2016302580096

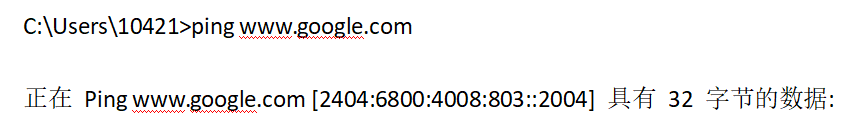
彭璇-卓一

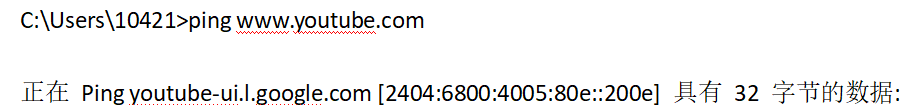
Ping

选取了几个生活中经常使用的主流网站：谷歌、必应、Youtube、百度来进行Ping命令的测试

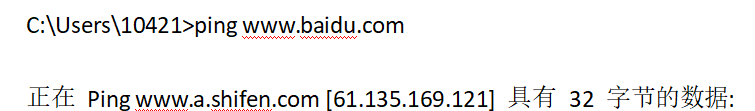
测试中获取到的信息：

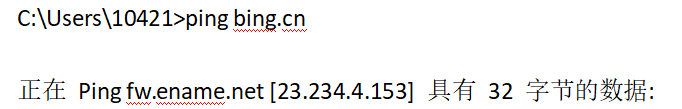
1. **Google和Youtube使用的是IPV6主机地址**



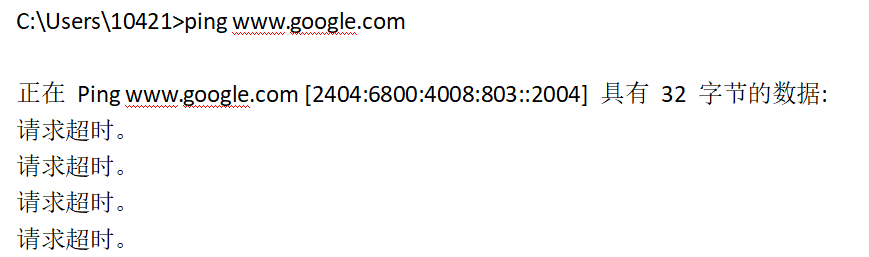


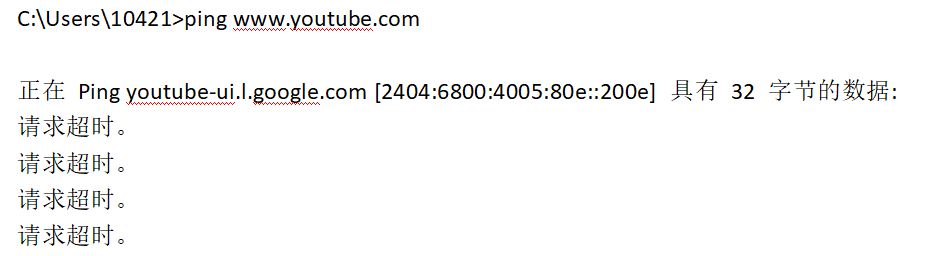
1. **当Ping百度和Bing国内版时，可以看到，数据包似乎被重定向到其他的域名上：**





1. 打开ShadowSocks的全局模式，连接到墙外的google服务器节点后，Ping 谷歌和Youtube，竟然还是出人意外地ping不通：





以下是测试的全过程

Microsoft Windows [版本 10.0.17134.590]

(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\10421>ping www.baidu.com

正在 Ping www.a.shifen.com [61.135.169.121] 具有 32 字节的数据:

来自 61.135.169.121 的回复: 字节=32 时间=271ms TTL=54

来自 61.135.169.121 的回复: 字节=32 时间=64ms TTL=54

来自 61.135.169.121 的回复: 字节=32 时间=75ms TTL=54

来自 61.135.169.121 的回复: 字节=32 时间=106ms TTL=54

61.135.169.121 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，

往返行程的估计时间(以毫秒为单位):

最短 = 64ms，最长 = 271ms，平均 = 129ms

C:\Users\10421>ping www.google.com

正在 Ping www.google.com [2404:6800:4008:803::2004] 具有 32 字节的数据:

请求超时。

请求超时。

请求超时。

请求超时。

2404:6800:4008:803::2004 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，

C:\Users\10421>

C:\Users\10421>

C:\Users\10421>ping www.google.com

正在 Ping www.google.com [2404:6800:4008:803::2004] 具有 32 字节的数据:

请求超时。

请求超时。

请求超时。

请求超时。

2404:6800:4008:803::2004 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，

C:\Users\10421>ping www.youtube.com

正在 Ping youtube-ui.l.google.com [2404:6800:4005:80e::200e] 具有 32 字节的数据:

请求超时。

请求超时。

请求超时。

请求超时。

2404:6800:4005:80e::200e 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，

C:\Users\10421>ping -6 www.youtube.com

正在 Ping youtube-ui.l.google.com [2404:6800:4005:80e::200e] 具有 32 字节的数据:

请求超时。

请求超时。

请求超时。

请求超时。

2404:6800:4005:80e::200e 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，

C:\Users\10421>ping 2404:6800:4005:80e::200e

正在 Ping 2404:6800:4005:80e::200e 具有 32 字节的数据:

请求超时。

请求超时。

请求超时。

请求超时。

2404:6800:4005:80e::200e 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，

C:\Users\10421>ping bing.cn

正在 Ping fw.ename.net [23.234.4.153] 具有 32 字节的数据:

来自 23.234.4.153 的回复: 字节=32 时间=198ms TTL=51

来自 23.234.4.153 的回复: 字节=32 时间=319ms TTL=51

来自 23.234.4.153 的回复: 字节=32 时间=230ms TTL=51

来自 23.234.4.153 的回复: 字节=32 时间=251ms TTL=51

23.234.4.153 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，

往返行程的估计时间(以毫秒为单位):

最短 = 198ms，最长 = 319ms，平均 = 249ms

C:\Users\10421>

C:\Users\10421>

C:\Users\10421>ping bing.com

正在 Ping bing.com [204.79.197.200] 具有 32 字节的数据:

请求超时。

请求超时。

来自 204.79.197.200 的回复: 字节=32 时间=189ms TTL=114

来自 204.79.197.200 的回复: 字节=32 时间=162ms TTL=114

204.79.197.200 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 2，丢失 = 2 (50% 丢失)，

往返行程的估计时间(以毫秒为单位):

最短 = 162ms，最长 = 189ms，平均 = 175ms

C:\Users\10421>ping google.com

正在 Ping google.com [2404:6800:4005:80c::200e] 具有 32 字节的数据:

请求超时。

请求超时。

请求超时。

请求超时。

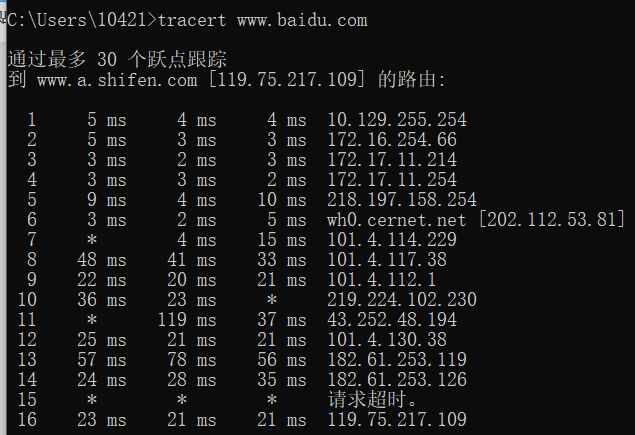
2404:6800:4005:80c::200e 的 Ping 统计信息:

数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，

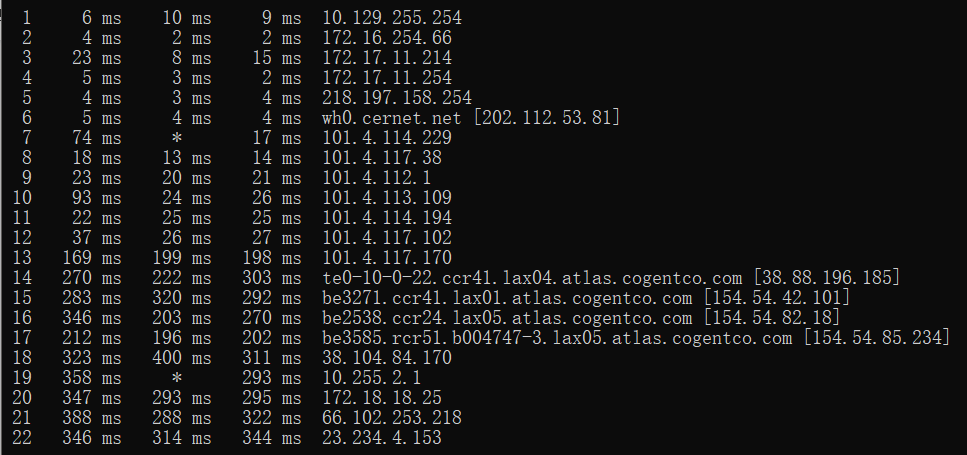
**Tracert**

相似的，选择了百度（国内网）和Bing（国外网）两个主机，在校园网（教育网）的实验条件下进行tracert命令的测试

测试结果如下：



（figure1: tracert 百度）



（figure2：tracert Bing）

可以看到，对两个不同主机的tracert，报文在传输过程中都经过了很多的中间点。

对出现过的IP地址进行地址查询，发现：

1. 以10、172、218开头的ip地址，全都属于武大的校内网
2. 共同出现的202.112.53.81，是属于广州的IP地址，也是全国校园网系统的节点。
3. 随后，报文将被转发到北京的校园网节点，随后才转发到目标地址（百度和bing的所在地）

发现从武大校园网发出去的报文，都要经过广州节点再转发到北京节点；

那么为什么，不能直接从武汉的节点发送到北京呢？

**Problem 3:**

1. Circuit-switched network will be more appropriate. Because the application will transmit data at a steady rate. We need to guarantee a certain amount of bandwidth for the app. More importantly, every time the application starts, it will be running for a long-period, so the circuit-switched network may well be more stable and suitable.
2. My opinion is: We still need preventative measures to inhibit congestion. Admittedly in the abovementioned circumstance, there is no chance for the congestion happen on the link. However, when too many packets arrive at the same router, it is highly likely to cause traffic jam because of the router’s limited rate of dispatching.

**Problem 5**

1. Suppose the serving time for a car at a tollbooth is 1min.

So at every tollbooth, 10 mins is needed for the whole caravan to pass.

Total-waiting-time: 30mins

And the time on the road is 150/100 = 1.5h = 90 mins.

So finally, it will cost the caravan 120mins in total to pass through the 3 tollbooths.

1. The total time on the road is identical as the situation of 10 cars. 90mins

The total waiting time at booth:

3x8 =24

So, total time: 114mins.

**Problem 7：**

For a bit in a packet:

Transmission delay: 56x 8/2000000 ≈0.24 ms

Propagation delay: 10ms

Queuing delay: 56 x8 /64000 = 7ms

So, total time on the way: 17.24ms