

Openstack 之 SDN 性能测试具体实施及结果

周威光整理^{*}

2017-07-21

^{*} 简介：恒天云 FTE

目 录

1	网络性能测试指标	3
2	网络性能测试工具选择	3
2.1	常见的网络性能测试工具简介	3
2.2	测试方法确定	3
3	网络性能测试工具 iperf 的使用	4
3.1	功能简介	4
3.1.1	TCP	4
3.1.2	UDP	4
3.1.3	其他	4
3.2	使用原理	5
3.3	iperf 使用方法与参数说明	5
3.4	tcp 包测试带宽示范	5
3.5	udp 包同时测试抖动和丢包示范	6
4	ping 命令的使用	7
4.1	ping 使用方法与参数说明	7
4.2	icmp 包测试延迟示范	8
5	测试用例设计	8
5.1	带宽测试用例设计	8
5.2	延迟测试用例设计	8
5.3	抖动测试用例设计	8
5.4	丢包测试用例设计	8

1 网络性能测试指标

常见的网络性能测试指标包含：网络带宽 (bandwidth)、网络延迟 (latency)、抖动 (jitter)、丢包率等

1. 网络带宽：网络带宽是指在一个固定的时间内（1 秒），能通过的最大位数据。
2. 网络延迟：通俗的讲，就是数据从电脑这边传到那边所用的时间。
3. 抖动：用于描述包在网络中的传输延时的变化，抖动越小，说明网络质量越稳定越好。抖动是评价一个网络性能的最重要的因素。
4. 丢包率：测试中所丢失的数据包数量占所发送的数据包的比率。

2 网络性能测试工具选择

2.1 常见的网络性能测试工具简介

常用的开源网络性能测试工具有两个：iperf 和 netperf，iperf 是美国伊利诺斯大学 (University of Illinois) 开发的一种开源的网络性能测试工具，netperf 是由惠普公司开发的一种网络性能的测量工具，测试网络栈。这两种工具都可以测试 TCP 协议和 UDP 协议，从可测试的网络性能指标，我们对两种工具进行下对比：

工具	带宽	网络延迟	抖动	丢包
iperf	是	否	是	是
netperf	是	是	是	否

可见 iperf 和 netperf 都可以完成基本的网络性能测试，但 netperf 更倾向于测试不同网络模式的数据传输，与本次性能测试需求不符。而 iperf 可以经过简单的参数设置，比较直观的给出带宽、抖动和丢包，并且还能设置测试时间、发送包的大小，以及带宽。虽然无法对网络延迟进行统一测试，但可以使用 ping 进行弥补

2.2 测试方法确定

测试对象	带宽	延时	抖动 (Jitter)	丢包
测试工具	iperf TCP 包测试	ping 测试	iperf UDP 测试	iperf UDP 测试

3 网络性能测试工具 iperf 的使用

3.1 功能简介

Iperf 的主要功能如下：

3.1.1 TCP

1. 测量网络带宽
2. 报告 MSS/MTU 值的大小和观测值
3. 支持 TCP 窗口值通过套接字缓冲
4. 当 P 线程或 Win32 线程可用时，支持多线程。客户端与服务端支持同时多重连接

3.1.2 UDP

1. 客户端可以创建指定带宽的 UDP 流
2. 测量丢包
3. 测量延迟
4. 支持多播
5. 当 P 线程可用时，支持多线程。客户端与服务端支持同时多重连接（不支持 Windows）

3.1.3 其他

1. 在适当的地方，选项中可以使用 K (kilo-) 和 M (mega-)。例如 131072 字节可以用 128K 代替。
2. 可以指定运行的总时间，甚至可以设置传输的数据总量。
3. 在报告中，为数据选用最合适的单位。
4. 服务器支持多重连接，而不是等待一个单线程测试。
5. 在指定时间间隔重复显示网络带宽，波动和丢包情况。
6. 服务器端可作为后台程序运行。
7. 服务器端可作为 Windows 服务运行。

