实验三

ICMP

**实验目的**

在本实验里，我们将探究ICMP协议的几个方面：

* 由Ping程序生成的ICMP消息;
* 由Traceroute程序生成的ICMP消息;
* ICMP消息的格式和内容。

在开始实验之前，请先复习课本第5.6节中的ICMP材料。我们将在Win10环境中进行这个实验室，但该实验很容易迁移到Unix或Linux环境。

**实验步骤**

1. ICMP和Ping

我们从捕获Ping程序生成的数据包开始ICMP实验。Ping程序是一个简单的工具，它允许任何人（例如网络管理员）验证主机是否处于活动状态。源主机中的Ping程序将数据包发送到目标IP地址；如果目标是活动的，则目标主机返回数据包到源主机来响应。你可能已经猜到（既然这个实验是关于ICMP的）这两个数据包都是ICMP包。

执行以下操作:

* 首先运行Windows命令提示符cmd
* 启动Wireshark数据包嗅探器，并开始Wireshark分组捕获。
* 在cmd命令行中键入"ping –n 10 hostname"(不含引号)，其中hostname是位于地球上另一大陆的主机。你可以输入斯坦福大学的Web服务器www.stanford.edu。参数 -n 10表示应该发送10条ping消息。然后输入return运行Ping程序。
* 当Ping程序终止时，停止Wireshark中的数据包捕获。

在实验结束时，你的命令提示符窗口应该类似于图1。在本例中，源ping程序来自位于武汉市的家庭网络，而目标位于斯坦福大学。在这个窗口中，可以看到源ping程序发送了10个查询包，并收到了10个响应。还要注意，对于每个响应，源计算往返时间（RTT），对于10个包，该时间平均为242毫秒。

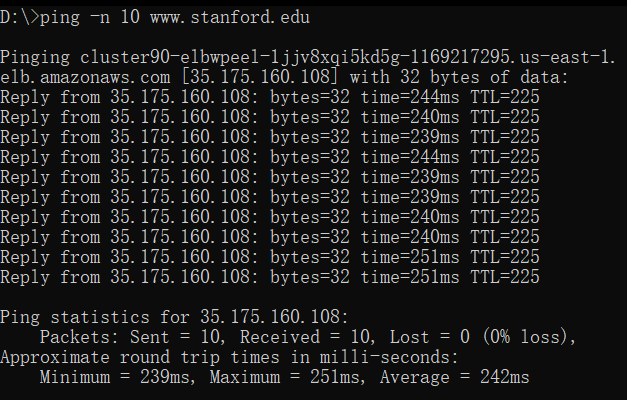


图1 输入Ping命令后的 cmd窗口

图2提供了Wireshark输出的屏幕截图，是在“icmp”被输入到过滤器窗口之后的显示。注意，数据包列表显示了20个数据包：源发送的10个Ping查询和源接收到的10个Ping响应。还要注意，源的IP地址是192.168/12格式的私有地址（在NAT之后）；目标的IP地址是斯坦福大学的Web服务器地址。现在来放大第一个数据包（由客户端发送）；在下图中，数据包内容区域提供有关此数据包的信息。我们看到这个包中的IP数据报的协议号01，这是ICMP的协议号。这意味着IP数据报的有效负载是ICMP包。

