**1 ss命令**

用来**显示处于活动状态的套接字信息**，ss命令可以用来获取**socket统计信息**，它可以提供的信息如下：

1） 所有的TCP sockets；

2） 所有的UDP sockets；

3） 所有ssh/ftp/ttp/https持久连接；

4） 所有连接到Xserver的本地进程；

5） **使用state（例如：connected, synchronized, SYN-RECV, SYN-SENT,TIME-WAIT）、地址、端口过滤**；

6） 所有的state FIN-WAIT-1 tcpsocket连接以及更多；

当服务器的socket连接数量变得非常大时，无论是使用netstat命令还是直接**cat /proc/net/tcp**，执行速度都会很慢。ss快的秘诀**在于它利用到了TCP协议栈中tcp\_diag**，tcp\_diag是一个用于分析统计的模块，可以获得Linux 内核中第一手的信息，这就确保了ss的快捷高效。当然，如果你的系统中没有tcp\_diag，ss也可以正常运行，只是效率会变得稍慢。

**2 和netstat对比**

统计服务器并发连接数：

**# time netstat -ant | grep EST | wc -l**

**3100**

**real 0m12.960s**

**user 0m0.334s**

**sys 0m12.561s**

**# time ss -o state established | wc -l**

**3204**

**real 0m0.030s**

**user 0m0.005s**

**sys 0m0.026s**

很明显ss效率远高于netstat**；**

**3 选项**

-h：显示帮助信息；

-V：显示指令版本信息；

-n：不解析服务名称，以数字方式显示；

-a：显示所有的套接字；

-l：**显示处于监听状态的套接字**；

-o：显示计时器信息；

-m：显示套接字的内存使用情况；

-p：**显示使用套接字的进程信息**；

-i： **显示内部的TCP信息**；

-4：只显示ipv4的套接字；

-6：只显示ipv6的套接字；

-t：**只显示tcp套接字**；

-u：只显示udp套接字；

-d：只显示DCCP套接字；

-w：仅显示RAW套接字；

-x：仅显示UNIX域套接字。

**4 实例**

**4.1 显示每个进程具体打开的socket**

**ss -pl**

**4.2 显示所有已建立的SMTP连接**

**ss -o state established '( dport = :smtp or sport = :smtp )'**

**4.3 显示所有已建立的HTTP连接**

**ss -o state established '( dport = :http or sport = :http )'**

**4.4 显示 Sockets 摘要**

**ss -s**

**4.5 列出所有FIN-WAIT-1状态的http，https连接**

**ss -o state fin-wait-1 '( sport = :http or sport = :https )'**

**4.6 找出打开套接字/端口应用程序**

**ss -pl | grep 3306**

**5 ss常用的state状态：**

established

syn-sent

syn-recv

fin-wait-1

fin-wait-2

time-wait

closed

close-wait

last-ack

listen

closing

all : All of the above states

connected : All the states except for listen and closed

synchronized : All the connected states except for syn-sent

bucket : Show states, which are maintained as minisockets, i.e. time-wait and syn-recv.

big : Opposite to bucket state.

**6 ss使用IP地址筛选**

**ss src ADDRESS\_PATTERN**

src：表示来源

ADDRESS\_PATTERN：表示地址规则，如下：

**# 列出来自120.33.31.1的连接**

**ss src 120.33.31.1**

**＃　列出来自120.33.31.1,80端口的连接**

ss src 120.33.31.1:http

ss src 120.33.31.1:80

**7 ss使用端口筛选**

**ss dport OP PORT**

OP:是运算符

PORT：表示端口

dport：表示过滤目标端口、相反的有sport

**OP运算符如下：**

<= or le : 小于等于 >= or ge : 大于等于

== or eq : 等于

!= or ne : 不等于端口

< or lt : 小于这个端口 > or gt : 大于端口

**OP实例：**

ss sport = :http 也可以是 ss sport = :80

ss dport = :http

ss dport \> :1024

ss sport \> :1024

ss sport \< :32000

ss sport eq :22

ss dport != :22

ss state connected sport = :http

ss \( sport = :http or sport = :https \)

ss -o state fin-wait-1 \( sport = :http or sport = :https \) dst 192.168.1/24