

Lab 3 – Aplicativos para verificar e configurar os parâmetros do TCP/IP e diagnosticar o atraso dos pacotes e traçar rotas em redes TCP/IP

Objetivo:

i) Utilizar aplicativos aplicativos padrão verificar a configuração de redes baseadas nos protocolos TCP/IP e testar a conectividade em rede. ii) Utilizar aplicativos aplicativos para diagnóstico do atraso de pacotes e traçado de rotas em redes baseadas nos protocolos TCP/IP.

ifconfig:

Somente o usuário **root** pode executar o aplicativo **ifconfig**.

No laboratório de Redes I, foi modificada as permissões de execução deste comando para que usuários normais possam executá-lo, contudo, deve-se digitar o caminho completo para que o comando possa ser encontrado **/sbin/ifconfig**.

Se nenhum argumento for passado na chamada do **ifconfig**, o comando mostra a configuração atual de cada interface de rede.

Consultar as páginas **man** do Linux para maiores detalhes sobre o funcionamento deste aplicativo, o qual permite ativar/desativar a interface, configurar o endereço IP, definir o tamanho da MTU, redefinir o endereço de hardware se a interface suporta, redefinir a interrupção utilizada pelo dispositivo, entre outros.

Exemplo:

```
[root@jagger root]# ifconfig
eth0  Encapsulamento do Link: Ethernet  Endereço de HW 00:07:95:BB:37:41
      inet end.: 192.168.1.35  Bcast:192.168.1.255  Masc:255.255.255.0
      endereço inet6: fe80::207:95ff:febb:3741/64  Escopo:Link
      UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
      RX packets:7543 errors:0 dropped:0 overruns:0 frame:0
      TX packets:4052 errors:0 dropped:0 overruns:0 carrier:0
      colisões:0 txqueuelen:1000
      RX bytes:3006444 (2.8 Mb)  TX bytes:602747 (588.6 Kb)
      IRQ:11  Endereço de E/S:0xd400

lo    Encapsulamento do Link: Loopback Local
      inet end.: 127.0.0.1  Masc:255.0.0.0
      endereço inet6: ::1/128  Escopo:Máquina
      UP LOOPBACKRUNNING  MTU:16436  Métrica:1
      RX packets:263 errors:0 dropped:0 overruns:0 frame:0
      TX packets:263 errors:0 dropped:0 overruns:0 carrier:0
      colisões:0 txqueuelen:0
      RX bytes:31287 (30.5 Kb)  TX bytes:31287 (30.5 Kb)
```

Algumas informações importantes:

Sobre as interfaces:

eth0 -> Configuração da interface (placa de rede) eth0

eth1, eth2, ... -> Configuração de outras interfaces, se houver.

lo -> Loopback local, utilizado para testes locais dos protocolos TCP/IP nos sistemas baseados no Unix

Parâmetros principais da interface Eth0:

Encapsulamento do Link: Ethernet -> Tipo do enlace

inet end: 192.168.1.35 -> Endereço IPv4

Masc: 255.255.255.0 -> Máscara de redes

Bcast:192.168.1.255 -> Endereço de broadcast

Endereço de HW: 00:07:95:BB:37:41 -> Endereço físico da interface

endereço inet6: fe80::207:95ff:febb:3741/64 Escopo:Link -> Endereço IPv6 de escopo local gerado por autoconfiguração

MTU: 1500 -> *Maximum Transfer Unit* – Tamanho máximo do pacote suportado pelo enlace.

ping:

Aplicativo que permite a um usuário verificar se um *host* remoto está ativo. É bastante utilizado para detectar problemas de comunicação na rede.

O **ping** está baseado no envio de mensagens de *solicitação de eco* (*echo request*) e de *resposta de eco* (*echo reply*). Estas mensagens fazem parte do rol de mensagens do protocolo ICMP, que é um protocolo de reportagem de erros, componente do protocolo IP.

Consultar as páginas **man** do ping para verificar as possibilidades de uso deste aplicativo.

O **ping** é um dos principais comandos a disposição do administrador de rede no sentido de verificar a conectividade em rede. Por exemplo, se houver resposta de um ping a partir de um servidor remoto, significa que a máquina local está rodando corretamente o TCP/IP, o enlace local está funcionando corretamente, o roteamento entre a origem e o destino está operando, e por fim, a máquina remota também está rodando corretamente o TCP/IP.