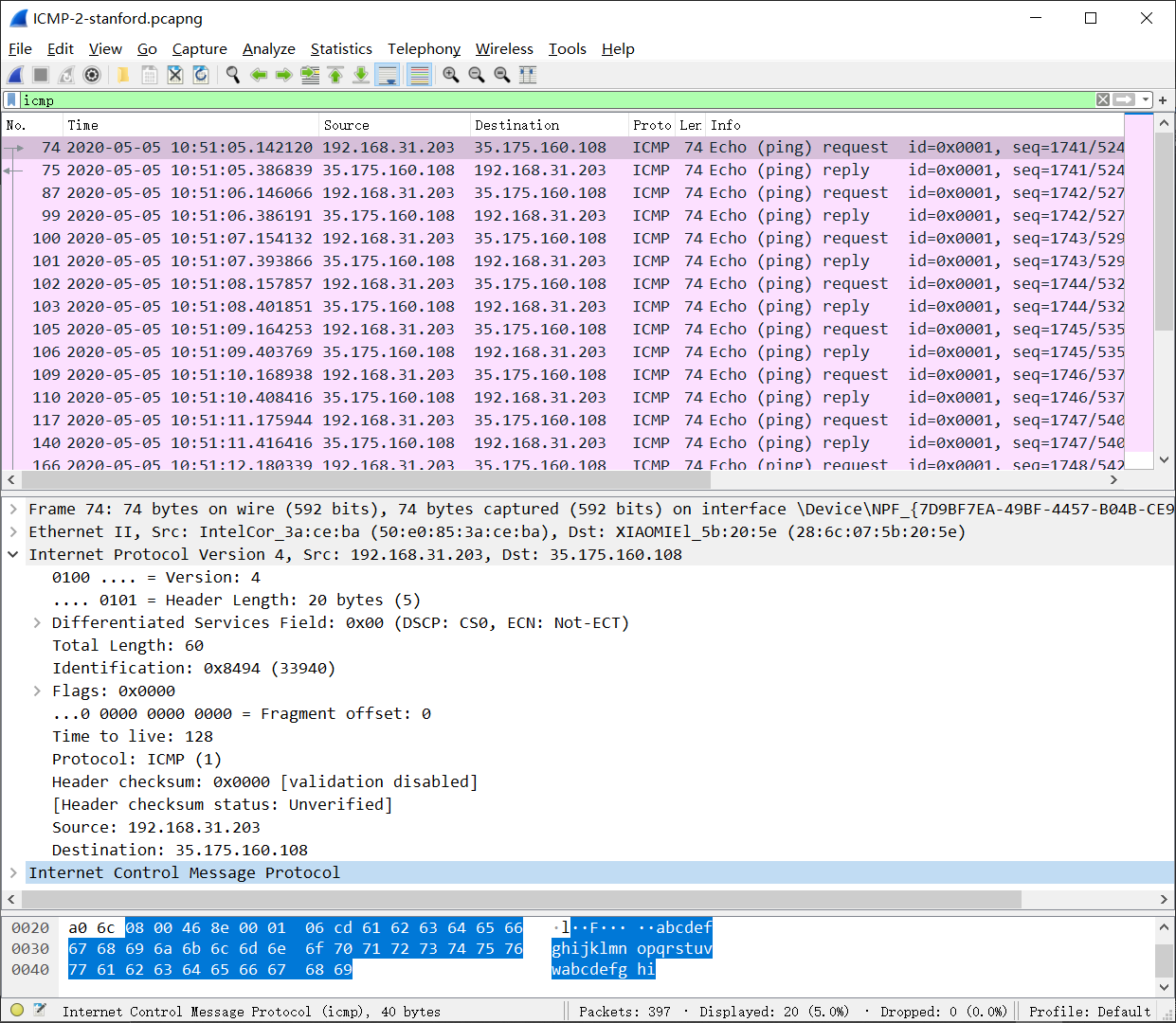


图2 Ping程序的Wireshark输出展开IP协议

图3聚焦的是同一个ICMP，但在分组内容窗口中展开了ICMP协议信息。注意，这个ICMP包是类型8，代码0，即所谓的ICMP“echo request”请求包（参见教材图5.19）。还要注意，这个ICMP包包含一个校验和、一个标识符和一个序列号。

