

常用网络工具介绍

三个常用的命令

- telnet
- nc
- netstat

命令一：telnet

现在 telnet server 几乎没有人用了，但是 telnet client 却被广泛的使用着。它的功能已经比较强大，有较多巧妙的用法

检查端口是否打开

telnet 的一个最大作用就是检查一个端口是否处于打开，使用的命令是 `telnet [domainname or ip] [port]`，这条命令能告诉我们到远端 server 指定端口的网连接是否可达。

```
telnet [domainname or ip] [port]
```

telnet 第一个参数是要连接的域名或者 ip，第二个参数是要连接的端口

如果这个网络连接可达，则会提示你Connected to 220.181.57.216，输入control]可以给这个端口发送数据包了

```
➔ ~ telnet 220.181.57.216 80
Trying 220.181.57.216...
Connected to 220.181.57.216.
Escape character is '^]'.
^]
telnet> █
```

如果网络不可达，则会提示telnet: Unable to connect to remote host和具体不能连上的原因，常见的有 Operation timed out、Connection refused。

比如我本机没有进程监听 90 端口，telnet 127.0.0.1 90的信息如下

```
➔ ~ telnet 127.0.0.1 90
Trying 127.0.0.1...
telnet: connect to address 127.0.0.1: Connection refused
telnet: Unable to connect to remote host
```

telnet 还能发 http 请求?

执行 `telnet www.baidu.com 80`, 粘贴下面的文本 (注意总共有四行, 最后两行为两个空行)

```
GET / HTTP/1.1
```

```
Host: www.baidu.com
```

telnet 还可以连接 Redis

假设 redis 服务器跑在本地，监听 6379 端口，用 `telnet 6379` 命令可以连接上。接下来就可以调用 redis 的命令。

调用 `"set hello world"`，给 key 为 hello 设置值为 "world"，随后调用 `get hello` 获取值

Redis 客户端和 Redis 服务器使用 RESP 协议通信，RESP 是 REdis Serialization Protocol 的简称。在 RESP 中，通过检查服务器返回数据的第一个字节来确定这个回复是什么类型：

- 对于 Simple Strings 来说，第一个字节是 "+"
- 对于 Errors 来说，第一个字节是 "-"
- 对于 Integers 来说，第一个字节是 ":"
- 对于 Bulk Strings 来说，首字节是 "\$"
- 对于 Arrays 来说，首字节是 "*"

RESP Simple Strings

Simple Strings 被用来传输非二进制安全的字符串，是按下面的方式进行编码：一个加号，紧接着是不包含 CR 或者 LF 的字符串(不允许换行)，最后以CRLF("\r\n")结尾。

执行 "set hello world" 命令成功，服务器会响应一个 "OK"，这是 RESP 一种 Simple Strings 的场景，这种情况下，OK 被编码为五个字节： +OK\r\n

RESP Bulk Strings

get 命令读取 hello 的值，redis 服务器返回 `$5\r\nworld\r\n`，这种类型属于是 Bulk Strings 被用来表示二进制安全的字符串。

Bulk Strings 的编码方式是下面这种方式：以 "\$" 开头，后跟实际要发送的字节数，随后是 CRLF，然后是实际的字符串数据，最后以 CRLF 结束。

所以 "world" 这个 string 会被编码成这样：`$5\r\nworld\r\n`

命令二： netcat

netcat 因为功能强大，被称为网络工具中的瑞士军刀，nc 是 netstat 的简称。这篇文章将介绍 nc 常用的几个场景。

用 nc 来当聊天服务器

当然，真正在现实场景中用 nc 来聊天用的非常少。nc -l 命令一个有价值的地方是可以快速的启动一个 tcp server 监听某个端口。

发送 http 请求

来回忆一下 HTTP 请求报文的组成：

1. 起始行 (start line)
2. 首部 (header)
3. 可选的内容主体 (body)

