链滴

Linux 服务器端口监听

作者：henryspace

原文链接：https://ld246.com/article/1666767846937

来源网站：链滴

列出Linux中的所有开放端口

sudo ss -tulpn

```
Netid     State      Recv-Q     Send-Q         Local Address:Port         Peer Address:Port    Process
udp       UNCONN     0          0                  127.0.0.1:323               0.0.0.0:*        users:(("chronyd",pid=736,fd=5))
udp       UNCONN     0          0                    0.0.0.0:4500              0.0.0.0:*        users:(("docker-proxy",pid=141694,fd=4))
udp       UNCONN     0          0                    0.0.0.0:500               0.0.0.0:*        users:(("docker-proxy",pid=141710,fd=4))
udp       UNCONN     0          0                      [::1]:323                  [::]:*        users:(("chronyd",pid=736,fd=6))
udp       UNCONN     0          0                       [::]:4500                 [::]:*        users:(("docker-proxy",pid=141698,fd=4))
udp       UNCONN     0          0                       [::]:500                  [::]:*        users:(("docker-proxy",pid=141714,fd=4))
tcp       LISTEN     0          128                  0.0.0.0:22                0.0.0.0:*        users:(("sshd",pid=1133,fd=5))
tcp       LISTEN     0          128                     [::]:22                   [::]:*        users:(("sshd",pid=1133,fd=7))
```

以上命令的输出中，“State”列显示端口是否处于侦听状态（LISTEN）。

在上面的命令中，参数标志：

- -t – 启用TCP端口列表。
- -u – 启用UDP端口列表。
- -l – 仅打印监听套接字。
- -n – 显示端口号。
- -p – 显示进程/程序名称。

实时观察TCP和UDP开放端口

sudo watch ss -tulpn

```
Every 2.0s: ss -tulpn                                                    ip-172-31-31-93.ec2.internal: Wed Oct

Netid State  Recv-Q Send-Q Local Address:Port Peer Address:PortProcess
udp   UNCONN 0      0          127.0.0.1:323       0.0.0.0:*    users:(("chronyd",pid=736,fd=5))
udp   UNCONN 0      0            0.0.0.0:4500      0.0.0.0:*    users:(("docker-proxy",pid=141694,fd=4))
udp   UNCONN 0      0            0.0.0.0:500       0.0.0.0:*    users:(("docker-proxy",pid=141710,fd=4))
udp   UNCONN 0      0              [::1]:323          [::]:*    users:(("chronyd",pid=736,fd=6))
udp   UNCONN 0      0               [::]:4500         [::]:*    users:(("docker-proxy",pid=141698,fd=4))
udp   UNCONN 0      0               [::]:500          [::]:*    users:(("docker-proxy",pid=141714,fd=4))
tcp   LISTEN 0      128          0.0.0.0:22        0.0.0.0:*    users:(("sshd",pid=1133,fd=5))
tcp   LISTEN 0      128             [::]:22           [::]:*    users:(("sshd",pid=1133,fd=7))
```

原文链接：Linux 服务器端口监听

客户端测试远程端口

```
# 测试目标端口是否正常
nc -vuz <host_ip> 1194

# 向UDP端口发送消息
nc -vu <host_ip> 1194

# 通过抓包命令进行抓包，接收UDP端口抓包
tcpdump -i eth0 -s 0 port 1194
```

nc命令参数介绍：

```
OpenBSD netcat（Debian 补丁级别 1.187-1ubuntu0.1）
用法：nc       [-46CDdFhklNnrStUuvZz] [-I 长度] [-i 间隔] [-M ttl]
                  [-m minttl] [-O 长度] [-P proxy_username] [-p source_port]
          [-q 秒] [-s 源] [-T 关键字] [-V rtable] [-W recvlimit] [-w 超时]
          [-X proxy_protocol] [-x proxy_address[:port]] [destination] [port]
      命令摘要：
            -4 使用 IPv4
            -6 使用 IPv6
            -b 允许广播
            -C 发送 CRLF 作为行尾
            -D 启用调试套接字选项
            -d 与标准输入分离
            -F 传递套接字 fd
            -h 这个帮助文本
            -I length TCP 接收缓冲区长度
            -i 间隔 发送线路的延迟间隔，扫描的端口
            -k 保持入站套接字为多个连接打开
            -l 监听模式，用于入站连接
            -M ttl 传出 TTL / 跳数限制
            -m minttl 最小传入 TTL / 跳数限制
            -N 在标准输入 EOF 后关闭网络套接字
            -n 禁止名称/端口解析
            -O length TCP 发送缓冲区长度
            -P proxyuser 代理认证的用户名
            -p port 指定远程连接的本地端口
            -q secs 在标准输入 EOF 后退出并延迟 secs
            -r 随机化远程端口
            -S 启用 TCP MD5 签名选项
            -s source 本地源地址
            -T 关键字 TOS 值
            -t 应答 TELNET 协商
            -U 使用 UNIX 域套接字
            -u UDP 模式
            -V rtable 指定备用路由表
            -v 详细
            -W recvlimit 收到一定数量的数据包后终止
            -w timeout 连接和最终网络读取超时
            -X proto 代理协议：“4”、“5”（SOCKS）或“connect”
            -x addr[:port] 指定代理地址和端口
            -Z DCCP 模式
            -z 零 I/O 模式 [用于扫描]
```

原文链接：Linux 服务器端口监听

端口号可以是单个或范围：lo-hi [含]