8. 使用典型数据流来测试链接层压缩对于可用带宽的影响。
9. 支持传送指定文件，可以定性和定量测试

3.2 使用原理

使用 Iperf 测试时必须将一台主机设置为客户端，一台主机设置为服务器。

3.3 iperf 使用方法与参数说明

```

1  参数说明
2  -s 以server模式启动，eg: iperf -s
3  -c host以client模式启动，host是server端地址，eg: iperf -c 222.35.11.23
4
5  通用参数
6  -f [kmKM] 分别表示以Kbits, Mbits, KBytes, MBytes显示报告，默认以Mbits为单位, eg:
   iperf -c 222.35.11.23 -f K
7  -i sec 以秒为单位显示报告间隔，eg: iperf -c 222.35.11.23 -i 2
8  -l 缓冲区大小，默认是8KB, eg: iperf -c 222.35.11.23 -l 16
9  -m 显示tcp最大mtu值
10 -o 将报告和错误信息输出到文件eg: iperf -c 222.35.11.23 -o ciperflog.txt
11 -p 指定服务器端使用的端口或客户端所连接的端口eg: iperf -s -p 9999; iperf -c
    222.35.11.23 -p 9999
12 -u 使用udp协议
13 -w 指定TCP窗口大小，默认是8KB
14 -B 绑定一个主机地址或接口（当主机有多个地址或接口时使用该参数）
15 -C 兼容旧版本（当server端和client端版本不一样时使用）
16 -M 设定TCP数据包的最大mtu值
17 -N 设定TCP不延时
18 -V 传输ipv6数据包
19
20 server专用参数
21 -D 以服务方式运行iperf，eg: iperf -s -D
22 -R 停止iperf服务，针对-D，eg: iperf -s -R
23
24 client端专用参数
25 -d 同时进行双向传输测试
26 -n 指定传输的字节数，eg: iperf -c 222.35.11.23 -n 100000
27 -r 单独进行双向传输测试
28 -t 测试时间，默认10秒, eg: iperf -c 222.35.11.23 -t 5
29 -F 指定需要传输的文件
30 -T 指定ttl值

```

3.4 tcp 包测试带宽示范

```

1  iperf -s #服务器默认等待接收tcp数据包
2  iperf -c 172.16.133.10 -i 1 #客户端向服务器默认发送tcp数据包

```

服务器给出的数据如下：

```

root@hty-controller:~# iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 4] local 172.16.133.10 port 5001 connected with 172.16.133.15 port 54719
[ ID] Interval      Transfer    Bandwidth
[ 4] 0.0-10.0 sec  1.10 GBytes  941 Mbits/sec

```

客户端给出的数据如下：

```

root@hty-controller:~# iperf -c 172.16.133.10 -i 1
-----
Client connecting to 172.16.133.10, TCP port 5001
TCP window size: 85.0 KByte (default)
-----
[ 3] local 172.16.133.15 port 54719 connected with 172.16.133.10 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 1.0 sec   114 MBytes  956 Mbits/sec
[ 3] 1.0- 2.0 sec   112 MBytes  942 Mbits/sec
[ 3] 2.0- 3.0 sec   112 MBytes  942 Mbits/sec
[ 3] 3.0- 4.0 sec   112 MBytes  942 Mbits/sec
[ 3] 4.0- 5.0 sec   113 MBytes  947 Mbits/sec
[ 3] 5.0- 6.0 sec   112 MBytes  937 Mbits/sec
[ 3] 6.0- 7.0 sec   113 MBytes  948 Mbits/sec
[ 3] 7.0- 8.0 sec   112 MBytes  942 Mbits/sec
[ 3] 8.0- 9.0 sec   111 MBytes  934 Mbits/sec
[ 3] 9.0-10.0 sec   113 MBytes  950 Mbits/sec
[ 3] 0.0-10.0 sec   1.10 GBytes  943 Mbits/sec
root@hty-controller:~#

```

3.5 udp 包同时测试抖动和丢包示范

- 1 iperf -s -u -i 1 #服务器等待接收udp数据包，每隔一秒钟显示信息
- 2 iperf -c 172.16.133.10 -u -i 1 #客户端向服务器发送udp数据包，每隔一秒钟显示信息