输入 "nc www.baidu.com 80", 然后输入enter,

```
nc www.baidu.com 80
```

```
<enter>
```

```
<enter>
```

除了狂按 enter，你也可以采用 unix 管道的方式，把 HTTP 请求报文传输过去

```
echo -ne "GET / HTTP/1.1\r\nhost:www.baidu.com\r\n\r\n" | nc www.baidu.com 80
```

echo 的 -n 参数很关键，echo 默认会在输出的最后增加一个换行，加上 -n 参数以后就不会在最后自动换行了。

执行上面的命令，可以看到也返回了百度的首页 html

查看远程端口是否打开

前面介绍过 telnet 命令也可以检查远程端口是否打开，既然 nc 被称为瑞士军刀，这个小功能不能说不行。

```
nc -zv [host or ip] [port]
```

其中 -z 参数表示不发送任何数据包，tcp 三次握手完后自动退出进程。有了 -v 参数则会输出更多详细信息（verbose）。

访问 redis

nc 为 在没有 redis-cli 的情况下访问 redis 又新增了一种方法

```
nc localhost 6379
```

```
ping
```

```
+PONG
```

```
get hello
```

```
$5
```

```
world
```

同样可以把命令通过管道的方式传给 redis 服务器。

```
echo ping | nc localhost 6379
```

```
+PONG
```

netstat

netstat 很强大的网络工具，可以用来显示套接字的状态。下面来介绍一下常用的命令选项

列出所有套接字

netstat -a

-a命令可以输出所有的套接字，包括监听的和未监听的套接字。

```
ya@c2 ~/dev/packetdrill_scripts$ netstat -a
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:webcache         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:domain          0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:postgres         0.0.0.0:*               LISTEN
tcp        0      0 localhost:smtp          0.0.0.0:*               LISTEN
tcp        0      0 c2.shared:ssh           test.ya.local:61594     ESTABLISHED
tcp        0      0 c2.shared:ssh           test.ya.local:56133     ESTABLISHED
tcp6       0      0 [::]:webcache           [::]:*                  LISTEN
tcp6       0      0 [::]:domain              [::]:*                  LISTEN
tcp6       0      0 [::]:ssh                 [::]:*                  LISTEN
tcp6       0      0 [::]:postgres            [::]:*                  LISTEN
tcp6       0      0 localhost:smtp          [::]:*                  LISTEN
udp        0      0 c2.shared:33028         ntp5.flashdance.cx:ntp ESTABLISHED
udp        0      0 localhost:323            0.0.0.0:*               ESTABLISHED
udp        0      0 c2.shared:46607         119.28.206.193:ntp      ESTABLISHED
udp        0      0 c2.shared:54965         ntp1.ams1.nl.leasew:ntp ESTABLISHED
udp        0      0 0.0.0.0:26586           0.0.0.0:*               ESTABLISHED
udp        0      0 0.0.0.0:domain          0.0.0.0:*               ESTABLISHED
udp        0      0 0.0.0.0:bootpc          0.0.0.0:*               ESTABLISHED
udp6       0      0 localhost:323           [::]:*                  ESTABLISHED
udp6       0      0 [::]:52625              [::]:*                  ESTABLISHED
udp6       0      0 localhost:38716         localhost:38716         ESTABLISHED
udp6       0      0 [::]:domain              [::]:*                  ESTABLISHED
raw6       0      0 [::]:ipv6-icmp          [::]:*                  7
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State       I-Node  Path
unix  2      [ ACC ] STREAM    LISTENING   12585    /var/run/dbus/system_bus_s
ket
```

