实验七 初步认识和使用 Wireshark

一、实验目的

Wireshark 是非常流行的网络封包分析软件，可以截取各种网络数据包，并显示数据包详细信息。

1、初步认识和理解 Wireshark 网络抓包工具。

2、通过观察 Wireshark 工具抓到的数据包，认识真实网络环境中传输的网络数据到底是什么。

3、通过过滤器可以筛选出想要分析的内容。包括按照协议过滤、端口和主机名过滤、数据包内容过滤。

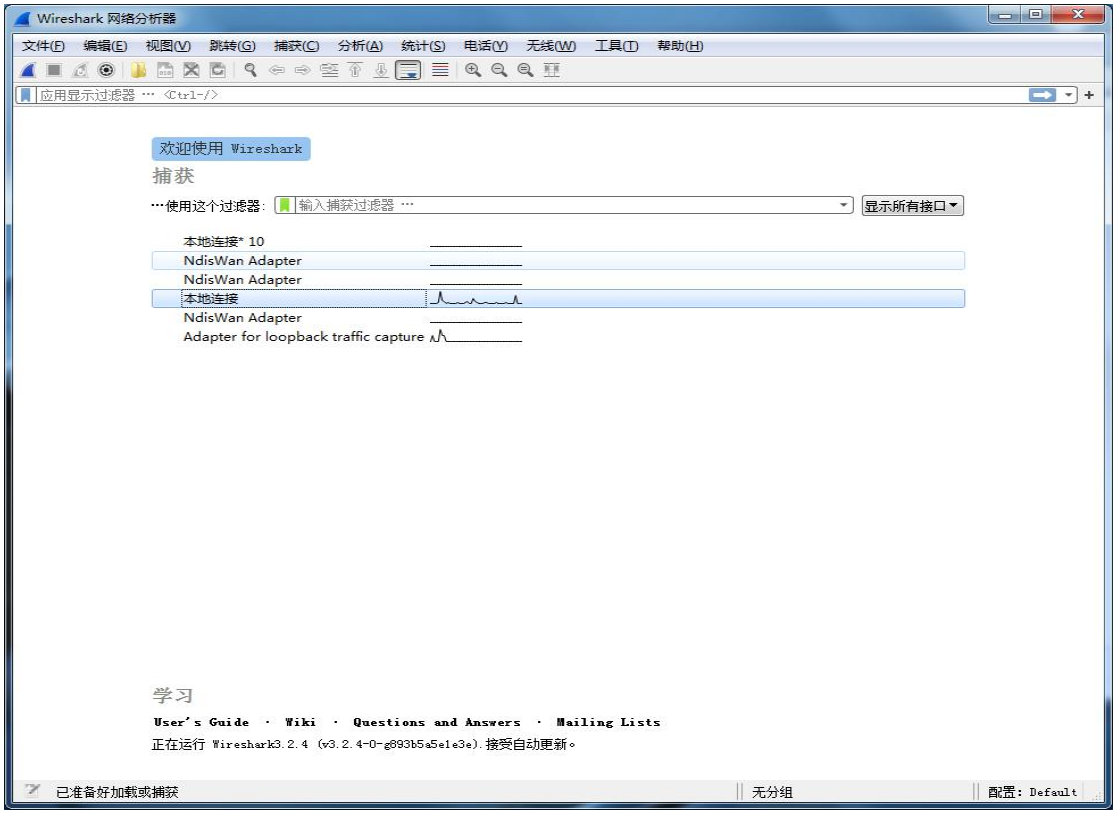
二、实验要求

1、掌握 Wireshark 抓包工具的常用功能：开始捕获、停止捕获、保存捕获、显示过滤器等。

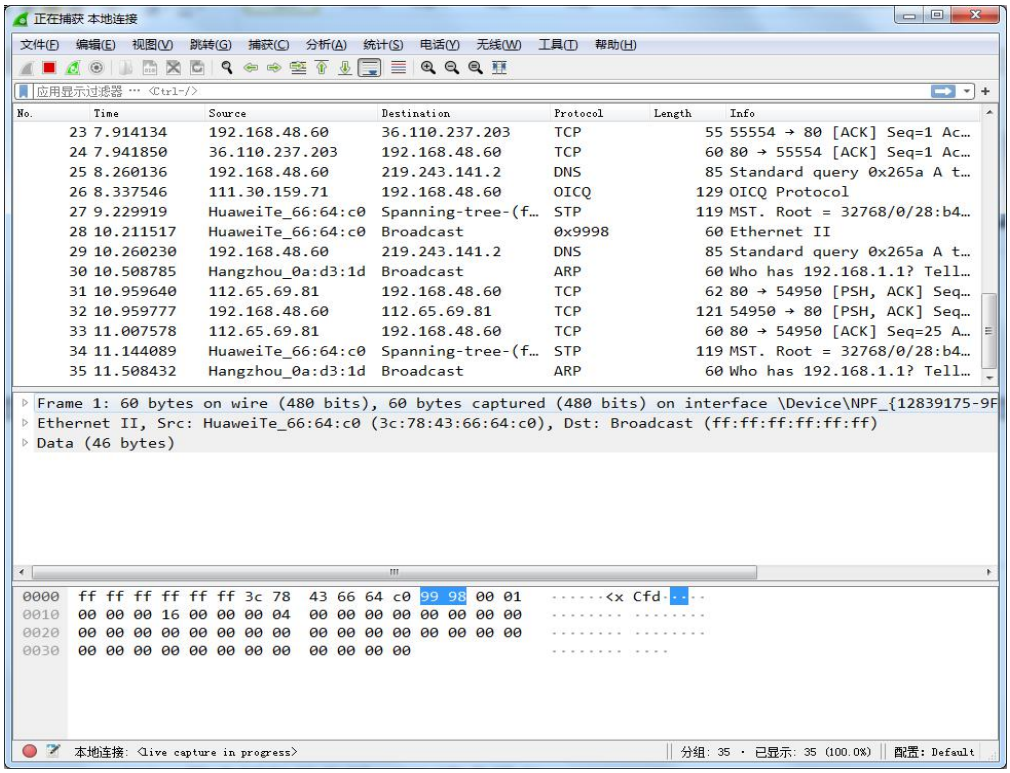
2、设置过滤器筛选出协议类别为 TCP、UDP、DNS、HTTP、ARP、ICMP 的报文，学会简单查看分析数据包内容。

