

XXX 大学

《计算机网路》实验报告

专业班级： 物联网 XXXX 学号： XXXXXXXXXX 姓名： 郭 XX

实验七 ICMP 协议及路由跟踪的研究

实验时间： 2018.11

【实验目的】

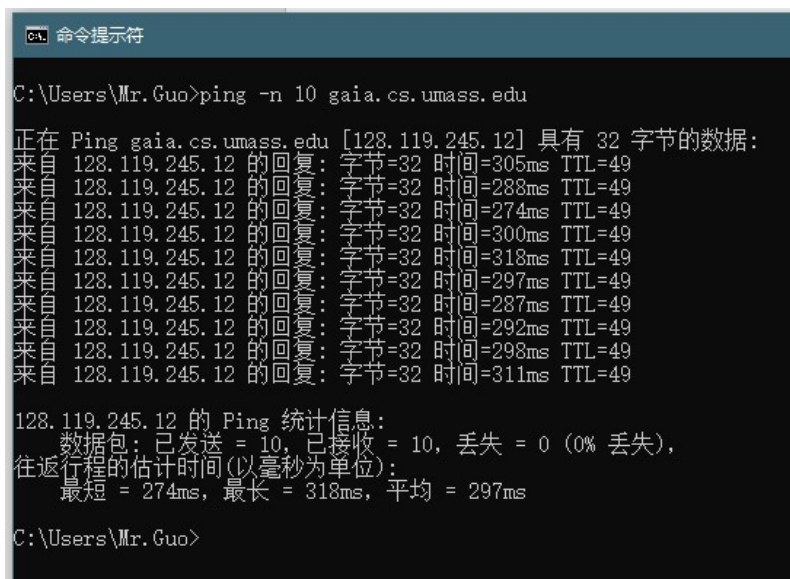
1. 快速简单了解 ICMP 协议
2. 了解 ICMP 的格式和内容
3. 了解 Ping 命令以及 Tracert 命令（Windows）与 ICMP 数据报的之间关系。
4. 了解 Traceroute 命令与 tracert 命令的区别

【实验步骤】

1. 打开 Wireshark 启动捕获(略)
2. 打开命令提示符使用 ping 命令 ping 其他大陆的服务器 10 次,这里以

gaia.cs.umass.edu 为例,命令语法

CMD > ping -n 10 gaia.cs.umass.edu



```
命令提示符
C:\Users\Mr.Guo>ping -n 10 gaia.cs.umass.edu

正在 Ping gaia.cs.umass.edu [128.119.245.12] 具有 32 字节的数据:
来自 128.119.245.12 的回复: 字节=32 时间=305ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=288ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=274ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=300ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=318ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=297ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=287ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=292ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=298ms TTL=49
来自 128.119.245.12 的回复: 字节=32 时间=311ms TTL=49

128.119.245.12 的 Ping 统计信息:
    数据包: 已发送 = 10, 已接收 = 10, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 274ms, 最长 = 318ms, 平均 = 297ms

C:\Users\Mr.Guo>
```

3. 保存数据包，并且启动下一次捕获（略）
4. 打开命令提示符进行路由跟踪，跟踪 gaia.cs.umass.edu，命令语法

CMD > tracert gaia.cs.umass.edu

```
命令提示符
C:\Users\Mr.Guo>tracert gaia.cs.umass.edu

通过最多 30 个跃点跟踪
到 gaia.cs.umass.edu [128.119.245.12] 的路由:

  1    2 ms    *      *      172.31.159.254
  2    <1 毫秒 <1 毫秒 <1 毫秒 10.161.147.145
  3    3 ms    1 ms    1 ms    pc77.zz.ha.cn [61.168.13.77]
  4    *      *      *      请求超时。
  5    28 ms   29 ms   27 ms   219.158.23.37
  6    30 ms   29 ms   30 ms   219.158.113.110
  7    30 ms   30 ms   30 ms   219.158.113.117
  8    192 ms  197 ms  197 ms  219.158.116.234
  9    228 ms  227 ms  225 ms  sjp-brdr-06.inet.qwest.net [63.146.27.85]
 10    299 ms  277 ms  268 ms  cmb-edge-03.inet.qwest.net [67.14.30.158]
 11    320 ms  326 ms  327 ms  65.126.225.186
 12    352 ms  299 ms  326 ms  core1-rt-et-4-3-0.gw.umass.edu [192.80.83.101]
 13    283 ms  272 ms  291 ms  128.119.0.8
 14    293 ms  271 ms  270 ms  cics-rt-xe-0-0-0.gw.umass.edu [128.119.3.32]
 15    335 ms  323 ms  278 ms  nsclbbs1.cs.umass.edu [128.119.240.253]
 16    276 ms  286 ms  302 ms  gaia.cs.umass.edu [128.119.245.12]

跟踪完成。
```