服务器给出的数据如下：

```

root@hty-controller:~ x root@hty-compute1:~ x root@hty-com
root@hty-controller:~# iperf -s -u -i 1
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 3] local 172.16.133.10 port 5001 connected with 172.16.133.15 port 40711
[ ID] Interval      Transfer    Bandwidth  Jitter    Lost/Total Datagrams
[ 3] 0.0- 1.0 sec   128 KBytes  1.05 Mbits/sec  0.013 ms  0/ 89 (0%)
[ 3] 1.0- 2.0 sec   128 KBytes  1.05 Mbits/sec  0.013 ms  0/ 89 (0%)
[ 3] 2.0- 3.0 sec   128 KBytes  1.05 Mbits/sec  0.012 ms  0/ 89 (0%)
[ 3] 3.0- 4.0 sec   128 KBytes  1.05 Mbits/sec  0.012 ms  0/ 89 (0%)
[ 3] 4.0- 5.0 sec   128 KBytes  1.05 Mbits/sec  0.012 ms  0/ 89 (0%)
[ 3] 5.0- 6.0 sec   129 KBytes  1.06 Mbits/sec  0.012 ms  0/ 90 (0%)
[ 3] 6.0- 7.0 sec   128 KBytes  1.05 Mbits/sec  0.014 ms  0/ 89 (0%)
[ 3] 7.0- 8.0 sec   128 KBytes  1.05 Mbits/sec  0.014 ms  0/ 89 (0%)
[ 3] 8.0- 9.0 sec   128 KBytes  1.05 Mbits/sec  0.013 ms  0/ 89 (0%)
[ 3] 9.0-10.0 sec   128 KBytes  1.05 Mbits/sec  0.012 ms  0/ 89 (0%)
[ 3] 0.0-10.0 sec   1.25 MBytes 1.05 Mbits/sec  0.013 ms  0/ 893 (0%)

```

客户端给出的数据如下:

```

root@hty-controller:~ x root@hty-compute1:~ x root@hty-c
root@hty-compute1:~# iperf -c 172.16.133.10 -u -i 1
-----
Client connecting to 172.16.133.10, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 3] local 172.16.133.15 port 40711 connected with 172.16.133.10 port 5001
[ ID] Interval      Transfer    Bandwidth  TCP window size: 65.3 KByte (default)
[ 3] 0.0- 1.0 sec   129 KBytes  1.06 Mbits/sec
[ 3] 1.0- 2.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 2.0- 3.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 3.0- 4.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 4.0- 5.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 5.0- 6.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 6.0- 7.0 sec   129 KBytes  1.06 Mbits/sec
[ 3] 7.0- 8.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 8.0- 9.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 9.0-10.0 sec   128 KBytes  1.05 Mbits/sec
[ 3] 0.0-10.0 sec   1.25 MBytes 1.05 Mbits/sec
[ 3] Sent 893 datagrams
[ 3] Server Report:
[ 3] 0.0-10.0 sec   1.25 MBytes 1.05 Mbits/sec  0.012 ms  0/ 893 (0%)
root@hty-compute1:~#