三、实验步骤

1、打开 Wireshark 抓包工具，选择网卡，双击进入主界面。

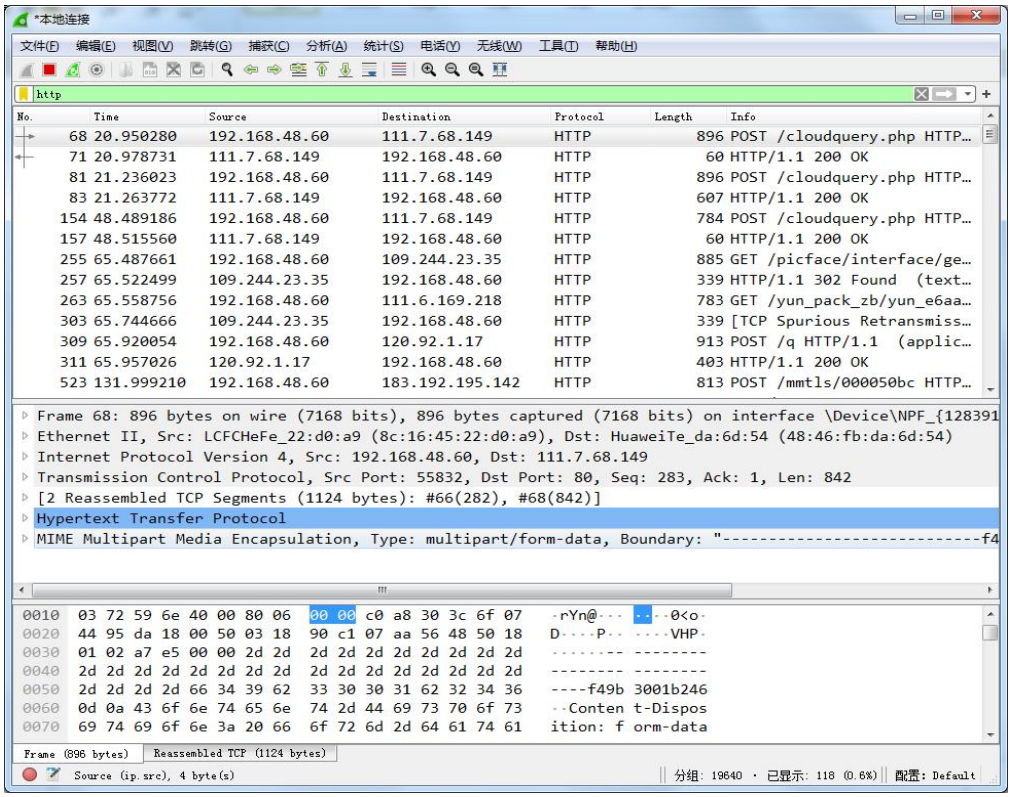


2、启动捕获，截取某个时刻抓到的网络数据包。



3、捕获到的数据包列表中有很多无效数据。通过添加过滤器筛选，可以筛选指定数据包。

（1）协议过滤。在显示过滤器中输入协议名称，例如 http，筛选出所有捕获到的 http 报文。



1）第一行，帧 Frame 序号

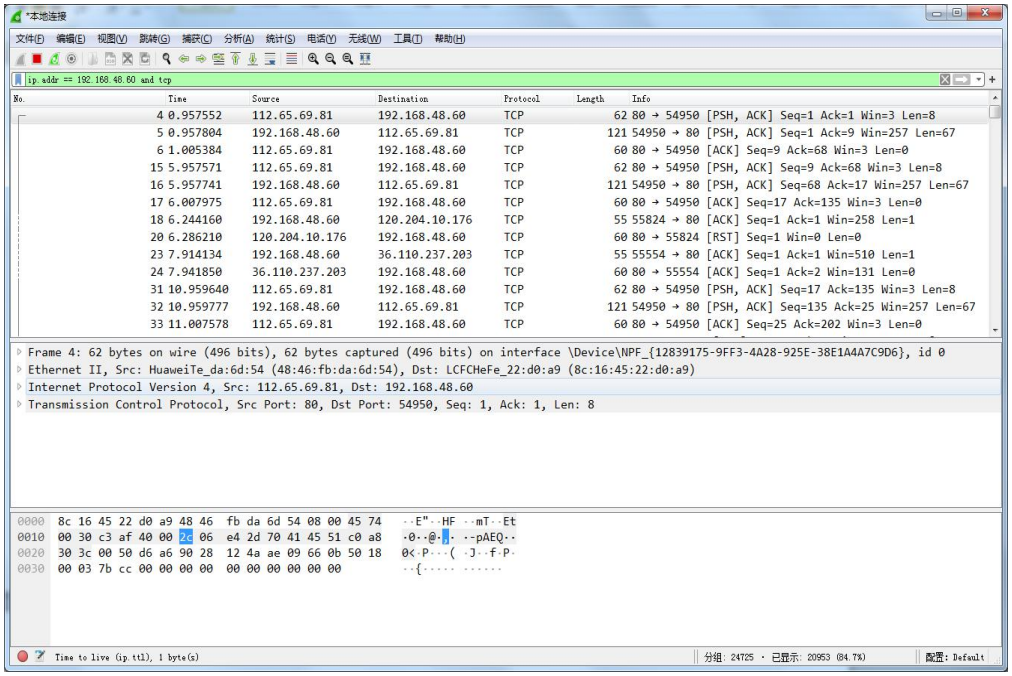
2）第二行，以太网，数据链路层

3）第三行，IPv4 协议，网络层

4）第四行，传输控制层协议

5）第五行，超文本传输协议，应用层

（2）IP 过滤。在筛选过滤框中输入 ip.addr == 192.168.48.60 and tcp。



1）第一行，帧 Frame 序号

2）第二行，以太网，数据链路层

3）第三行，IPv4 协议，网络层

4）第四行，传输控制层协议

（3）通过设置过滤规则，筛选出本机 IP 地址并且是 DNS 协议的数据包。

（4）命令窗口中输入 ping www.baidu.com，在过滤栏设置过滤条件进行数据包列表过滤，ip.addr == x.x.x.x and icmp。

四、实验总结

实验十 访问控制列表 ACL

一、实验目的

访问控制列表：为了过滤数据包，需要配置一些规则，规定什么样的数据包可以通过，什么样的数据包不能通过。ACL 根据 IP 报文的协议号、源地址、目的地址、源端口和目的端口等信息来过滤数据包。