5. 保存数据包，并且启动下一次捕获（略），捕获接口选择虚拟机虚拟网卡
6. 在 Linux 虚拟机中进行路由跟踪，地址仍然是 gaia.cs.umass.edu，命令语法

traceroute gaia.cs.umass.edu

```
[root@localhost ~]# traceroute gaia.cs.umass.edu
traceroute to gaia.cs.umass.edu (128.119.245.12), 30 hops max, 60 byte packets
 1 gateway (192.168.184.2) 0.175 ms 0.141 ms 0.126 ms
 2 * * *
 3 * * *
 4 * * *
 5 * * *
 6 * * *
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 * * *
13 * * *
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
22 * * *
23 * * *
24 * * *
25 * * *
26 * * *
27 * * *
28 * * *
29 * * *
30 * * *
```

可以看到除了我虚拟机的虚拟路由网关，其他网关为了安全，均使用防火墙屏蔽过滤我的 UDP 路由跟踪请求，所以会出现都是*情况，后文有说。

7. 同样，Linux 还可以指定路由跟踪使用 ICMP 数据报。命令语法

traceroute gaia.cs.umass.edu -I

```

[root@localhost ~]# traceroute gaia.cs.umass.edu -I
traceroute to gaia.cs.umass.edu (128.119.245.12), 30 hops max, 60 byte packets
 1 gateway (192.168.184.2) 0.217 ms 0.062 ms 0.095 ms
 2 * * 172.31.159.254 (172.31.159.254) 26.811 ms
 3 10.161.147.145 (10.161.147.145) 1.292 ms 1.693 ms 1.669 ms
 4 pc77.zz.ha.cn (61.168.13.77) 1.635 ms 2.665 ms 2.632 ms
 5 * * *
 6 219.158.23.37 (219.158.23.37) 28.076 ms 28.734 ms 28.658 ms
 7 219.158.113.110 (219.158.113.110) 35.844 ms 35.812 ms 35.778 ms
 8 219.158.113.117 (219.158.113.117) 34.705 ms 34.651 ms 34.616 ms
 9 219.158.116.234 (219.158.116.234) 191.347 ms 191.328 ms 191.294 ms
10 sjp-brdr-04.inet.qwest.net (63.146.27.85) 200.401 ms 200.362 ms 194.195 ms
11 cmb-edge-03.inet.qwest.net (67.14.30.158) 267.877 ms 267.859 ms 267.710 ms
12 65.126.225.186 (65.126.225.186) 282.694 ms 282.662 ms 292.916 ms
13 core1-rt-et-4-3-0.gw.umass.edu (192.80.83.101) 305.513 ms 305.806 ms 307.985 ms
14 n5-rt-1-1-et-0-0-0.gw.umass.edu (128.119.0.8) 302.827 ms 302.812 ms 289.889 ms
15 cics-rt-xe-0-0-0.gw.umass.edu (128.119.3.32) 261.251 ms 260.933 ms 261.292 ms
16 nscslbbs1.cs.umass.edu (128.119.240.253) 320.320 ms 328.028 ms 326.568 ms
17 gaia.cs.umass.edu (128.119.245.12) 285.200 ms 288.083 ms 288.027 ms