只列出 TCP 套接字

`netstat -at`

`-t` 选项可以只列出 TCP 的套接字，也可也用 `--tcp`

```
ya@c2 ~/dev/packetdrill_scripts$ netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:webcache         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:domain          0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:postgres          0.0.0.0:*               LISTEN
tcp        0      0 localhost:smtp          0.0.0.0:*               LISTEN
tcp        0      0 c2.shared:ssh           test.ya.local:61594     ESTABLISHED
tcp        0      0 c2.shared:ssh           test.ya.local:56133     ESTABLISHED
tcp6       0      0 [::]:webcache           [::]:*                 LISTEN
tcp6       0      0 [::]:domain             [::]:*                 LISTEN
tcp6       0      0 [::]:ssh                 [::]:*                 LISTEN
tcp6       0      0 [::]:postgres           [::]:*                 LISTEN
tcp6       0      0 localhost:smtp          [::]:*                 LISTEN
```

只列出 UDP 连接

`netstat -au`

`-u` 选项用来指定显示 UDP 的连接，也可也用 `--udp`

示例输出：

```
ya@c2 ~/dev/packetdrill_scripts$ netstat -au
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 c2.shared:44274         ntp1.ams1.nl.leasew:ntp ESTABLISHED
udp        0      0 localhost:323           0.0.0.0:*
udp        0      0 c2.shared:53693         ntp5.flashdance.cx:ntp ESTABLISHED
udp        0      0 0.0.0.0:26586           0.0.0.0:*
udp        0      0 0.0.0.0:domain          0.0.0.0:*
udp        0      0 0.0.0.0:bootpc           0.0.0.0:*
udp6       0      0 localhost:323           [::]:*
udp6       0      0 [::]:52625              [::]:*
udp6       0      0 localhost:38716         localhost:38716         ESTABLISHED
udp6       0      0 [::]:domain              [::]:*
```

只列出处于监听状态的连接

netstat -l

-l 选项用来指定处于 LISTEN 状态的连接，也可以用--listening
示例输出：

```
ya@c2 ~/dev/packetdrill_scripts$ netstat -l
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 0.0.0.0:webcache        0.0.0.0:*               LISTEN
tcp      0      0 0.0.0.0:domain         0.0.0.0:*               LISTEN
tcp      0      0 0.0.0.0:ssh            0.0.0.0:*               LISTEN
tcp      0      0 0.0.0.0:postgres       0.0.0.0:*               LISTEN
tcp      0      0 localhost:smtp         0.0.0.0:*               LISTEN
tcp6     0      0 [::]:webcache         [::]:*                  LISTEN
tcp6     0      0 [::]:domain           [::]:*                  LISTEN
tcp6     0      0 [::]:ssh              [::]:*                  LISTEN
tcp6     0      0 [::]:postgres         [::]:*                  LISTEN
tcp6     0      0 localhost:smtp        [::]:*                  LISTEN
udp      0      0 localhost:323          0.0.0.0:*
udp      0      0 0.0.0.0:26586         0.0.0.0:*
udp      0      0 0.0.0.0:domain        0.0.0.0:*
udp      0      0 0.0.0.0:bootpc        0.0.0.0:*
udp6     0      0 localhost:323         [::]:*
udp6     0      0 [::]:52625            [::]:*
udp6     0      0 [::]:domain           [::]:*
raw6     0      0 [::]:ipv6-icmp        [::]:*                  7
Active UNIX domain sockets (only servers)
Proto RefCnt Flags      Type       State      I-Node  Path
ket
unix  2      [ ACC ]   STREAM    LISTENING   12585   /var/run/dbus/system_bus_
ket
unix  2      [ ACC ]   STREAM    LISTENING   10027   /run/systemd/private
unix  2      [ ACC ]   STREAM    LISTENING   17241   public/cleanup
```


与-a一样，可以组合-t来过滤处于 listen 状态的 TCP 连接

```
netstat -lt
```

示例输出


```
ya@c2 ~/dev/packetdrill_scripts$ netstat -lt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:webcache         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:domain          0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:postgres          0.0.0.0:*               LISTEN
tcp        0      0 localhost:smtp          0.0.0.0:*               LISTEN
tcp6       0      0 [::]:webcache           [::]:*                  LISTEN
tcp6       0      0 [::]:domain             [::]:*                  LISTEN
tcp6       0      0 [::]:ssh                 [::]:*                  LISTEN
tcp6       0      0 [::]:postgres           [::]:*                  LISTEN
tcp6       0      0 localhost:smtp          [::]:*                  LISTEN
```