1、掌握访问列表顺序配置的重要性。

2、掌握扩展访问列表的配置方法。

3、理解扩展访问列表丰富的过滤条件。

二、应用场景

1、某些安全性要求比较高的主机或网络需要进行访问控制。

2、其它的很多应用都可以使用访问控制列表提供操作条件。

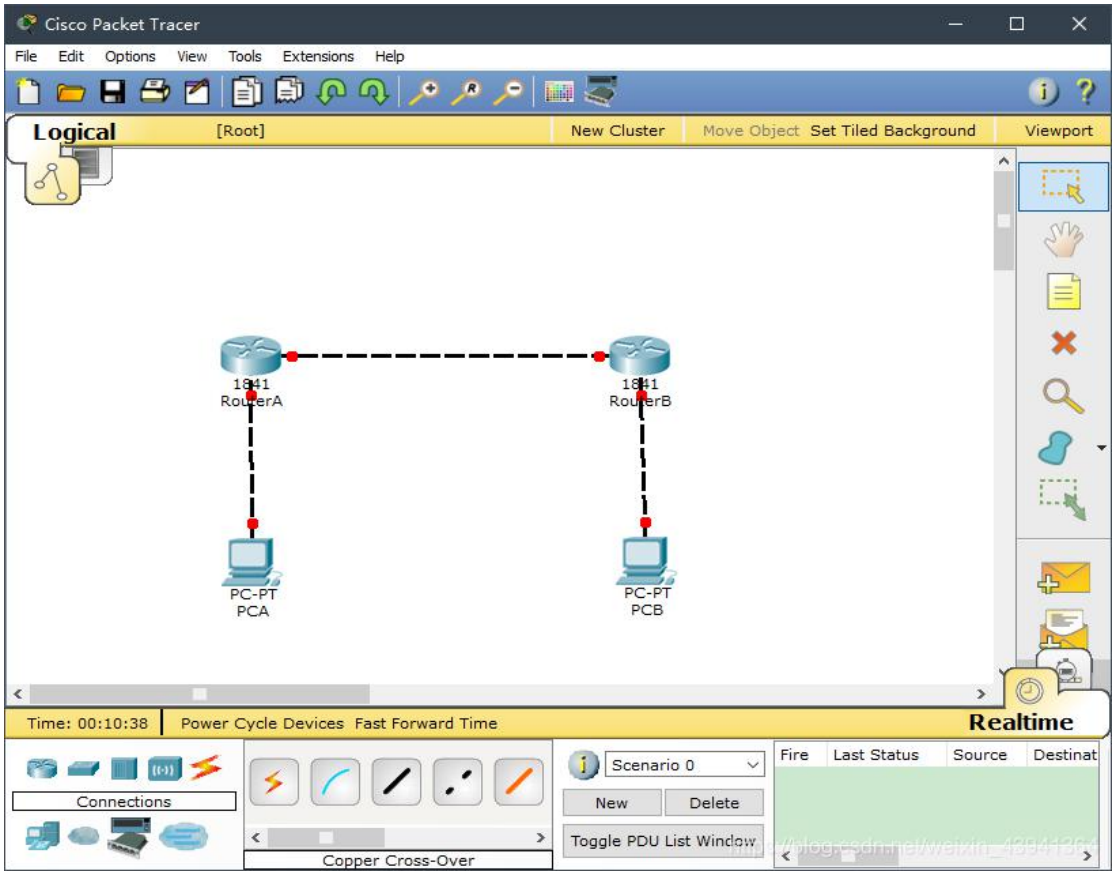
3、可以针对目标 IP 地址进行控制。

4、可以针对某些协议进行控制。

三、实验环境

1、Cisco 路由器 2 台、PC 机 2 台、交叉线 3 条。

四、实验拓扑结构图

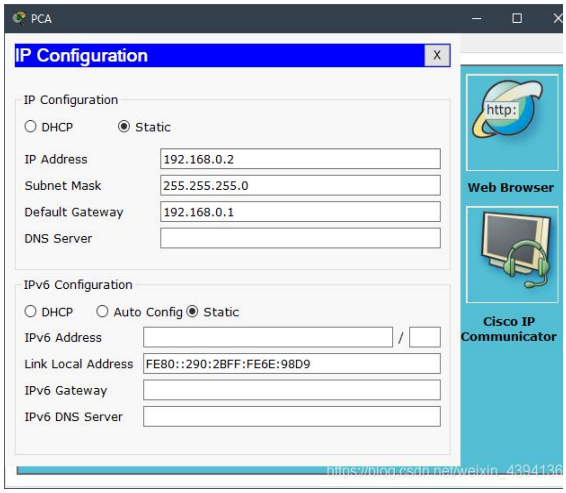


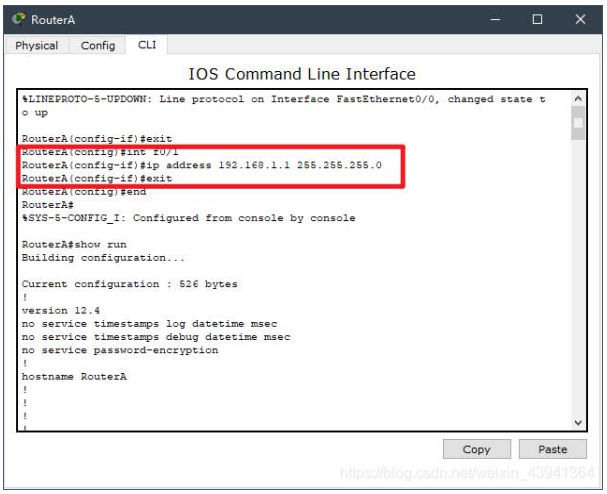
五、实验配置

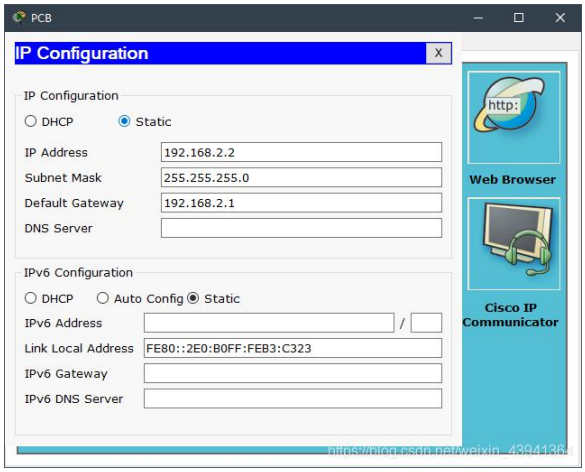
设备	IP 地址	
	接口 F0/0	接口 F0/1
Router-A	192.168.0.1/24	192.168.1.1/24
Router-B	192.168.2.1/24	192.168.1.2/24
设备	IP 地址	网关
PC-A	192.168.0.2/24	192.168.0.1
PC-B	192.168.2.2/24	192.168.2.1

