```

   As interfaces incluem as interfaces mostradas pelo comando `ifconfig`, mas não estão limitadas a elas. O comando `tcpdump` também inclui, além delas, interfaces de Bluetooth e USB, por exemplo.

2. Para poder observar o endereço IP dos nós, a leitura do arquivo de log pode ser feita usando o comando `tcpdump -r tcpdump.dat > outfile-ip.txt -n`. Esse comando evita que os endereços sejam traduzidos para nomes.

   A primeira linha do arquivo de saída está transcrita abaixo. Nela, é possível observar que o endereço IP do nó *willow* é `128.30.4.222`, enquanto o do nó *maple* é `128.30.4.223`.

```
21:34:41.473036 ARP, Request who-has 128.30.4.223 tell 128.30.4.222, length 28
```

3. Para poder observar o endereço MAC dos nós, a leitura do arquivo de log pode ser feita usando o comando `tcpdump -t tcpdump.dat > outfile-mac.txt -e`. Esse comando imprime os endereços MAC dos nós em cada linha.

   As primeiras linhas do arquivo de saída estão transcritas abaixo. Nelas, é possível observar que o endereço MAC do nó *willow* é `00:16:ea:8e:28:44` e que o endereço MAC do nó *maple* é `00:16:ea:8d:e5:8a`.

```
21:34:41.473036 00:16:ea:8e:28:44 (oui Unknown) > Broadcast, ethertype ARP (0x0806), length 42:
    Request who-has maple.csail.mit.edu tell willow.csail.mit.edu, length 28
21:34:41.473505 00:16:ea:8d:e5:8a (oui Unknown) > 00:16:ea:8e:28:44 (oui Unknown), ethertype ARP
    (0x0806), length 42: Reply maple.csail.mit.edu is-at 00:16:ea:8d:e5:8a (oui Unknown),
    length 28
```

```
21:34:41.473518 00:16:ea:8e:28:44 (oui Unknown) > 00:16:ea:8d:e5:8a (oui Unknown), ethertype
    IPv4 (0x0800), length 74: willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [S],
    seq 1258159963, win 14600, options [mss 1460,sackOK,TS val 282136473 ecr 0,nop,wscale 7],
    length 0
```

4. As portas TCP usadas pelos nós podem ser obtidas da saída do comando obtida com a flag tt -n. Uma linha qualquer da saída está representada a seguir.

```
21:34:41.474232 IP 128.30.4.222.39675 > 128.30.4.223.5001: Flags [.], seq 2921:4369, ack 1, win
    115, options [nop,nop,TS val 282136474 ecr 282202089], length 1448
```

   É possível observar que o nó com endereço `128.30.4.222` (o nó *willow*) se comunica a partir da porta 39675, enquanto o nó que possui endereço `128.30.4.223` (o nó *maple*) se comunica a partir da porta 5001.

   A porta 5001, usada pelo nó *maple*, é uma porta registrada. Portas registradas são atribuídas pela IANA (Internet Assigned Numbers Authority) para uso por um certo protocolo ou aplicação.

5. O número de kilobytes transferido pode ser deduzido do número de sequência do último pacote enviado/-recebido.

```
21:34:44.329956 IP maple.csail.mit.edu.5001 > willow.csail.mit.edu.39675: Flags [.], ack
    1572890, win 820, options [nop,nop,TS val 282204945 ecr 282139320], length 0
```

   Isso indica que o nó *maple* recebeu todos os bytes desde o byte #0 até o byte #1572889. Dessa maneira, foram transferidos aproximadamente 1,573 MB ou 1 573 kB durante essa sessão.

   A duração da sessão pode ser calculada pela diferença dos *timestamps* que marcam o início e fim da sessão.

```
21:34:41.473036 ARP, Request who-has maple.csail.mit.edu tell willow.csail.mit.edu, length 28
```

```
21:34:44.339015 IP willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [.], ack 2, win
    115, options [nop,nop,TS val 282139339 ecr 282204955], length 0
```

   Portanto, a sessão durou $2,865979$ segundos. A vazão do fluxo TCP entre os nós foi de, aproximadamente, 548 813 kB/s.

6. A linha referente ao envio do pacote `1473:2921` está representada a seguir. Na sequência, podemos observar a linha referente a seu acknowledgement.

```
21:34:41.474225 IP willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [.], seq
    1473:2921, ack 1, win 115, options [nop,nop,TS val 282136474 ecr 282202089], length 1448
```

```
21:34:41.482047 IP maple.csail.mit.edu.5001 > willow.csail.mit.edu.39675: Flags [.], ack 2921,
    win 159, options [nop,nop,TS val 282202095 ecr 282136474], length 0
```

   O round-trip time (RTT) entre os dois nós, baseado no pacote 1473:2921, é, portanto, igual a $0,007822$ segundo.

   Por outro lado, as linhas referentes ao envio do pacote `13057:14505` e ao recebimento de seu acknowledgement estão representadas a seguir.

```
21:34:41.474992 IP willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [.], seq
    13057:14505, ack 1, win 115, options [nop,nop,TS val 282136474 ecr 282202090], length 1448
```

```
21:34:41.499373 IP maple.csail.mit.edu.5001 > willow.csail.mit.edu.39675: Flags [.], ack 14505,
    win 331, options [nop,nop,TS val 282202114 ecr 282136474], length 0
```

   Nesse caso, o round-trip time (RTT) entre os dos nós foi de $0,024381$ segundo. A diferença nos dois valores dá-se por conta de a velocidade com que os pacotes são enviados/recebidos é influenciada pelo tráfego de rede e o volume de dados cuja transmissão está planejada.

7. O procedimento de *three-way handshake* é utilizado para dar início à conexão e é identificado pela flag `[S]`. Podemos observá-lo na saída representada a seguir.

```
21:34:41.473518 IP willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [S], seq
    1258159963, win 14600, options [mss 1460,sackOK,TS val 282136473 ecr 0,nop,wscale 7],
    length 0
21:34:41.474055 IP maple.csail.mit.edu.5001 > willow.csail.mit.edu.39675: Flags [S.], seq
    2924083256, ack 1258159964, win 14480, options [mss 1460,sackOK,TS val 282202089 ecr
    282136473,nop,wscale 7], length 0
21:34:41.474079 IP willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [.], ack 1, win
    115, options [nop,nop,TS val 282136474 ecr 282202089], length 0
```

O procedimento de *connection termination* é identificado pela flag `[F]`. Podemos observá-lo na saída representada a seguir.

```
21:34:44.339007 IP maple.csail.mit.edu.5001 > willow.csail.mit.edu.39675: Flags [F.], seq 1, ack
    1572890, win 905, options [nop,nop,TS val 282204955 ecr 282139320], length 0
21:34:44.339015 IP willow.csail.mit.edu.39675 > maple.csail.mit.edu.5001: Flags [.], ack 2, win
    115, options [nop,nop,TS val 282139339 ecr 282204955], length 0
```