禁用端口 和 IP 映射

netstat -ltn

上面的例子中，常用端口都被映射为了名字，比如 22 端口输出显示为 ssh，8080 端口被映射为 webcache。大部分情况下，我们并不想 netstat 帮我们做这样的事情，可以加上 -n 禁用

```
ya@c2 ~/dev/packetdrill_scripts$ netstat -lt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:webcache         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:domain          0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:postgres         0.0.0.0:*               LISTEN
tcp        0      0 localhost:smtp          0.0.0.0:*               LISTEN
tcp6       0      0 [::]:webcache           [::]:*                  LISTEN
tcp6       0      0 [::]:domain             [::]:*                  LISTEN
tcp6       0      0 [::]:ssh                 [::]:*                  LISTEN
tcp6       0      0 [::]:postgres            [::]:*                  LISTEN
tcp6       0      0 localhost:smtp          [::]:*                  LISTEN
```



```
ya@c2 ~/dev/packetdrill_scripts$ netstat -ltn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:8080            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:53              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:5432            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp6       0      0 :::8080                 :::*                     LISTEN
tcp6       0      0 :::53                   :::*                     LISTEN
tcp6       0      0 :::22                   :::*                     LISTEN
tcp6       0      0 :::5432                  :::*                     LISTEN
tcp6       0      0 :::1:25                  :::*                     LISTEN
```

显示进程

```
netstat -ltnp
```

使用 -p命令可以显示连接归属的进程信息，在查看端口被哪个进程占用时非常有用
示例输出如下：

```
ya@c2 ~/dev/packetdrill_scripts$ sudo netstat -ltnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:8080             0.0.0.0:*               LISTEN      24972/nc
tcp        0      0 0.0.0.0:53               0.0.0.0:*               LISTEN      1287/dnsmasq
tcp        0      0 0.0.0.0:22               0.0.0.0:*               LISTEN      1285/sshd
tcp        0      0 0.0.0.0:5432             0.0.0.0:*               LISTEN      1300/postmaster
tcp        0      0 127.0.0.1:25             0.0.0.0:*               LISTEN      1675/master
tcp6       0      0 :::8080                  :::*                   LISTEN      24972/nc
tcp6       0      0 :::53                    :::*                   LISTEN      1287/dnsmasq
tcp6       0      0 :::22                    :::*                   LISTEN      1285/sshd
tcp6       0      0 :::5432                   :::*                   LISTEN      1300/postmaster
tcp6       0      0 :::1:25                   :::*                   LISTEN      1675/master
```

显示所有的网卡信息

`netstat -i`

用 `-i` 命令可以列出网卡信息，比如 MTU 等

示例输出

```
ya@c2 ~/dev/packetdrill_scripts$ netstat -i
Kernel Interface table
Iface      MTU      RX-OK RX-ERR RX-DRP RX-OVR      TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0       1500    3063358      0      0 0      1690544      0      0      0 BMRU
lo         65536    17803      0      0 0      17803      0      0      0 LRU
```

显示 8080 端口所有处于 ESTABLISHED 状态的连接

```
netstat -atnp | grep ":8080" | grep ESTABLISHED
```

```
tcp          0      0 10.211.55.10:8080      10.211.55.5:45438      ESTABLISHED 24972/nc
```

统计处于各个状态的连接个数

```
netstat -ant | awk '{print $6}' | sort | uniq -c | sort -n  
1 established)  
1 Foreign  
2 LISTEN  
3 TIME_WAIT  
30 ESTABLISHED
```

使用 awk 截取出状态行，然后用 sort、uniq 进行去重和计数即可

