Comandos básicos - Linux

Redes de Computadores

António Pinto
apinto@estgf.ipp.pt

ESTGF | POLITÉCNICO DO PORTO

Fevereiro 2012

1 Comandos básicos de rede - *Linux*

Neste capítulo será descrita a funcionalidade de um pequeno conjunto de comandos de rede que poderão ser executados num terminal. Muito embora o conjunto de comandos de rede apresentado esteja disponível em muitas plataformas (*Linux*, *Windows*), a sua utilização apresenta algumas diferenças de plataforma para plataforma.

A plataforma adoptada nos exemplos deste Capítulo é a plataforma *Linux*, em particular a distribuição Ubuntu.

1.1 ifconfig

Permite visualizar a configuração IP de um computador. Quando executado sem parâmetros apresenta toda a informação que dispõem para todos os interfaces de rede disponíveis (ver Listagem 1). Em particular são apresentados os endereços MAC (**HWaddr**) e os endereços IP (**inet addr**) de cada interface, caso os tenha.

```
aluno@pc ~ $ ifconfig                                                    1
eth0      Link encap:Ethernet  HWaddr 03:23:b3:63:93:37                  2
          UP BROADCAST MULTICAST  MTU:1500  Metric:1                     3
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0             4
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0           5
          collisions:0 txqueuelen:1000                                   6
```

```
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
          Interrupt:17

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:9245 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9245 errors:0 dropped:0 overruns:0 carrier
              :0
          collisions:0 txqueuelen:0
          RX bytes:626177 (626.1 KB)  TX bytes:626177 (626.1 KB)

wlan0   Link encap:Ethernet  HWaddr 03:23:d3:93:3a:50
          inet addr:192.168.1.12  Bcast:192.168.1.255  Mask
              :255.255.255.0
          inet6 addr: fe80::224:d6ff:fe98:4a50/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:527256 errors:0 dropped:0 overruns:0 frame
              :0
          TX packets:390209 errors:0 dropped:0 overruns:0
              carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:665653414 (665.6 MB)  TX bytes:179846472
              (179.8 MB)
```

Listing 1: ifconfig

A Listagem 2 demonstra a utilização do comando com recurso ao parâmetro **lo**, que identifica o interface interno de *loopback*. Este interface tem sempre o endereço IP 127.0.0.1.

```
aluno@pc ~ $ ifconfig lo
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:9245 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9245 errors:0 dropped:0 overruns:0 carrier
              :0
          collisions:0 txqueuelen:0
          RX bytes:626177 (626.1 KB)  TX bytes:626177 (626.1 KB)
```

Listing 2: Comando ifconfig lo

O comando pode ainda ser utilizado para desligar (*down*), ligar (*up*) ou atribuir um endereço IP (1.1.1.1 no exemplo) a uma interface de rede (*eth0* no exemplo), como demonstrado na Listagem 3.

```
aluno@pc ~ $ ifconfig eth0 down                                    1
aluno@pc ~ $ ifconfig eth0 up                                      2
aluno@pc ~ $ ifconfig eth0 1.1.1.1 netmask 255.255.255.0           3
```

Listing 3: Outras utilizações do ifconfig

1.2 ping

O comando **ping** permite testar a conectividade IP entre dois pontos de rede. Para o fazer, recorre ao protocolo ICMP (*Internet Message Control Protocol*) que é parte integrante do protocolo IP. Funciona enviando um pacote IP para o destino e aguarda por uma resposta, se esta resposta chegar é sinal que existe conectividade entre os dois pontos de rede.

Sem qualquer parâmetro, para além do IP ou nome da máquina destino, este comando tenta enviar pedidos continuamente de resposta. Em *Linux*, o comando *ping* só termina quando o utilizador pressiona CTRL+C. De seguida é apresentado um pequeno quadro estatístico de resumo. A Listagem 4 demonstra a sua utilização.

```
aluno@pc ~ $ ping www.cisco.com                                    1
PING origin-www.cisco.com (72.163.4.161) 56(84) bytes of data.     2
64 bytes from www1.cisco.com (72.163.4.161): icmp_req=1 ttl=106    3
    time=186 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_req=2 ttl=106    4
    time=177 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_req=3 ttl=106    5
    time=184 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_req=4 ttl=106    6
    time=186 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_req=5 ttl=106    7
    time=182 ms
64 bytes from www1.cisco.com (72.163.4.161): icmp_req=6 ttl=106    8
    time=186 ms
^C                                                                 9
--- origin-www.cisco.com ping statistics ---                       10
6 packets transmitted, 6 received, 0% packet loss, time 5006ms     11
rtt min/avg/max/mdev = 177.836/184.258/186.956/3.332 ms            12
```