```

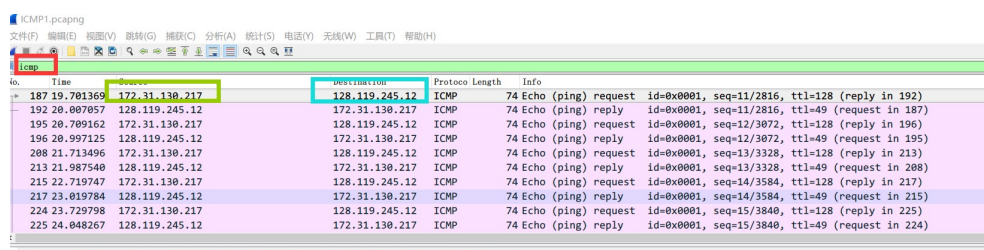
8. 进行实验数据结果分析。

【实验结果】

1. 您的主机的 IP 地址是多少？ 目标主机的 IP 地址是多少？

ANS:我的 IP 172.31.130.217。

目标 IP 128.119.245.12



No.	Time	Source	Destination	Protocol	Length	Info
187	19.701369	172.31.130.217	128.119.245.12	ICMP	74	Echo (ping) request id=0x0001, seq=11/2816, ttl=128 (reply in 192)
192	20.007057	128.119.245.12	172.31.130.217	ICMP	74	Echo (ping) reply id=0x0001, seq=11/2816, ttl=49 (request in 187)
195	20.709162	172.31.130.217	128.119.245.12	ICMP	74	Echo (ping) request id=0x0001, seq=12/3072, ttl=128 (reply in 196)
196	20.997125	128.119.245.12	172.31.130.217	ICMP	74	Echo (ping) reply id=0x0001, seq=12/3072, ttl=49 (request in 195)
208	21.713496	172.31.130.217	128.119.245.12	ICMP	74	Echo (ping) request id=0x0001, seq=13/3328, ttl=128 (reply in 213)
213	21.987540	128.119.245.12	172.31.130.217	ICMP	74	Echo (ping) reply id=0x0001, seq=13/3328, ttl=49 (request in 208)
215	22.719747	172.31.130.217	128.119.245.12	ICMP	74	Echo (ping) request id=0x0001, seq=14/3584, ttl=128 (reply in 217)
217	23.019784	128.119.245.12	172.31.130.217	ICMP	74	Echo (ping) reply id=0x0001, seq=14/3584, ttl=49 (request in 215)
224	23.729798	172.31.130.217	128.119.245.12	ICMP	74	Echo (ping) request id=0x0001, seq=15/3840, ttl=128 (reply in 225)
225	24.048267	128.119.245.12	172.31.130.217	ICMP	74	Echo (ping) reply id=0x0001, seq=15/3840, ttl=49 (request in 224)

2. 为什么 ICMP 数据包没有源端口号和目的端口号？

ANS: 因为 ICMP 协议是网络层的协议，它不需要传输层 TCP 和 UDP 承载，直接使用 IP 报承载，因此不需要源端口号目的端口号，只需要目的地址即可。

请协助改善这篇条目，更进一步的信息可能会在讨论页或扩充请求中找到。请在扩充条目后将此模板移除。

互联网控制消息协议（英语：Internet Control Message Protocol，缩写：**ICMP**）是互联网协议族的核心协议之一。它用于TCP/IP网络中发送控制消息，提供可能发生在通信环境中的各种问题反馈，通过这些信息，使管理者可以对所发生的问题作出诊断，然后采取适当的措施解决。

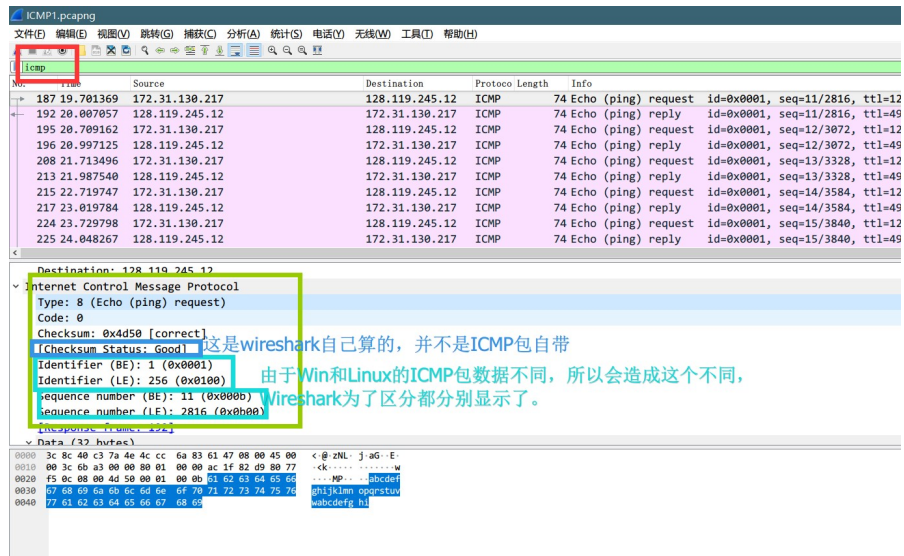
ICMP^[1]依靠IP来完成它的任务，它是IP的主要部分。它与传输协议（如TCP和UDP）显著不同：它一般不用于在两点间传输数据，它通常不由网络程序直接使用，除了ping和traceroute这两个特别的例子，^[2]4中的ICMP被称作ICMPv4，IPv6中的ICMP则被称作ICMPv6。