Exemplo 1:

```
-bash-2.05b$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=63 time=0.233 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=63 time=0.266 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=63 time=0.287 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=63 time=0.317 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=63 time=0.216 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=63 time=0.247 ms
--- 192.168.1.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5000ms
rtt min/avg/max/mdev = 0.216/0.261/0.317/0.033 ms
```

No exemplo foram enviados 6 pacotes ICMP, cada um com um número de seqüência (*icmp_seq*), os quais foram recebidos com sucesso com o tempo de viagem assinalado (*time*). Cada pacote tem ainda um *tempo de vida* (*ttl*), o qual é decrementado em cada roteador, sendo o pacote descartado quando chegar a zero; isto evita pacotes perdidos na rede. Quando o ping é interrompido (CTRL-C), uma estatística é apresentada indicando o percentual de pacotes transmitidos, recebidos e perdidos. O tempo de viagem (*rtt* – *round trip time*) mínimo (*min*), médio (*avg*) e máximo (*max*) é calculado, assim como o desvio padrão (*mdev*).

Exemplo 2:

Se desejarmos descobrir o endereço IP de vários *hosts* em uma rede local, poderemos utilizar o **ping** no endereço de *broadcast*. Neste caso, deve-se utilizar a opção **-b**.

```
[cantu@jagger Labs]$ ping -b 192.168.1.255
WARNING: pinging broadcast address
PING 192.168.1.255 (192.168.1.255) 56(84) bytes of data.
64 bytes from 192.168.1.35: icmp_seq=1 ttl=64 time=0.082 ms
64 bytes from 192.168.1.22: icmp_seq=1 ttl=64 time=0.230 ms (DUP!)
64 bytes from 192.168.1.254: icmp_seq=1 ttl=64 time=0.267 ms (DUP!)
64 bytes from 192.168.1.10: icmp_seq=1 ttl=254 time=0.870 ms (DUP!)
64 bytes from 192.168.1.14: icmp_seq=1 ttl=64 time=1.03 ms (DUP!)
64 bytes from 192.168.1.4: icmp_seq=1 ttl=64 time=1.65 ms (DUP!)
64 bytes from 192.168.1.12: icmp_seq=1 ttl=60 time=2.54 ms (DUP!)
64 bytes from 192.168.1.13: icmp_seq=1 ttl=64 time=6.20 ms (DUP!)
```

```
64 bytes from 192.168.1.11: icmp_seq=1 ttl=64 time=6.54 ms (DUP!)
64 bytes from 192.168.1.35: icmp_seq=2 ttl=64 time=0.063 ms
64 bytes from 192.168.1.4: icmp_seq=2 ttl=64 time=0.178 ms (DUP!)
64 bytes from 192.168.1.22: icmp_seq=2 ttl=64 time=0.203 ms (DUP!)
64 bytes from 192.168.1.254: icmp_seq=2 ttl=64 time=0.233 ms (DUP!)
--- 192.168.1.255 ping statistics ---
2 packets transmitted, 2 received, +11 duplicates, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.063/2.053/6.542/2.411 ms
```

Exercícios:

Enviar ping para diferentes hosts e comparar tempos de resposta:

- no endereço local de loopback;
- máquina de um colega do laboratório;
- servidor e roteador da rede da escola;
- servidores externos:

www.ufsc.br (UFSC)

www.diario.com.br (Diário Catarinense)

www.folha.com.br (Folha de São Paulo)

www.clarin.com.ar (Clarín – Argentina)

www.liberation.fr (Liberation – França)

www.nytimes.com (New York Times – USA)

Explicar as possíveis diferenças entre os tempos de resposta dos ping realizados.

traceroute:

Somente o usuário **root** pode executar o aplicativo **traceroute**.

No laboratório de Redes I, foi modificada as permissões de execução deste comando para que usuários normais possam executá-lo, contudo, deve-se digitar o caminho completo para que o comando possa ser encontrado **/usr/sbin/traceroute**.

