Complemento ao Capítulo 3 da apostila de redes 9ª edição

3.9 Comandos complementares

3.9.1 mii-tool

Instale com: # apt-get install mii-tool

Utilizado para mostrar a situação de interfaces de rede. O comando é capaz de mostrar se a interface está conectada a um ativo de rede ou a outra interface. Com a opção -w, mostra constantemente a situação da interface.

3.9.2 netcat

Instale com: # apt-get install netcat

O netcat é similar ao telnet, que faz conexões TCP. No entanto, é capaz de realizar conexões UDP com a opção -u. Também pode ser utilizado para gerar um tráfego do tipo cliente-servidor. Aceita nc como comando (ao invés de netcat).

3.9.3 tcpflow

Instale com: # apt-get install tcpflow

O tcpflow lê dumps do tcpdump com dados TCP e extrai o payload dos mesmos. Útil para análises de conteúdo. Utilizar "# tcpflow -r arq.dump".

3.9.4 bittwist

Instale com: # apt-get install bittwist

Permite construir pacotes de rede.

3.9.5 wicd

Instale com: # apt-get install wicd

Excelente gerenciador de conexões de rede, incluindo wireless, para ambiente gráfico. Encontra e conecta redes conhecidas automaticamente. Cada usuário que pretender utilizar o wicd deverá estar presente no grupo netdev (em /etc/group). Assim, execute: # adduser *usuário* netdev. Será necessário realizar logout/logon no ambiente gráfico depois dessa operação (adduser).

3.9.6 kismet

Instale com: # apt-get install kismet

Scanner para descobrir redes wireless. Necessita de configuração. Leia a sua documentação (arquivo /usr/share/doc/kismet/README.gz).

3.9.7 airodump-ng

Instale com: # apt-get install aircrack-ng

Outro scanner wireless.

3.9.8 windump

Tcpdump para MS Windows. Idêntico ao tcpdump. Siga os passos:

- **Baixe e instale o winpcap, disponível em http://tiny.cc/windump.**
- **Baixe o windump, disponível em http://tiny.cc/windump, colocando-o dentro de c:\windows.**
- **Execute, no prompt do MS DOS, o comando windump -D. Identifique o número correspondente à sua placa de rede.**
- **Utilize o Windump com o comando: windump -i <nr_placa_rede>, seguido de outros parâmetros desejados (utilize os mesmos do tcpdump).**

3.10 Complementos a comandos e erratas

3.10.1 ping

Com a opção -R realiza um record route. A opção -s permite definir o tamanho do pacote enviado. A opção -i define o intervalo de tempo de envio. Estas duas últimas opções são úteis em testes de fluxo de dados nas redes. É possível localizar placas e equipamentos com defeitos, mediante tráfego mais pesado.

3.10.2 telnet

O telnet é utilizado para realizar conexões TCP. Também pode controlar máquinas remotamente. Para conexões UDP (e, opcionalmente, TCP), utilizar o netcat.

3.10.3 mtr-tiny

A instalação será feita com o comando apt-get install mtr-tiny.

3.10.4 tcpdump

Na opção -s, o valor correto para ver todo o conteúdo de um frame ethernet é 1518 e não 1500. No entanto, seria melhor utilizar sempre "-s0". Ainda, a opção -e mostra também a camada de enlace na captura.

3.10.5 ipcalc

É possível fazer com que o ipcalc calcule as redes de um determinado range. Exemplo: "# ipcalc 200.20.20.20-200.25.20.20".

3.11 O comando route

O comando route edita as tabelas de roteamento de rede (roteamento estático). A máquina, ao ser desligada, perderá os dados inseridos. Convém colocar os comandos de rotas no arquivo /etc/init.d/rc.local. Outra alternativa seria colocar os comandos dentro do arquivo de configuração de rede do Debian, como será mostrado adiante.

A sintaxe utilizada é a seguinte:

Sintaxe	Função
# route	**Mostra a tabela de roteamento, resolvendo nomes de hosts.**
# route -n	**Mostra a tabela de roteamento, sem resolver nomes de hosts.**
# route add -net <rd> netmask <mrd> <adpt>	**Roteia um adaptador local de rede para uma rede de destino.**
# route add -host <hd> <adpt>	**Roteia um adaptador local de rede para um host de destino.**
# route add default gw <hr>	**Estabelece um roteador default (default gateway).**
# route add -net <rd> netmask <mrd> gw <hr>	**Estabelece o roteador (gateway) a ser utilizado para atingir uma determinada rede.**
# route add -net <rd> netmask <mrd> reject	**Impede o roteamento desta máquina para a rede mencionada.**
# route add -host <hd> reject	**Impede o roteamento desta máquina para o host mencionado.**

Obs:

<rd> - rede destino. Ex: 10.0.0.0.
<mrd> - máscara da rede destino. Ex: 255.0.0.0.
<adpt> - adaptador local de rede. Ex: eth0, ppp0.
<hd> - host destino. Ex 10.5.5.56.
<hr> - host roteador. Ex: 10.0.0.1.

Para remover qualquer elemento da tabela de roteamento, bastará trocar o add por del.

Caso você deseje adicionar os roteamentos no fim do arquivo de configuração de redes (/etc/network/interfaces), bastará inserir a expressão "up" antes do comando. Exemplo:

```
auto eth0
iface eth0 inet static
   address 10.0.0.1
   netmask 255.0.0.0
   network 10.0.0.0
   broadcast 10.255.255.255
   gateway 10.10.10.250

up route add -net 172.20.0.0 netmask 255.255.0.0 gw 10.0.0.89
```