目录 [隐藏]

3. 查看任意的请求 ICMP 数据包，ICMP 类型和代码是什么？ 该 ICMP 数据包还有哪些其他字段？ 校验和，序列号和标识符字段有多少字节

ANS: ICMP 类型是 8（代表 ICMP 请求），代码是 0，ICMP 数据包还包括校验码，ID 值，以及序号。校验和，序列号，标识符都是 16 个字符，4 个字节。

图在下页。



这是维基百科的解释：

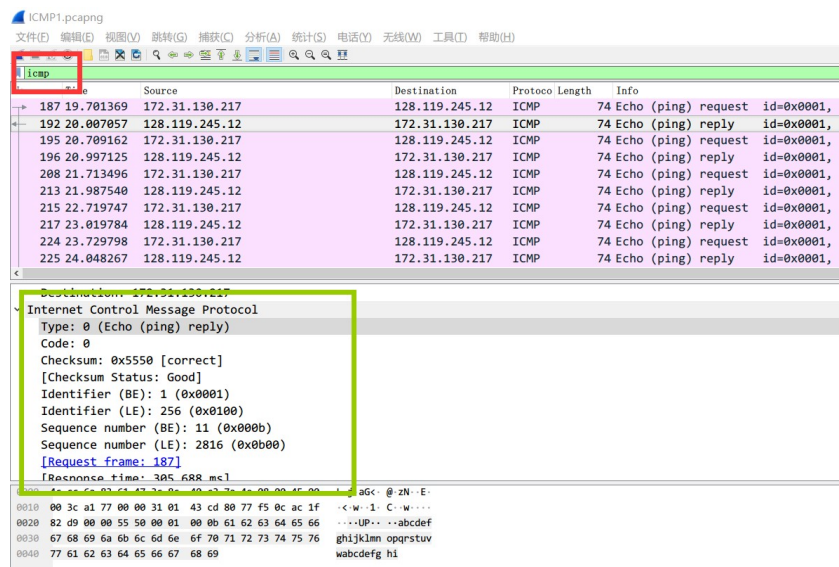
ICMP包头从IP报头的第160位开始（IP首部20字节）（除非使用了IP报头的可选部分）。

Bits	160-167	168-175	176-183	184-191
160	Type	Code	校验码 (checksum)	
192	ID		序号 (sequence)	

- **Type** - ICMP的类型,标识生成的错误报文;
- **Code** - 进一步划分ICMP的类型,该字段用来查找产生错误的原因; 例如, ICMP的目标不可达类型可以把这个位设为1至15来表示不同的意思。
- **Checksum** - 校验码部分,这个字段包含有从ICMP报头和数据部分计算得来的,用于检查错误的的数据, 其中此校验码字段的值视为0。
- **ID** - 这个字段包含了ID值,在Echo Reply类型的消息中要返回这个字段。
- **Sequence** - 这个字段包含一个序号,同样要在Echo Reply类型的消息中要返回这个字段。

4. 查看任意的响应 ICMP 数据包，ICMP 类型和代码是什么？ 该 ICMP 数据包还有哪些其他字段？ 校验和，序列号和标识符字段有多少字节？

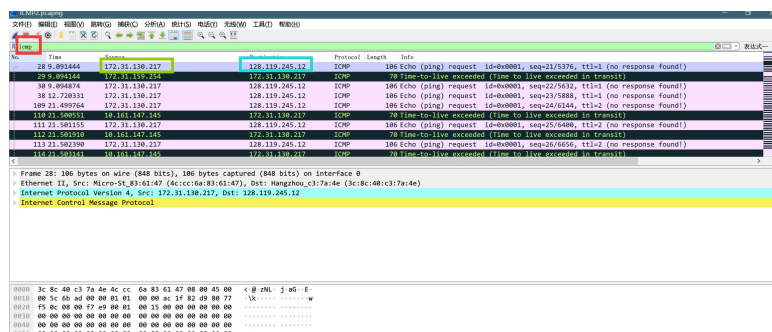
ANS: ICMP 类型是 0（代表 ICMP 响应），代码是 0，ICMP 数据包还包括校验码，ID 值，以及序号。校验和，序列号，标识符都是 16 个字符，4 个字节。