O *traceroute*, que é capaz de traçar uma rota aproximada entre dois *hosts*. Este comando também usa mensagens ICMP. Para determinar o nome e o endereço dos roteadores entre a fonte e o destino, o *traceroute* na fonte envia uma série de datagrama IP ordinários ao destino. O primeiro datagrama tem o TTL igual a 1, o segundo 2, o terceiro 3, e assim por diante, e inicia temporizadores para cada datagrama. Quando o enésimo datagrama chega ao enésimo roteador, este verifica que o tempo de sobrevivência do datagrama acaba de terminar. Pelas regras do IP, o datagrama é então descartado e uma mensagem ICMP de advertência é enviada a fonte com o nome do roteador e seu endereço IP. Quando a resposta chega de volta a fonte, a mesma calcula o tempo de viagem em função dos temporizadores.

Exemplo:

```
[root@jagger cantu]# traceroute 200.135.233.1
traceroute to 200.135.233.1 (200.135.233.1), 30 hops max, 38 byte packets
 1 sol.uned.sj (192.168.1.254) 0.734 ms 0.151 ms 0.128 ms
 2 hendrix.sj.cefetsc.edu.br (200.135.233.1) 0.206 ms 0.212 ms 0.191 ms
```

O exemplo mostra a rota dos pacotes entre dois computadores da rede do CEFET, o computador (192.168.1.35), onde foi executado o comando, e o servidor hendrix (200.135.233.1). Veja pelo mapa da rede do CEFET que entre estes dois computadores está o host sol.uned.sj (192.168.1.254).

Exercícios:

5. Traçar a rota dos pacotes entre seu computador e diferentes hosts:
 - máquina de um colega do laboratório;
 - servidor e roteador da rede da escola;
 - servidores externos.

6. Explicar as possíveis diferenças entre os tempos de resposta dos ping realizados, evidenciando os trechos onde os atrasos são maiores.
7. Para os servidores externos, acompanhe a rota de entrada/saída dos pacotes da rede interna.

Configuração manual do TCP/IP no Linux

Somente o usuário **root** pode alterar a configuração do TCP/IP.

Arquivo `/etc/sysconfig/network` configura nome do host e roteador padrão:

```
[root@jagger root]# cat /etc/sysconfig/network
HOSTNAME=jagger
NETWORKING=yes
GATEWAY=192.168.1.254
```

Arquivo `/etc/sysconfig/network-scripts/ifcfg-eth0` configura parâmetros do TCP/IP para a interface apropriada:

```
[root@jagger root]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=static
IPADDR=192.168.1.35
NETMASK=255.255.255.0
NETWORK=192.168.1.0
BROADCAST=192.168.1.255
ONBOOT=yes
MII_NOT_SUPPORTED=yes
```

Caso queira configurar para usar IP dinâmico (DHCP), configurar o arquivo `/etc/sysconfig/network-scripts/ifcfg-eth0` da seguinte forma:

```
[root@jagger root]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

Arquivo `/etc/hosts` é o primeiro arquivo usado para resolver nomes. Atualmente utiliza-se um servidor DNS, todavia pode ainda ser utilizado:

```
[root@jagger root]# cat /etc/hosts
127.0.0.1    Jagger localhost
```

Arquivo `/etc/resolv.conf` informa quem é o servidor DNS:

```
[root@jagger root]# cat /etc/resolv.conf
search uned.sj
nameserver 192.168.1.1
```

Cada vez que a configuração do TCP/IP for alterada, deve-se reiniciar os serviços de rede como root:

```
[root@jagger root]# service network restart
```

ou reiniciar a interface de rede:

```
[root@jagger root]# ifdown eth0
[root@jagger root]# ifup eth0
```

Para testar o funcionamento da configuração do TCP/IP, fazer **ifconfig** e verificar se a configuração está correta.

Testar a conectividade de sua máquina fazendo **ping** em outra máquina da rede local.

Configuração do TCP/IP a partir do ambiente gráfico

O Linux também permite realizar configuração do TCP/IP a partir do ambiente gráfico, a partir do menu Sistema -> Configurar seu computador. Quando utilizar este recurso, a senha do usuário **root** será requisitada.

Para o sistema operacional Windows, a configuração do TCP/IP pode ser realizada a partir do Painel de Controle -> Conexões de Rede.