

# iperf3 使用

- 1: wget <https://iperf.fr/download/source/iperf-3.1.3-source.tar.gz>
- 2: tar xzvf iperf-3.1.3-source.tar.gz
- 3: 设定编译器, 和prefix (默认使用8.2.1, 如使用其他编译器, 此处指定)

```
./configure --host=arm-linux-gnueabi \
CC=arm-linux-gnueabi-gcc \
CXX=arm-linux-gnueabi-g++ \
RANLIB=arm-linux-gnueabi-ranlib \
STRIP=arm-linux-gnueabi-strip \
ac_cv_func_malloc_0_nonnull=yes \
CFLAGS=-static \
CXXFLAGS=-static \
--prefix=/Home/jackson.pan/opensource/iperf-3.1.3/out
```

4: make && make install

5: 将out中的iperf3 拷贝到target board

测试范例:

```
./iperf3 -s
```

```
-----
Server listening on 5201
-----
```

结果:

```
Accepted connection from 172.19.30.175, port 57290
[ 5] local 172.19.24.242 port 5201 connected to 172.19.30.175 port 50363
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[ 5] 0.00-1.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 1.00-2.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 2.00-3.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 3.00-4.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 4.00-5.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 5.00-6.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 6.00-7.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 7.00-8.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 8.00-9.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
[ 5] 9.00-10.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
-----
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[ 5] 0.00-10.00 sec 0.00 Bytes 0.00 bits/sec 0.000 ms 0/0 (0%)
-----
```

```
Server listening on 5201
-----
```

远端:

```
iperf3 -c 172.19.24.242 -u -i 1 -t 10
Connecting to host 172.19.24.242, port 5201
[ 4] local 172.19.30.175 port 50363 connected to 172.19.24.242 port 5201
[ ID] Interval Transfer Bandwidth Total Datagrams
[ 4] 0.00-1.00 sec 120 KBytes 983 Kbits/sec 15
[ 4] 1.00-2.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 2.00-3.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 3.00-4.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 4.00-5.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 5.00-6.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 6.00-7.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 7.00-8.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 8.00-9.00 sec 128 KBytes 1.05 Mbits/sec 16
[ 4] 9.00-10.00 sec 128 KBytes 1.05 Mbits/sec 16
-----
[ ID] Interval Transfer Bandwidth Jitter Lost/Total Datagrams
[ 4] 0.00-10.00 sec 1.24 MBytes 1.04 Mbits/sec 0.000 ms 0/0 (-nan%)
[ 4] Sent 0 datagrams
```

iperf Done.

参数:

-p, --port #, Server 端监听、Client 端连接的端口号;  
-f, --format [kmgKMG], 报告中所用的数据单位, Kbits, Mbits, KBytes, Mbytes;  
-i, --interval #, 每次报告的间隔, 单位为秒;  
-F, --file name, 测试所用文件的文件名。如果用在 Client 端, 发送该文件用作测试; 如果用在 Server 端, 则是将数据写入该文件, 而不是丢弃;  
-A, --affinity n/n,m, 设置 CPU 亲和性;  
-B, --bind, 绑定指定的网卡接口;  
-V, --verbose, 运行时输出更多细节;  
-J, --json, 运行时以 JSON 格式输出结果;  
--logfile f, 输出到文件;  
-d, --debug, 以 debug 模式输出结果;  
-v, --version, 显示版本信息并退出;  
-h, --help, 显示帮助信息并退出。

Server 端参数:

- s, --server, 以 Server 模式运行;
- D, --daemon, 在后台以守护进程运行;
- I, --pidfile file, 指定 pid 文件;
- l, --one-off, 只接受 1 次来自 Client 端的测试, 然后退出。

Client 端参数

- c, --client, 以 Client 模式运行, 并指定 Server 端的地址;
- u, --udp, 以 UDP 协议进行测试;
- b, --bandwidth #[KMG][/#], 限制测试带宽。UDP 默认为 1Mbit/秒, TCP 默认无限制;
- t, --time #, 以时间为测试结束条件进行测试, 默认为 10 秒;
- n, --bytes #[KMG], 以数据传输大小为测试结束条件进行测试;
- k, --blockcount #[KMG], 以传输数据包数量为测试结束条件进行测试;
- l, --len #[KMG], 读写缓冲区的长度, TCP 默认为 128K, UDP 默认为 8K;
- cport, 指定 Client 端运行所使用的 TCP 或 UDP 端口, 默认为临时端口;
- P, --parallel #, 测试数据流并发数量;
- R, --reverse, 反向模式运行 (Server 端发送, Client 端接收);
- w, --window #[KMG], 设置套接字缓冲区大小, TCP 模式下为窗口大小;
- C, --congestion, 设置 TCP 拥塞控制算法 (仅支持 Linux 和 FreeBSD);
- M, --set-mss #, 设置 TCP/SCTP 最大分段长度 (MSS, MTU 减 40 字节);
- N, --no-delay, 设置 TCP/SCTP no delay, 屏蔽 Nagle 算法;
- 4, --version4, 仅使用 IPv4;
- 6, --version6, 仅使用 IPv6;
- S, --tos N, 设置 IP 服务类型 (TOS, Type Of Service);
- L, --flowlabel N, 设置 IPv6 流标签 (仅支持 Linux);
- Z, --zerocopy, 使用 “zero copy” (零拷贝) 方法发送数据;
- O, --omit N, 忽略前 n 秒的测试;
- T, --title str, 设置每行测试结果的前缀;
- get-server-output, 从 Server 端获取测试结果;
- udp-counters-64bit, 在 UDP 测试包中使用 64 位计数器 (防止计数器溢出)。