5. 您的主机的 IP 地址是多少？ 目标主机的 IP 地址是多少？

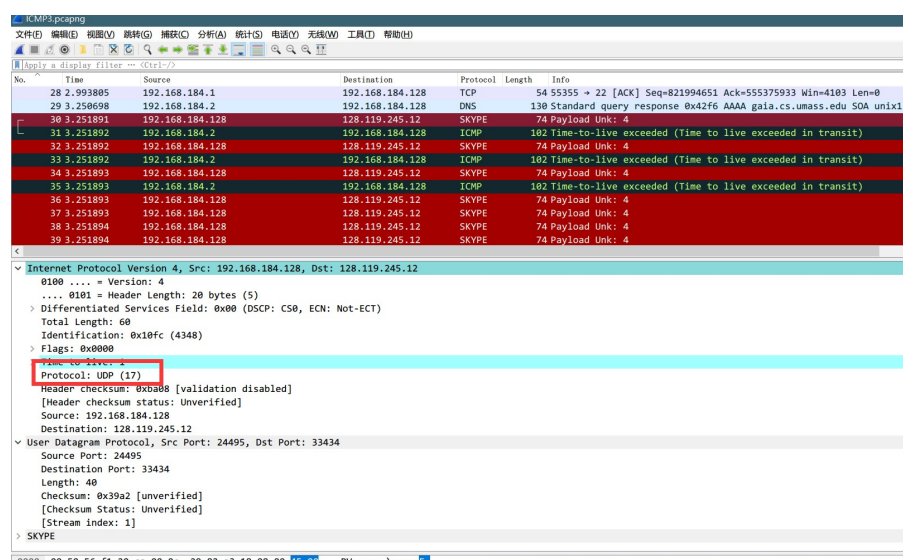
ANS: 我的 IP 172.31.130.217。

目标 IP 128.119.245.12



6. 如果 ICMP 发送了 UDP 数据包（如在 Unix / Linux 中），那么探测数据包的 IP 协议号仍然是 01 吗？ 如果没有，它会是什么？

ANS:根据我抓包 Linux 虚拟机路由跟踪结果，发送请求路由跟踪的数据包时 UDP 数据包，因此 IP 承载上层协议号时 17。虽然说是路由跟踪请求，但是 Wireshark 把这里的请求全部识别成了 SKYPE 协议，我不知道为什么，但是可以确认这就是 TTL 和发送端口逐渐增大的路由跟踪请求，且除了我的虚拟机虚拟网络网关响应，其他路由网关均未响应。所以可以看到除了我的虚拟机虚拟网络网关发了三个 TTL 超时 ICMP 消息，其他只有请求没有回复。



7. 检查屏幕截图中的 ICMP 响应数据包。 这与本实验的前半部分中的 ICMP ping 查询数据包不同吗？ 如果不同，请解释为什么？

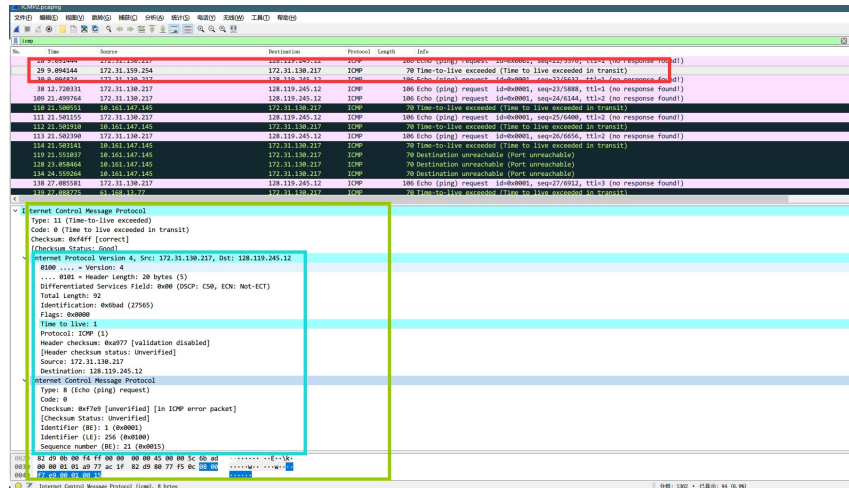
ANS:路由跟踪的 ICMP 响应数据包（非超时错误数据包）的 ICMP 的 TYPE