Listing 4: Comando ping

1.3 tracepath

O comando *tracepath* permite obter uma lista detalhada dos equipamentos (tipicamente *routers*) por onde uma mensagem tem de passar para chegar ao

seu destino. A Listagem 5 demonstra a sua utilização. Realça-se em especial a utilização da opção **-n** que, ao impedir a resolução reversa de nomes, acelera o processo de apresentação da informação. Caso não se usa-se esta opção, para cada endereço IP apresentado, o comando tentaria descobrir o seu nome.

```
aluno@pc ~ $ tracepath -n www.slackware.com                          1
 1:  192.168.0.12                                          0.206     2
     ms pmtu 1500
 1:  82.102.48.1                                          60.376     3
     ms asymm  2
 1:  82.102.48.1                                          62.816     4
     ms asymm  2
 2:  82.102.34.1                                          63.822     5
     ms
 3:  82.102.32.1                                          71.224     6
     ms
 4:  213.248.99.9                                         93.584     7
     ms asymm  5
 5:  80.91.248.128                                        74.574     8
     ms asymm  6
 6:  80.91.251.98                                        213.562     9
     ms asymm  7
 7:  213.248.101.198                                     155.941     10
     ms
 8:  209.63.115.101                                      235.219     11
     ms asymm 17
 9:  209.63.82.89                                        234.215     12
     ms asymm 16
10:  209.63.82.98                                        238.220     13
    ms asymm 15
11:  209.63.82.166                                       244.468     14
    ms asymm 14
12:  209.63.82.14                                        236.566     15
    ms asymm 13
13:  209.63.114.169                                      231.821     16
    ms
14:  no reply                                                        17
15:  208.186.199.158                                     236.965     18
    ms asymm 12
16:  64.57.96.30                                         235.336     19
    ms asymm 13
17:  64.57.102.34                                        244.129     20
    ms reached
     Resume: pmtu 1500 hops 17 back 51                               21
```

Listing 5: Comando tracepath

1.4 Ferramentas de DNS

Os *Domain Name Services* (ou DNS) são serviços que permitem atribuir nomes a endereços IP. Tal facilita a utilização de serviços sem a necessidade de os utilizadores memorizarem endereços IP. Tipicamente, a generalidade dos utilizadores da Internet desconhece a existência de endereços IP, sabendo apenas os nomes dos serviços que utilizam (ex.: www.google.pt). Contrariamente, os equipamentos (PCs, servidores, *routers*, ...) apenas comunicam se conhecerem o endereço IP aonde se devem ligar. A sua importância levou então ao surgimento de várias ferramentas que facilitam a sua utilização, manutenção e administração.

1.4.1 host

O comando *host* permite obter tanto o endereço IP de um determinado nome, como o nome de um determinado endereço IP (ver Listagem 6

```
aluno@pc ~ $ host www.estgf.ipp.pt                                          1
www.estgf.ipp.pt has address 193.136.56.238                                 2
                                                                            3
aluno@pc ~ $ host 193.136.56.238                                            4
238.56.136.193.in-addr.arpa domain name pointer fw-priv.estgf.              5
    wan.ipp.pt.
                                                                            6
aluno@pc ~ $ host www.google.pt                                             7
www.google.pt is an alias for www-cctld.l.google.com.                       8
www-cctld.l.google.com has address 173.194.34.216                           9
www-cctld.l.google.com has IPv6 address 2a00:1450:4003:801::1018           10
```

Listing 6: Comando host

1.4.2 dig

O comando *dig* é um comando mais complexo que o comando *host*, no entanto permite obter informação de DNS muito mais completa. Em particular, o comando *host* apenas permite a consulta de registos do tipo A (endereço IP de um nome) ou CNAME (nomes alternativos para o mesmo endereço IP).

O serviço de DNS armazena outros tipos de registos, em particular os registos NS (que indicam quais são os servidores de nomes para um domínio) e os registos MX (que indicam os servidores de e-mail de um domínio) não podem ser obtidos com o comando anterior.

A Listagem 7 demonstra como utilizar o comando *dig* para obter o(s) endereço(s) do(s) servidor(es) de e-mail do domínio *estgf.ipp.pt*.

```
aluno@pc ~ $ dig estgf.ipp.pt mx

; <<>> DiG 9.7.3 <<>> estgf.ipp.pt mx
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15279
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
   ADDITIONAL: 2

;; QUESTION SECTION:
;estgf.ipp.pt.          IN   MX

;; ANSWER SECTION:
estgf.ipp.pt.    86400 IN   MX  5 mailserver01.core.ipp.pt.
estgf.ipp.pt.    86400 IN   MX  10 mailman.ipp.pt.

;; AUTHORITY SECTION:
estgf.ipp.pt.    10486 IN   NS  dns2.ipp.pt.
estgf.ipp.pt.    10486 IN   NS  dns1.ipp.pt.

;; ADDITIONAL SECTION:
dns1.ipp.pt.     3309  IN   A 193.136.56.10
dns2.ipp.pt.     3309  IN   A 193.136.56.9

;; Query time: 63 msec
;; SERVER: 192.168.0.254#53(192.168.0.254)
;; WHEN: Mon Feb 27 18:25:15 2012
;; MSG SIZE  rcvd: 158
```

Listing 7: Comando dig

1.5 telnet

O comando *telnet* permite duas grandes funcionalidades: 1) aceder a um terminal numa máquina remota; 2) abrir uma ligação TCP/IP para um serviço a correr numa máquina.