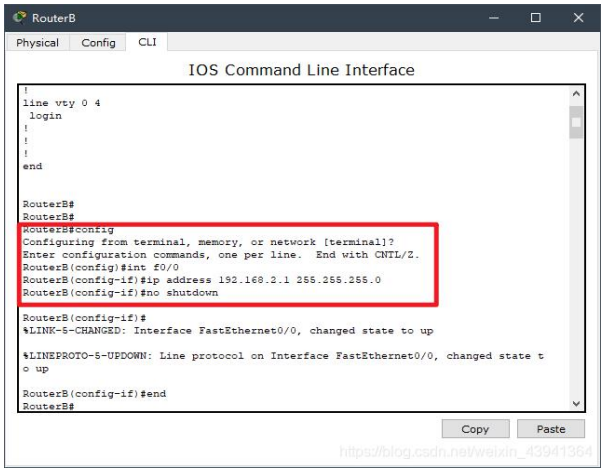
六、实验步骤

1、配置各个设备的名称及 IP 地址并测试连接情况。注：设置完路由器的接口 IP 地址不要忘记执行 no shutdown 命令。









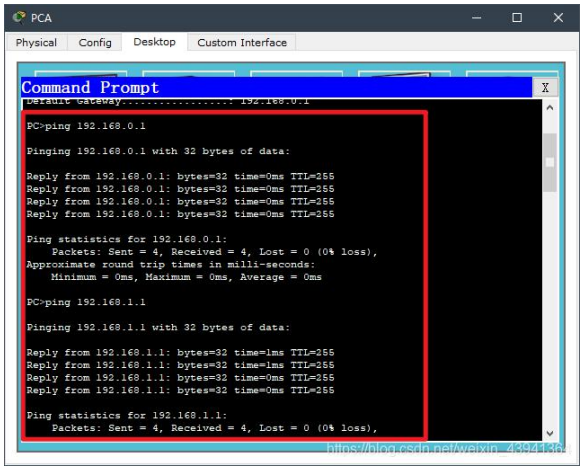
2、分别用 PC-A ping 其它各个设备并记结果。

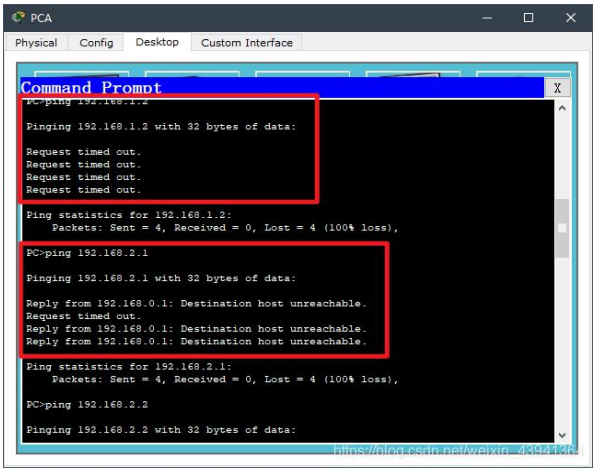
（1）首先是 ping 路由器 A 的两个接口，结果：都可以 ping 通。

（2）其次是 ping 路由器 B 的两个接口，结果：ping 不通。