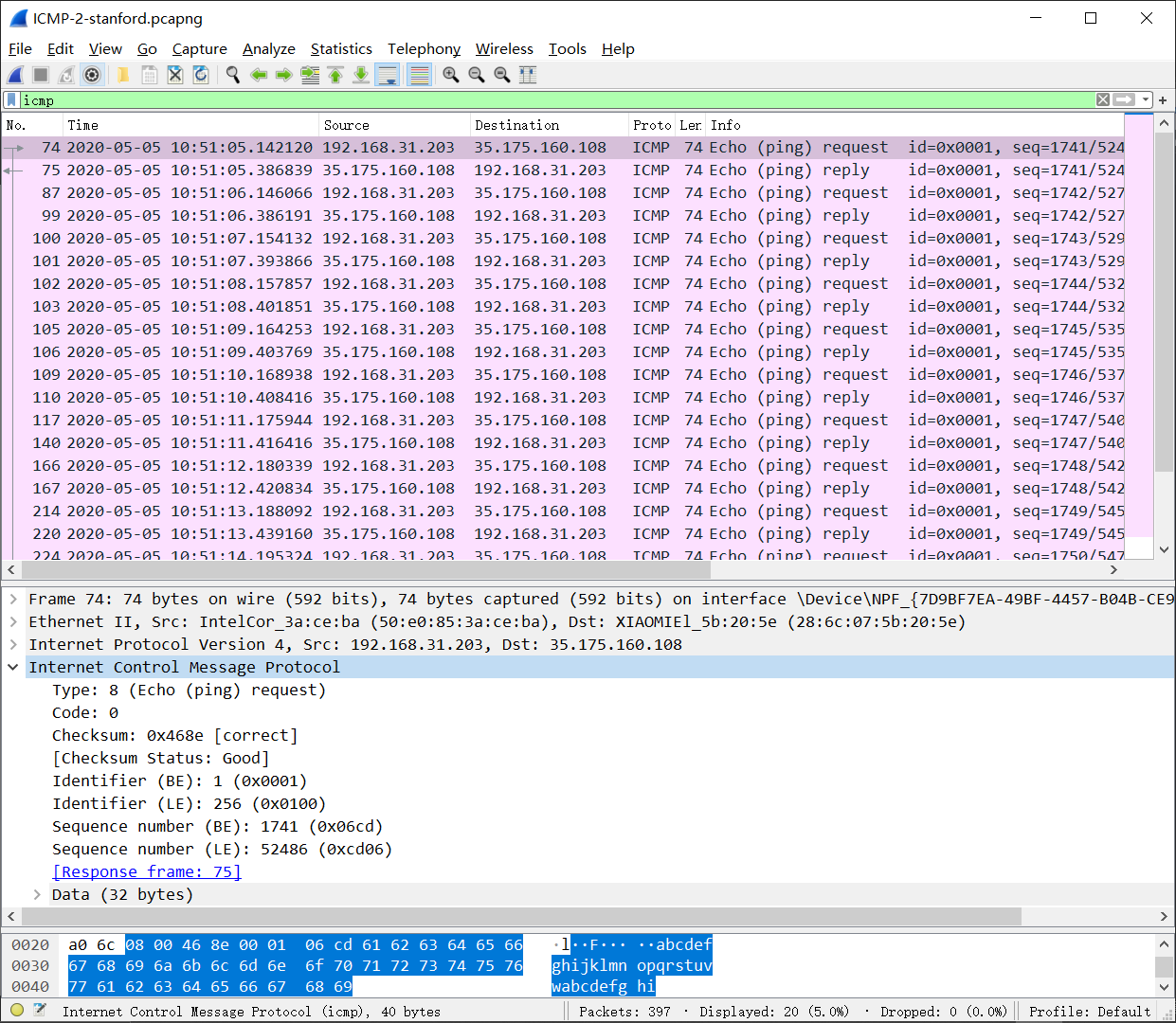


图3 Ping程序的Wireshark捕获展开ICMP

实验报告中记录这个实验的内容：

包括命令提示窗口的屏幕截图，类似于上面的图1。在回答下面的问题时，应该尽可能在用来回答问题的追踪捕获中含一份打印出来的数据包。要打印数据包，请使用“文件”->“打印到pdf文件”，选择“仅选定数据包”，选择“数据包摘要行”，然后选择回答问题所需的数据包最小详细信息量。

实验报告中回答下列问题：

1. 你的主机的IP地址是什么？目标主机的IP地址是什么？
2. 为什么ICMP数据包没有源端口号和目标端口号？
3. 检查主机发送的ping请求数据包之一。ICMP类型和代码是什么？这个ICMP包还有哪些字段？校验和、序列号和标识符字段是多少字节？
4. 检查相应的ping应答包。ICMP类型和代码是什么？这个ICMP包还有哪些字段？校验和、序列号和标识符字段是多少字节？
5. ICMP和Traceroute

现在让我们通过捕获Traceroute程序生成的包来继续我们的ICMP探险。你可能还记得Traceroute程序可以用来计算包从源到目的地的路径。Traceroute在教材的第1.4节和第5.6节中有讨论。

Traceroute在Unix/Linux/MacOS和Windows中实现的方式不同。在Unix/Linux中，源使用一个不可能的目标端口号（通常80）向目标目标发送一系列UDP数据包；在Windows中，源向目标目标发送一系列ICMP数据包。对于这两个操作系统，程序发送TTL=1的第一个包，TTL=2的第二个包，依此类推。回想一下，当数据包通过路由器时，路由器将递减数据包的TTL值。当数据包到达TTL=1的路由器时，路由器将ICMP错误数据包发送回源主机。下面我们将使用本机Windows tracert程序。pingplotter是一个非常好的Windows Traceroute程序的共享版本（www.pingplotter.com）。我们将在后面的 Wireshark IP 实验中使用 pingplotter，该程序提供了更多的功能。