A primeira utilização referida anteriormente não é de todo recomendada já que o comando *telnet* não cifra os seus dados. Por outras palavras, toda a informação visualizada pelo utilizador pode ser capturada e lida por qualquer outro utilizador com ligação à mesma rede. Actualmente, para este fim, usa-se o SSH que é essencialmente um *telnet* com cifra de dados.

A segunda utilização é no entanto muito prática para testar se um determinado serviço está operacional e a responder correctamente. A Listagem 8 mostra como se pode utilizar o comando *telnet* para comunicar com um servidor de Web. Em particular, a linha n. 5 (*HEAD / HTTP/1.0*) foi

introduzida pelo utilizador. Já as restantes linhas, foram produzidas pelo servidor.

```
aluno@pc ~ $ telnet www2.estgf.ipp.pt 80
Trying 193.136.56.98...
Connected to www2.estgf.ipp.pt.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Mon, 27 Feb 2012 18:29:32 GMT
Server: Zope/(Zope 2.10.4-final, python 2.4.4, linux2) ZServer
    /1.1 Plone/3.1.7
Content-Length: 625
Accept-Ranges: none
Last-Modified: Mon, 27 Feb 2012 18:29:32 GMT
Content-Type: text/html; charset=ISO-8859-15
Connection: close

Connection closed by foreign host.
```

Listing 8: Comando telnet

1.6 netstat

O comando *netstat* permite a visualização de informação relativa às ligações de rede activas no PC, tabelas de rotas (caminhos), processos com ligações de rede, entre outras ...

A Listagem 9 demonstra como visualizar as ligações TCP/IP (opção **t**) activas no momento, bem como o seu estado (estabelecidas, em espera, a fechar).

```
aluno@pc ~ $ netstat -t
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address
          State
tcp        0      0 pc.local:59826      ww-in-f125.:xmpp-client
    ESTABLISHED
tcp        0      0 pc.local:41101      mad01s08-in-f21.1:https
    ESTABLISHED
tcp       38      0 pc.local:49315      v-client-2b.sjc.d:https
    CLOSE_WAIT
tcp       38      0 pc.local:46236      ec2-107-20-249-25:https
    CLOSE_WAIT
tcp       38      0 pc.local:53139      ec2-23-21-220-16.:https
    CLOSE_WAIT
```

```
tcp        0      0 pc.local:54215          mad01s08-in-f22.1:https   9
    ESTABLISHED
tcp        0      0 pc.local:34705          ec2-204-236-220-1:https   10
    ESTABLISHED
tcp        0      0 pc.local:36517          mad01s03-in-f30.1:https   11
    TIME_WAIT
tcp        0      0 pc.local:40804          sjc-not7.sjc.dropbo:www   12
    ESTABLISHED
tcp        0      0 pc.local:35987          v-client-3a.sjc.d:https   13
    ESTABLISHED
tcp       38      0 pc.local:54160          v-d-1b.sjc.dropbo:https   14
    CLOSE_WAIT
tcp        0      0 pc.local:56857          baymsg1020116.gate:msnp   15
    ESTABLISHED
```

Listing 9: Comando netstat

O comando pode ainda ser utilizado para visualizar a tabelas de rotas (opção **r**) em uso (ver Listagem 10).

```
aluno@pc ~ $ netstat -rn                                               1
Kernel IP routing table                                                2
Destination     Gateway         Genmask         Flags    MSS          3
    Window  irtt Iface
0.0.0.0         192.168.0.254   0.0.0.0         UG         0 0        4
             0 wlan0
169.254.0.0     0.0.0.0         255.255.0.0     U          0 0        5
             0 wlan0
192.168.0.0     0.0.0.0         255.255.255.0   U          0 0        6
             0 wlan0
```

Listing 10: Comando netstat

Ou ainda, para visualizar os processos em execução que estão à espera de ligações de outras máquinas. A Listagem 11 mostra os processos (opção **p**) à espera de ligações (opção **l**) TCP/IP (opção **t**).

```
aluno@pc ~ $ netstat -ltp                                              1
Active Internet connections (only servers)                             2
Proto Recv-Q Send-Q Local Address           Foreign Address            3
            State       PID/Program name
tcp        0      0 *:50194                 *:*                        4
                     LISTEN      1897/pidgin
tcp        0      0 *:ssh                   *:*                        5
                     LISTEN      672/sshd
tcp        0      0 localhost:ipp           *:*                        6
                     LISTEN      891/cupsd
tcp        0      0 *:17500                 *:*                        7
                     LISTEN      1653/dropbox
```

```
tcp6       0      0 [::]:ssh                [::]:*                  8
                    LISTEN      672/sshd
tcp6       0      0 ip6-localhost:ipp       [::]:*                  9
                    LISTEN      891/cupsd
```

Listing 11: Comando netstat