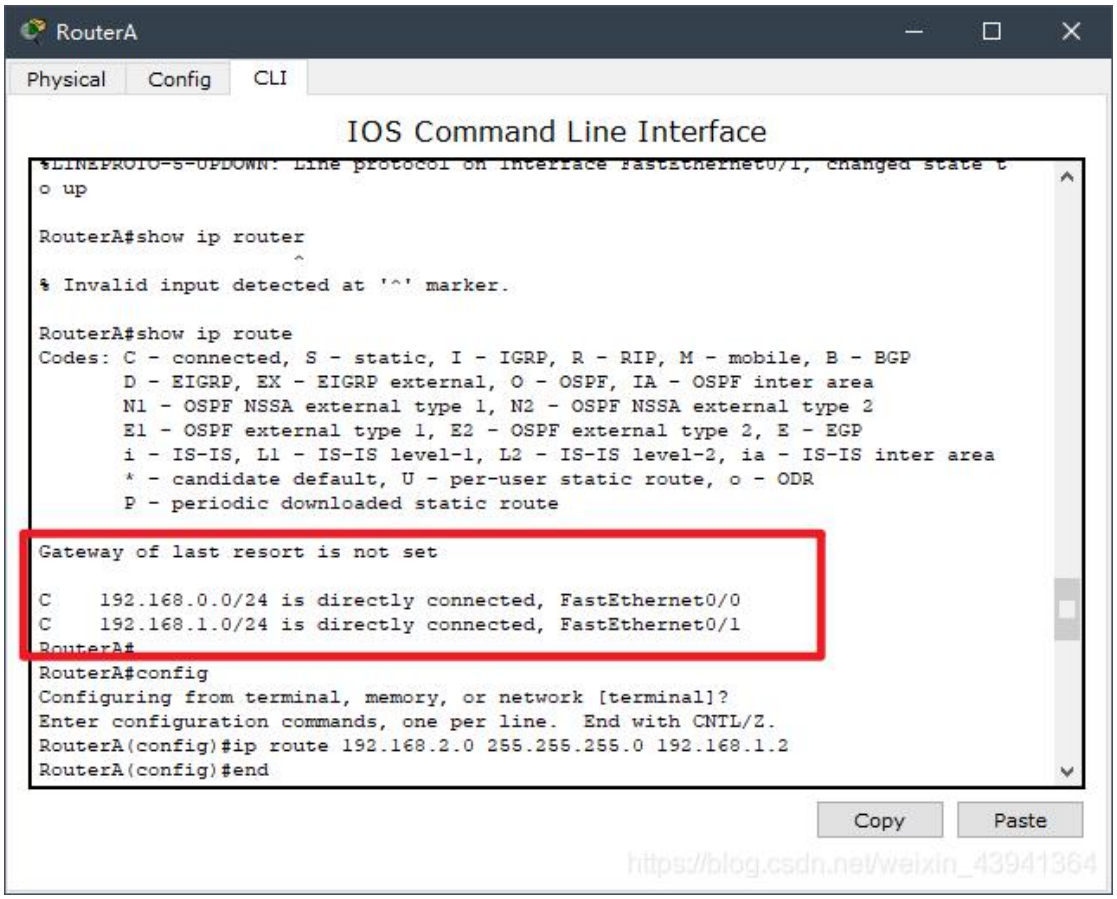
因

（3）因为路由器只能连接相邻的两个网络，它的路由表中只有和它相邻的两个网络的



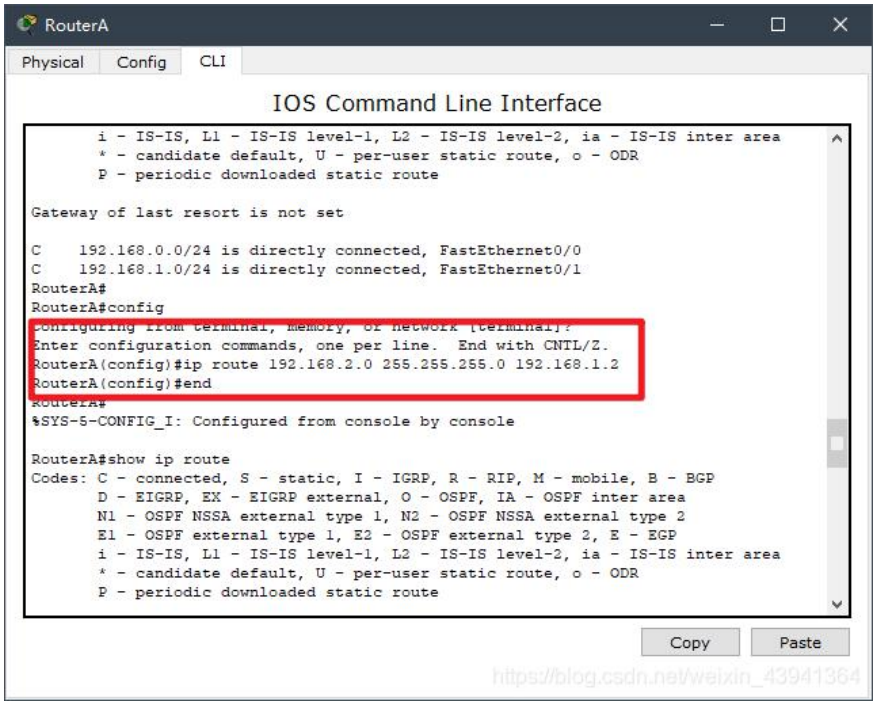


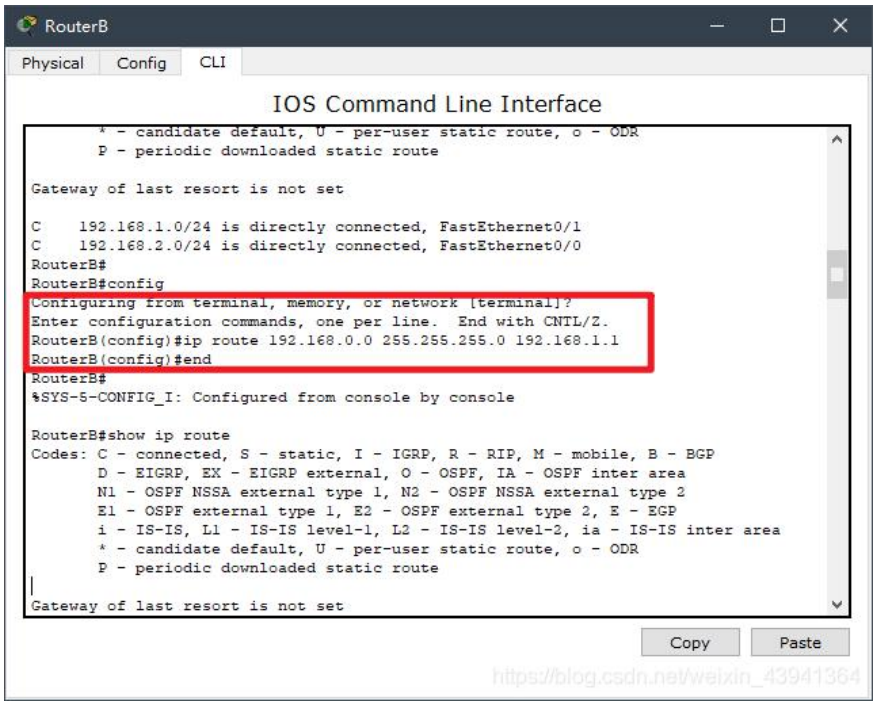
路由信息，所以一旦跨越两个网络以上就不可达了，可以先看一下现在的路由表：



3、配置静态路由

（1）配置路由器 A 和 B 的路由信息。

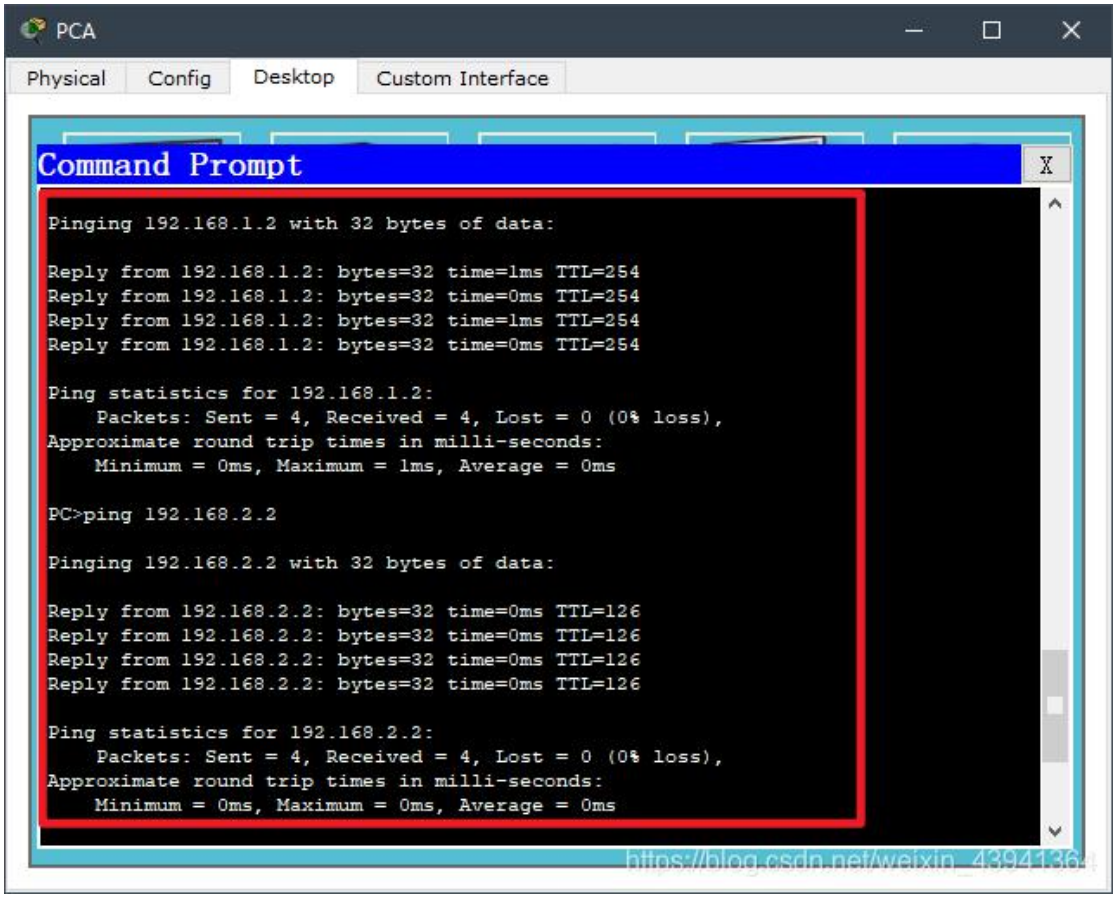




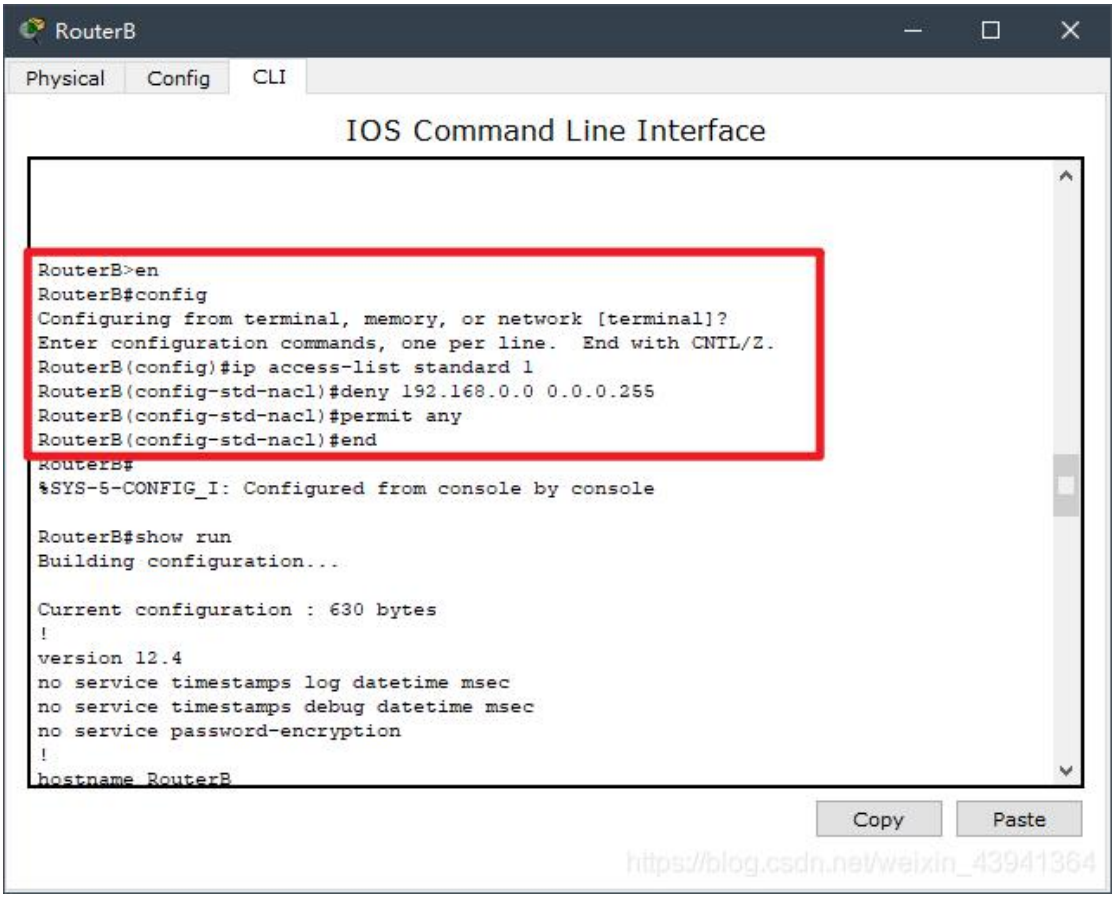