执行以下操作：

* 首先打开Windows命令行
* 启动Wireshark分组嗅探器，并开始Wireshark数据包捕获。
* 在MS-DOS命令行中键入"tracert hostname"（不带引号），其中hostname是另一个大陆上的主机（注意，在Windows机器上，命令是tracert而不是traceroute）。在这里，我们输入法国计算机科学研究所的Web服务器www.inria.fr。然后键入回车运行Traceroute程序。
* 当Traceroute程序终止时，停止Wireshark中的数据包捕获。

在实验完成后，你的命令行提示符窗口应该类似于图4。在这个截图中，客户端Traceroute程序位于武汉市，目标目的地则位于法国。从这个图中我们可以看到，对于每个TTL值，源程序发送三个探测包。Traceroute显示每个探测包的rtt，以及返回ICMP TTL exceeded消息的路由器的IP地址（可能还有名称）。

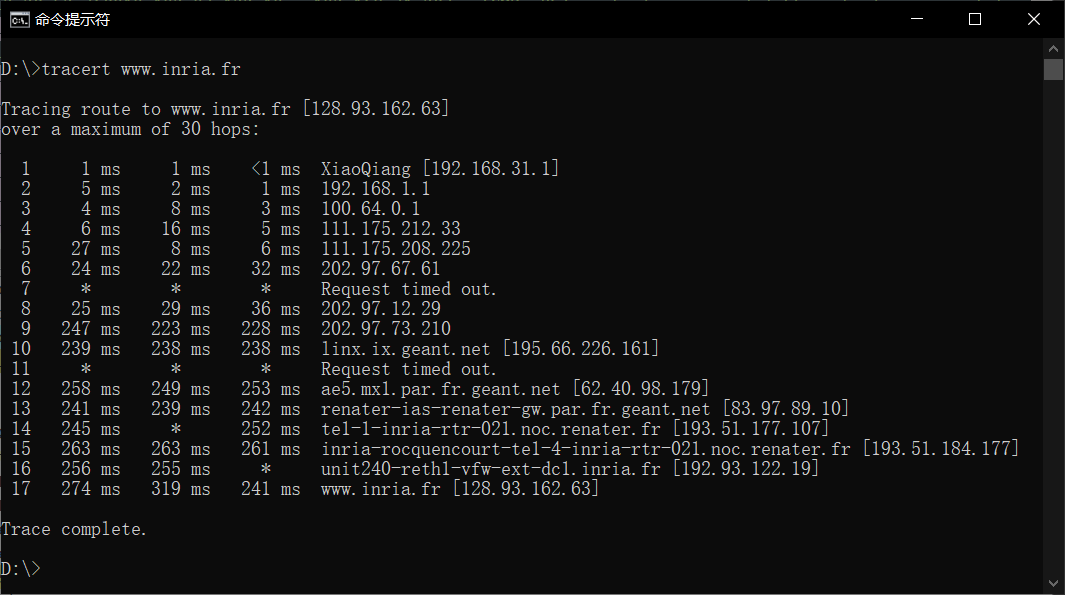


图4 cmd中显示的Traceroute程序的结果

图5显示路由器返回的ICMP数据包的Wireshark窗口。请注意，此ICMP错误数据包包含的字段比Ping ICMP消息多得多。

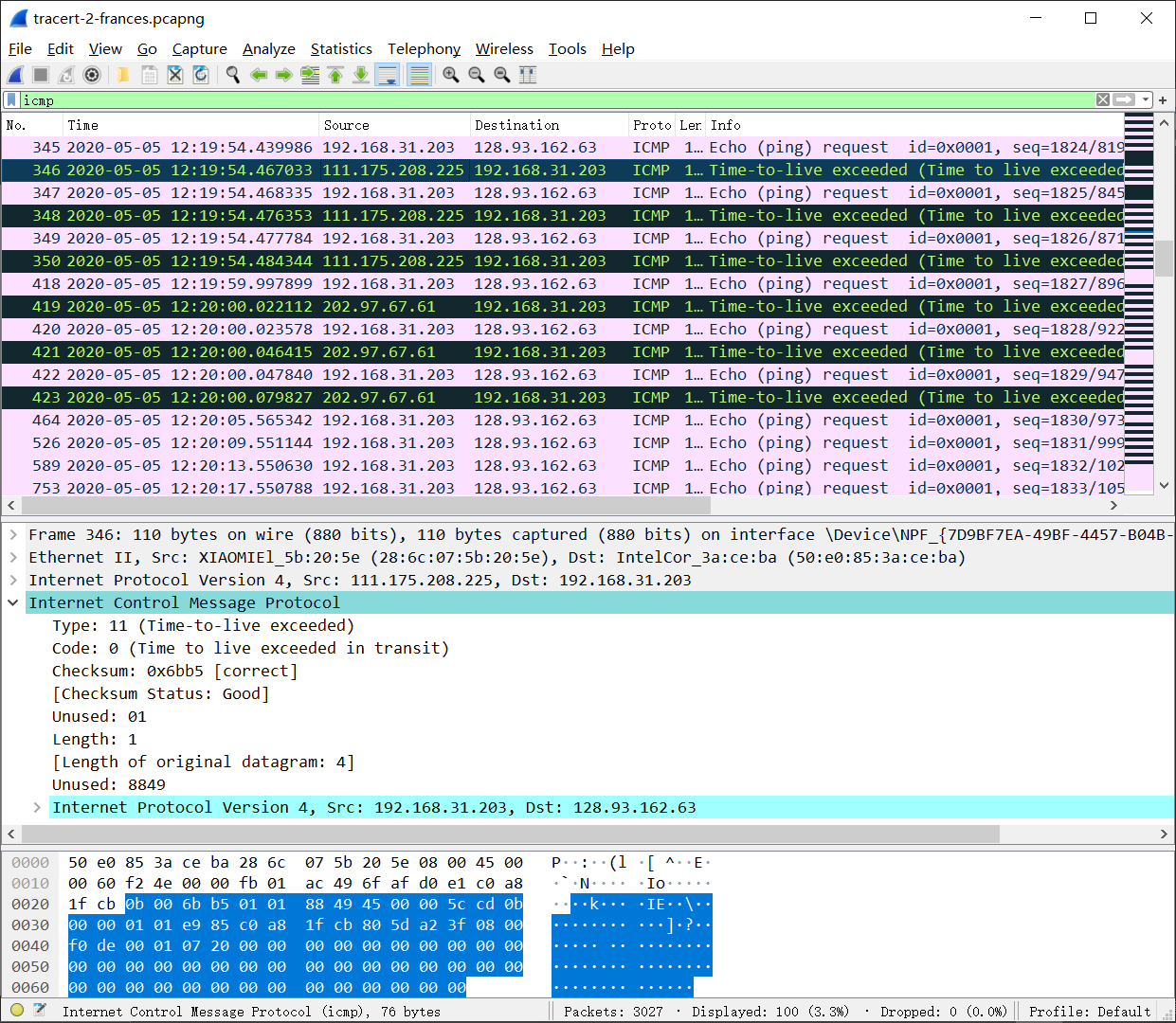


图5 Wireshark窗口中一个ICMP错误数据包展开ICMP字段

实验报告中记录这个实验的内容：