和序列号不同于前半部分 ICMP 的 PING 的查询数据包。

因为 TYPE=0 代表 ICMP 响应，且每次的序列号都不同。

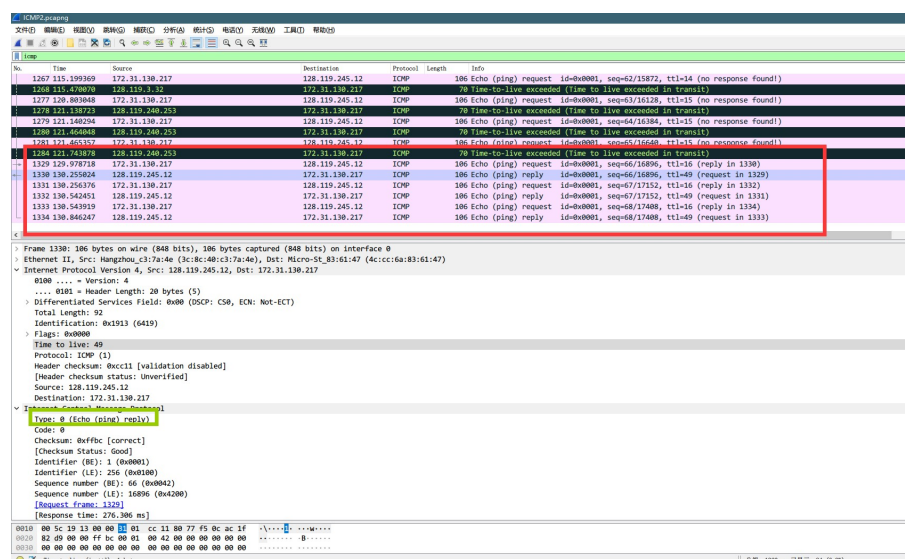
8. 检查屏幕截图中的 ICMP 错误数据包。它具有比 ICMP 响应数据包更多字段。这个数据包包含哪些内容？

ANS:多了 ICMP 请求数据包的内容，图中青色部分。



9. 检查源主机收到的最后三个 ICMP 数据包。这些数据与 ICMP 错误数据包有何不同？他们为什么不同？

ANS:最后三个 ICMP（响应）数据包是目标主机发送给我的 ICMP 回应数据包，因为路由查询是使用逐渐递增 TTL 的 ICMP 查询数据包，最后的 ICMP 查询数据包的 TTL 已经大于到达目的主机中间路由跃点数，因此不会被目标主机丢弃，发送 ICMP 超时的数据包，所以只会收到 ICMP 响应数据包。