（2）查看路由表信息，命令行输入：show ip route。

（3）发现路由器 A 和 B 都多了一条静态路由记录信息。

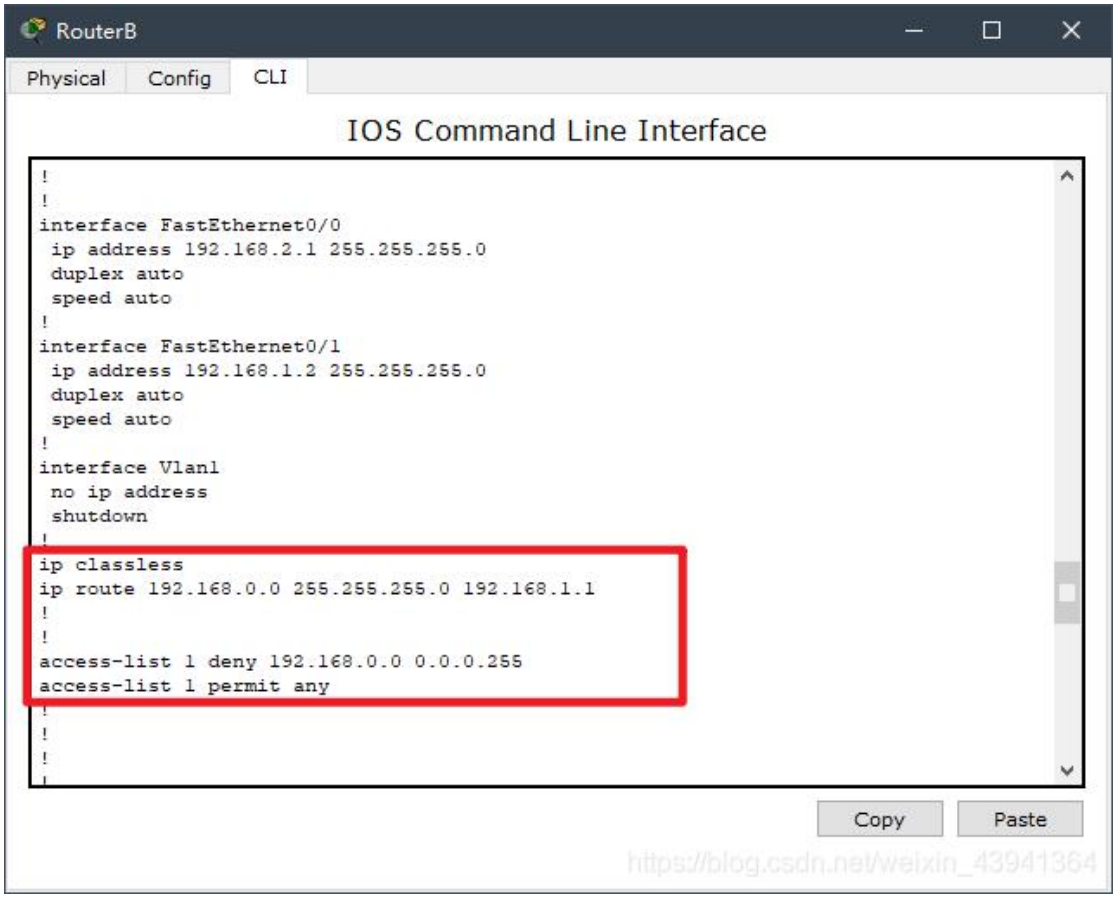
4、检查 PC-A 能否与 PC-B 通讯？



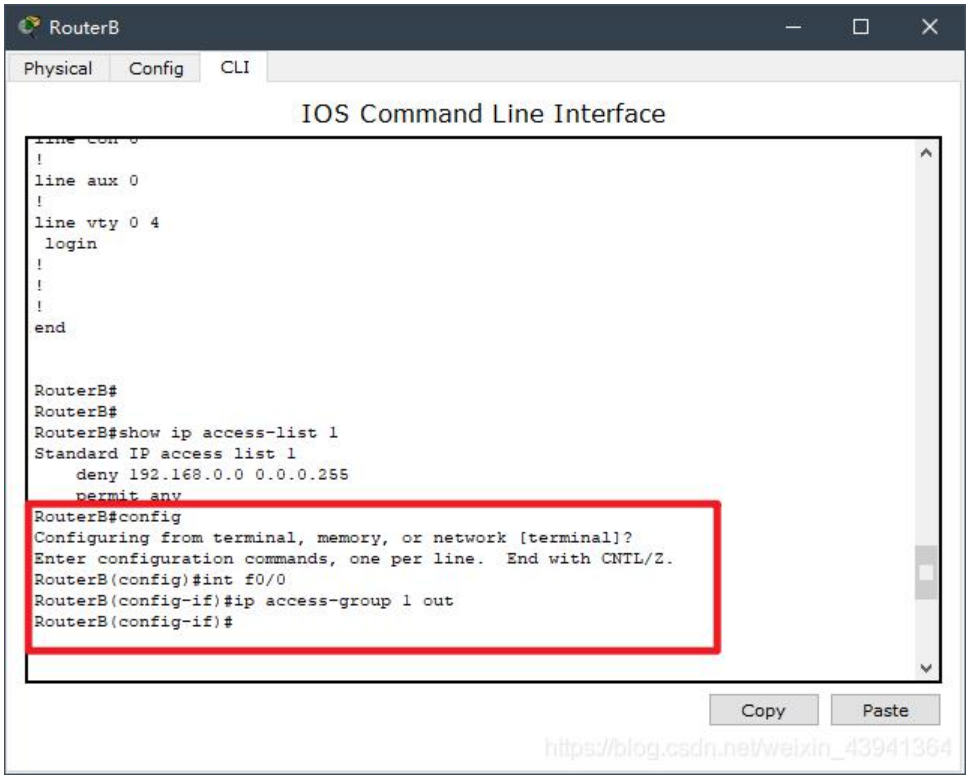
5、配置访问控制器列表禁止 PC-A 所在的网段对 PC-B 的访问。



6、验证访问控制列表是否配置成功，命令行输入：show run 。



7、此时，再用 ping 命令测试 PC-A 和 PC-B 的连通性，发现 PC-A 可以 ping 通 PC-B。因为：仅仅是配置了访问控制列表，但是并没有指定给哪一个端口。