```

4 ping 命令的使用

4.1 ping 使用方法与参数说明

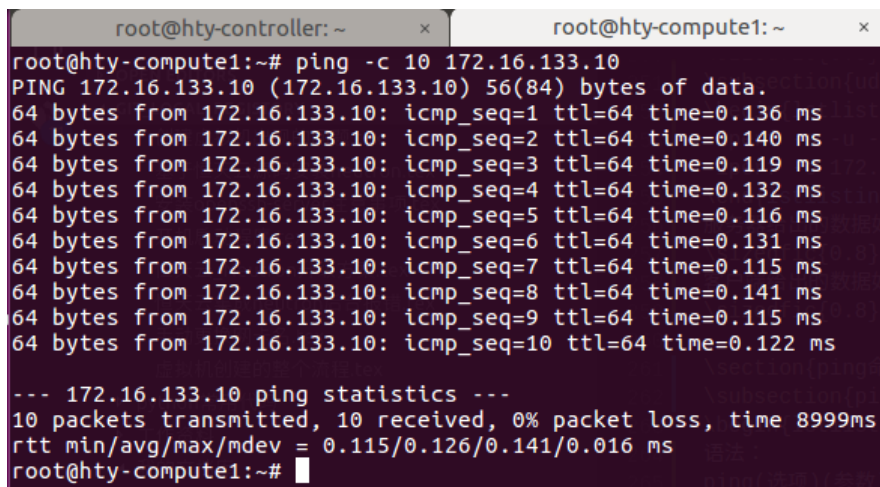
- 1 语法:
- 2 ping(选项)(参数)
- 3
- 4 选项:
- 5 -b : 后面接的是 broadcast 的 IP, 用在你『需要对整个网域的主机进行 ping』时;
- 6 -c : 后面接的是运行 ping 的次数, 例如 -c 5 ;
- 7 -n : 不进行 IP 与主机名的反查, 直接使用 IP ;
- 8 -s : 发送出去的 ICMP 封包大小, 默认为 56(bytes), 再加 8 bytes 的 ICMP 表头数据
- 9 -t : TTL 的数值, 默认是 255, 每经过一个节点就会少一;

```
10 -M [do|dont] : 主要在检测网络的 MTU 数值大小，两个常见的项目是：
11 do : 代表传送一个 DF (Don't Fragment) 旗标，让封包不能重新拆包与打包；
12 dont: 代表不要传送 DF 旗标，表示封包可以在其他主机上拆包与打包
13
14 参数：
15 目的主机：指定发送ICMP报文的目的地主机。
```

4.2 icmp 包测试延迟示范

```
1 ping -c 10 172.16.133.10
```

ping 服务器显示如下：



```
root@hty-controller: ~ x root@hty-compute1: ~ x
root@hty-compute1:~# ping -c 10 172.16.133.10
PING 172.16.133.10 (172.16.133.10) 56(84) bytes of data.
64 bytes from 172.16.133.10: icmp_seq=1 ttl=64 time=0.136 ms
64 bytes from 172.16.133.10: icmp_seq=2 ttl=64 time=0.140 ms
64 bytes from 172.16.133.10: icmp_seq=3 ttl=64 time=0.119 ms
64 bytes from 172.16.133.10: icmp_seq=4 ttl=64 time=0.132 ms
64 bytes from 172.16.133.10: icmp_seq=5 ttl=64 time=0.116 ms
64 bytes from 172.16.133.10: icmp_seq=6 ttl=64 time=0.131 ms
64 bytes from 172.16.133.10: icmp_seq=7 ttl=64 time=0.115 ms
64 bytes from 172.16.133.10: icmp_seq=8 ttl=64 time=0.141 ms
64 bytes from 172.16.133.10: icmp_seq=9 ttl=64 time=0.115 ms
64 bytes from 172.16.133.10: icmp_seq=10 ttl=64 time=0.122 ms

--- 172.16.133.10 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 8999ms
rtt min/avg/max/mdev = 0.115/0.126/0.141/0.016 ms
root@hty-compute1:~#
```

5 测试用例设计

5.1 带宽测试用例设计

1. 组一：使用 iperf 的 tcp 包测试物理机各网卡的网络带宽
2. 组二：使用 iperf 的 tcp 包测试不同情况 vm 的网络带宽
3. 对比两组数据，给出可用性结论

5.2 延迟测试用例设计

1. 组一：使用 ping 命令发送 icmp 包，测试物理机各网卡的延迟
2. 组二：使用 ping 命令发送 icmp 包，测试不同情况 vm 的延迟

3. 对比两组数据，给出可用性结论

5.3 抖动和丢包测试用例设计

1. 组一：使用 iperf 的 udp 包，指定带宽、传输字节数、测试时间等测试物理机
2. 组二：使用 iperf 的 udp 包，指定带宽、传输字节数、测试时间等测试虚拟机
3. 对比两组数据，给出稳定性结论

参考文献

- [1] [网络性能测试工具 Iperf 介绍](#)
- [2] [使用 iperf 测试网络性能](#)
- [3] [ping 命令](#)
- [4] [网络性能测试工具 iperf 详细使用图文教程](#)