10. 在 traceroute 跟踪测量中，是否有一个连接的延迟比其他连接长得多？请参

阅图 4 中的屏幕截图，是否有连接的延迟明显长于其他连接？根据路由器名称，您能猜出这个连接末端的两个路由器的位置吗？

ANS:没看懂这题什么意思，我去查了下网上其他人答案，意思是说在路由跟踪中有一个节点延迟会突然增大（过太平洋电缆/光缆）到美国大陆，然后让我们确认这两个路由器的位置。

```
C:\Users\Mr.Guo>tracert gaia.cs.umass.edu

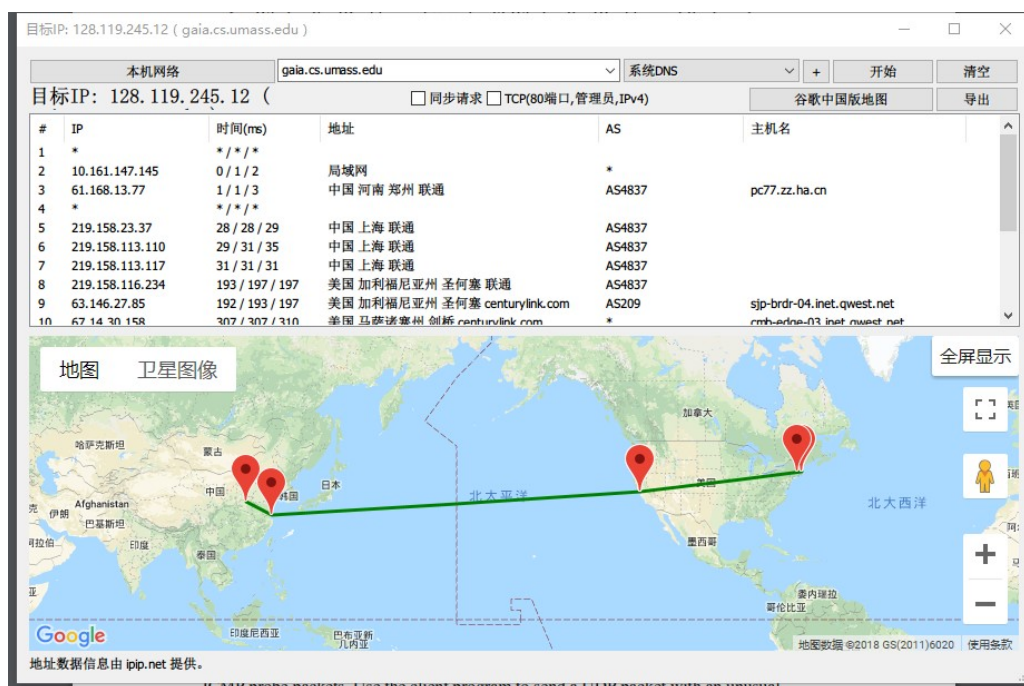
通过最多 30 个跃点跟踪
到 gaia.cs.umass.edu [128.119.245.12] 的路由:

 1  2 ms    *      *      172.31.159.254
 2  <1 毫秒 <1 毫秒 <1 毫秒 10.161.147.145
 3  3 ms    1 ms   1 ms   pc77.zz.ha.cn [61.168.13.77]
 4  *      *      *      请求超时。
 5  28 ms   29 ms  27 ms  219.158.23.37
 6  30 ms   29 ms  30 ms  219.158.113.110
 7  30 ms   30 ms  30 ms  219.158.113.117
 8  192 ms  197 ms  197 ms  219.158.116.234
 9  228 ms  227 ms  225 ms  sjp-brdr-04.inet.qwest.net [63.146.27.85]
10  299 ms  277 ms  268 ms  cmb-edge-03.inet.qwest.net [67.14.30.158]
11  320 ms  326 ms  327 ms  65.126.225.186
12  352 ms  299 ms  326 ms  core1-rt-et-4-3-0.gw.umass.edu [192.80.83.101]
13  283 ms  272 ms  291 ms  128.119.0.8
14  293 ms  271 ms  270 ms  cics-rt-xe-0-0-0.gw.umass.edu [128.119.3.32]
15  335 ms  323 ms  278 ms  nscslbbs1.cs.umass.edu [128.119.240.253]
16  276 ms  286 ms  302 ms  gaia.cs.umass.edu [128.119.245.12]

跟踪完成。
```

如图所示，在 TTL=7，经过第 7 个路由节点延迟还很小，在 TTL=8，即经过第 8 个节点之后突然增大。

我使用 Best trace 重新进行了可视化路由跟踪，发现这两个节点分别在上海和圣何塞。



额外题：对于一个编程任务，您可能创建了一个 UDP 客户端 ping 程序。与标准 ping 程序不同，此 ping 程序发送 UDP 探测包而不是 ICMP 探测包。使用客户端程序将具有异常目标端口号的 UDP 数据包发送到某个活动主机。同时，使用 Wireshark 捕获目标主机的任何响应。提供响应的 Wireshark 屏幕截图以及响应分析。

ANS:我同样使用我虚拟机的 Linux (CentOS) 进行了相同的路由跟踪，但是在实验过程中发现了除了我的虚拟机虚拟网络的路由网关响应了之外，其他任何网关，包括目标主机都没有响应我的查询。

上网查阅了下资料：

<https://www.zhihu.com/question/50220087>

<https://serverfault.com/questions/334029/what-does-mean-when-tracerte>

根据网上资料猜测为了安全，所有路由网关防火墙都封了 UDP 路由跟踪数据包查询，因此不会响应。

还可以看到我指定为 ICMP 数据包路由跟踪后，效果跟 Windows 的 tracert 差不多。

图在实验过程中。