8、ACL 应用于指定接口后，就生效了，即：PC-A　ping 不通 PC-B。注：取消访问控制列表的命令是 no ip access-group 1 out。

四、实验总结

实验十二 利用 Wireshark 分析 TCP 三次握手

一、实验目的

1．学习 TCP 报文的格式。

2．学习使用 Wireshark 对 TCP 协议进行分析。

二、实验准备

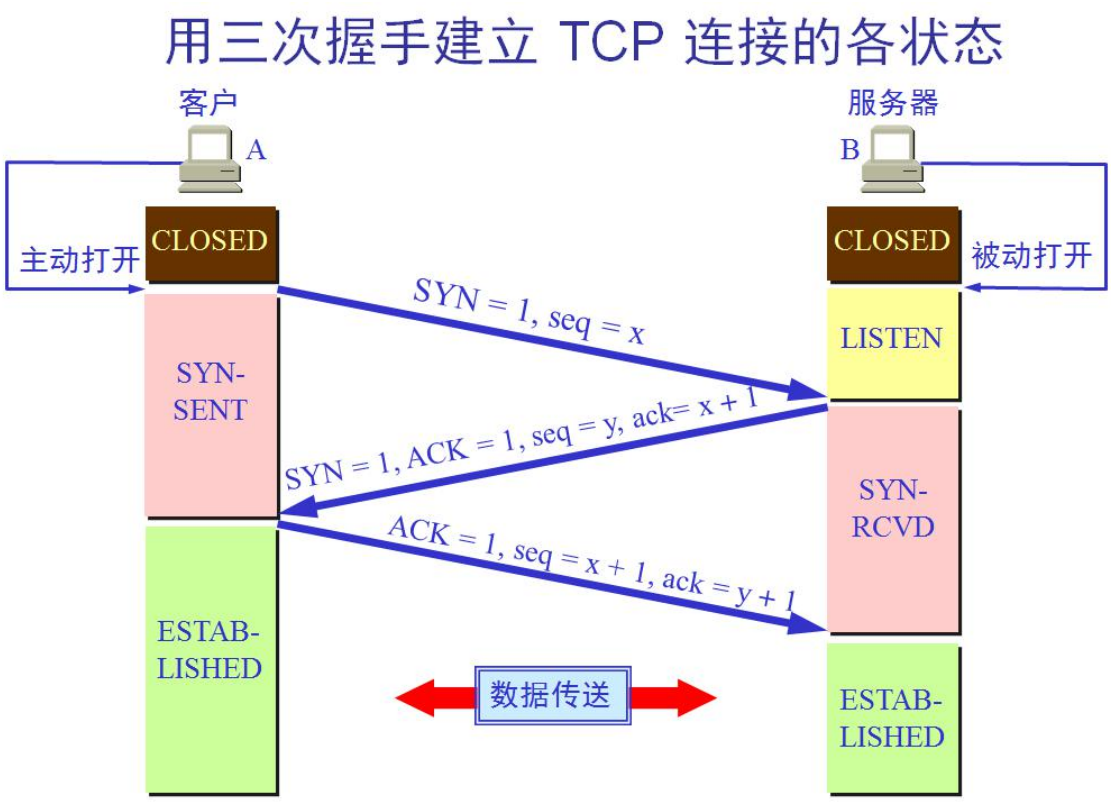
TCP 是一种面向连接（连接导向）的、可靠的基于字节流的传输层通信协议。TCP 将用户数据打包成报文段，它发送后启动一个定时器，另一端收到的数据进行确认、对失序的数据重新排序、丢弃重复数据。

抓包分析主要关注的是： SYN、ACK、Seq

确认比特 ACK： ACK = 1 时代表这是一个确认的 TCP 包，取值 0 则不是确认包。

同步比特 SYN： 在建立连接是用来同步序号。SYN=1， ACK=0 表示一个连接请求报文段。SYN=1，ACK=1 表示同意建立连接。

<table>
<tr><td colspan="5">源 端 口</td><td colspan="4">目 的 端 口</td></tr>
<tr><td colspan="9">序 号</td></tr>
<tr><td colspan="9">确 认 号</td></tr>
<tr><td>数据
偏移</td><td colspan="3">保 留</td><td>URG</td><td>ACK</td><td>PSH</td><td>RST</td><td>SYN</td><td>FIN</td><td colspan="3">窗 口</td></tr>
<tr><td colspan="5">检 验 和</td><td colspan="4">紧 急 指 针</td></tr>
<tr><td colspan="7">选 项 （长 度 可 变）</td><td colspan="2">填 充</td></tr>
</table>