对于这部分实验，你应该提交命令提示窗口的屏幕截图。如果可能的话，在回答下面的问题时，你应该在你用来回答所问问题的轨迹中交一份打印出来的数据包。要打印数据包，请使用“文件”->“打印到pdf文件”，选择“仅选定数据包”，选择“数据包摘要行”，然后选择回答问题所需的最小数据包详细信息量。

回答关于的以下问题:

1. 你的主机的IP地址是什么？目标主机的IP地址是什么？
2. 如果ICMP改为发送UDP数据包（在Unix/Linux中），探测数据包的IP协议号是否仍然是01？如果不是，会是什么？
3. 检查截屏图中的ICMP echo包。这与本实验前半部分的ICMP ping查询数据包不同吗？如果是，怎么会这样？
4. 检查截屏图中的ICMP错误数据包。它的字段比ICMP echo包多。包括哪些字段？
5. 检查源主机接收的最后三个ICMP数据包。这些数据包与ICMP错误数据包有何不同？它们为什么不同？
6. 在tracert测量中，是否有一个链路的延迟比其他链路长得多？参考图4中的截图，是否有一个链接的延迟比其他链接长得多？根据路由器的名称，你能猜出这两个路由器在这个链路的末端的位置吗？
7. 额外加分

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探测数据包，而不是ICMP探测数据包。使用客户端程序将目标端口号异常的UDP数据包发送到某个活动主机。同时，使用Wireshark捕获来自目标主机的任何响应。提供响应的Wireshark截图来进行分析。