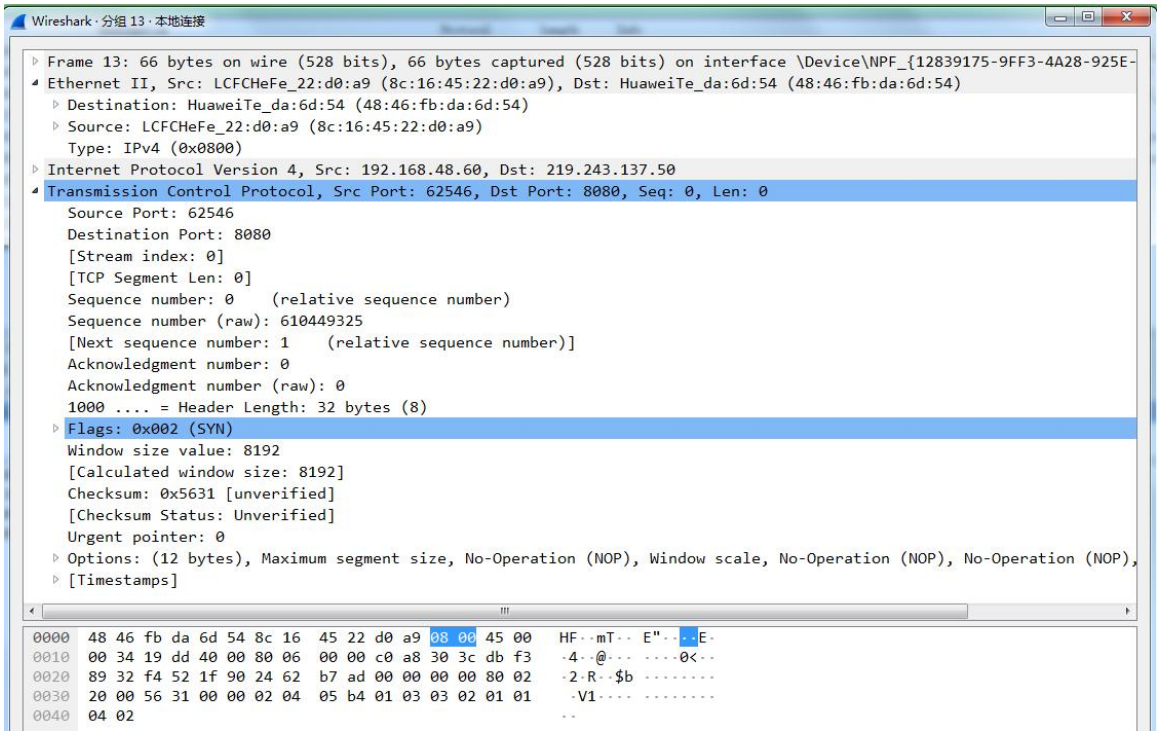


三、实验步骤

1、打开 Wireshark，双击本地连接，开始抓包。

2、打开浏览器，浏览任意一个网页。

3、粘贴捕获的 TCP 报文。



4、 TCP 报文格式分析

Source Port（源端口）：_____

Destination Port(目的端口)：_____

Sequence number(序号) ：_____

Acknowledgement number（确认号）：_____

Flags（标志位）：_____

Windows size value（窗口）：_____

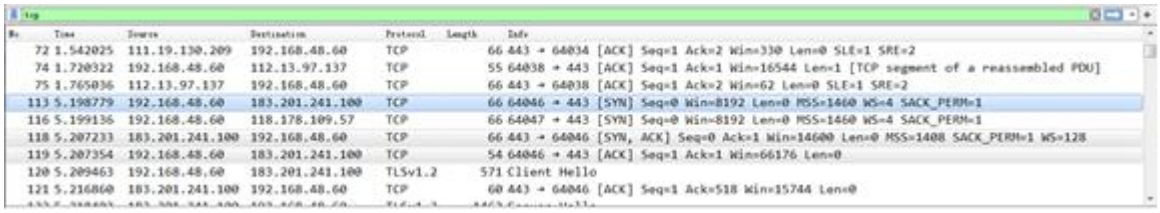
Checksum（检验和）：_____

Urgent point（紧急指针）：_____

Options（选项）：_____

填充（可选）：_____

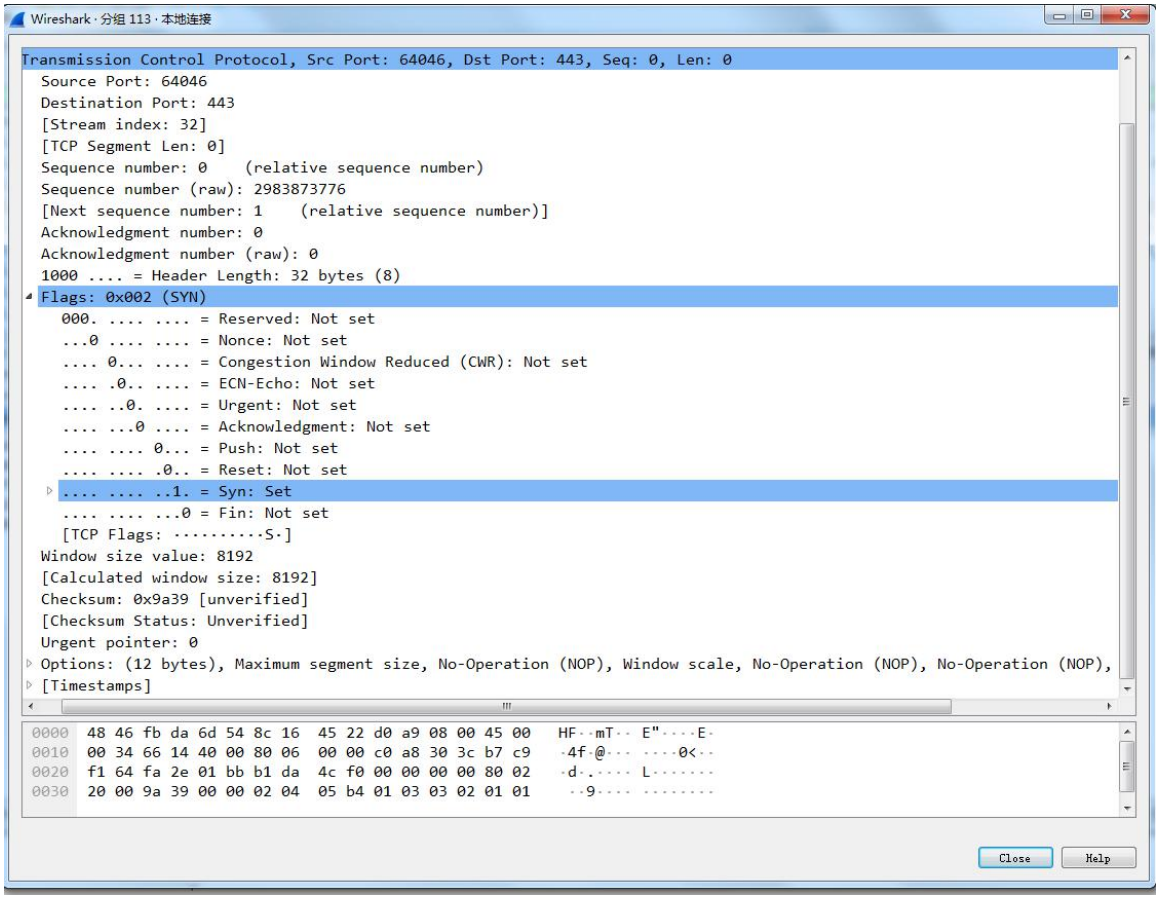
5、查找建立连接时的三次握手，截图如下。



No.	Time	Source	Destination	Protocol	Length	Info
72	1.542025	111.19.130.209	192.168.48.60	TCP	66	443 → 64034 [ACK] Seq=1 Ack=2 Win=330 Len=0 SLE=1 SRE=2
74	1.720322	192.168.48.60	112.13.97.137	TCP	55	64038 → 443 [ACK] Seq=1 Ack=1 Win=16544 Len=1 [TCP segment of a reassembled PDU]
75	1.765036	112.13.97.137	192.168.48.60	TCP	66	443 → 64038 [ACK] Seq=1 Ack=2 Win=62 Len=0 SLE=1 SRE=2
113	5.198779	192.168.48.60	183.201.241.100	TCP	66	64046 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
116	5.199136	192.168.48.60	118.178.109.57	TCP	66	64047 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
118	5.207233	183.201.241.100	192.168.48.60	TCP	66	443 → 64046 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1408 SACK_PERM=1 WS=128
119	5.207354	192.168.48.60	183.201.241.100	TCP	54	64046 → 443 [ACK] Seq=1 Ack=1 Win=66176 Len=0
120	5.209463	192.168.48.60	183.201.241.100	TLSv1.2	571	Client Hello
121	5.216860	183.201.241.100	192.168.48.60	TCP	60	443 → 64046 [ACK] Seq=1 Ack=518 Win=15744 Len=0

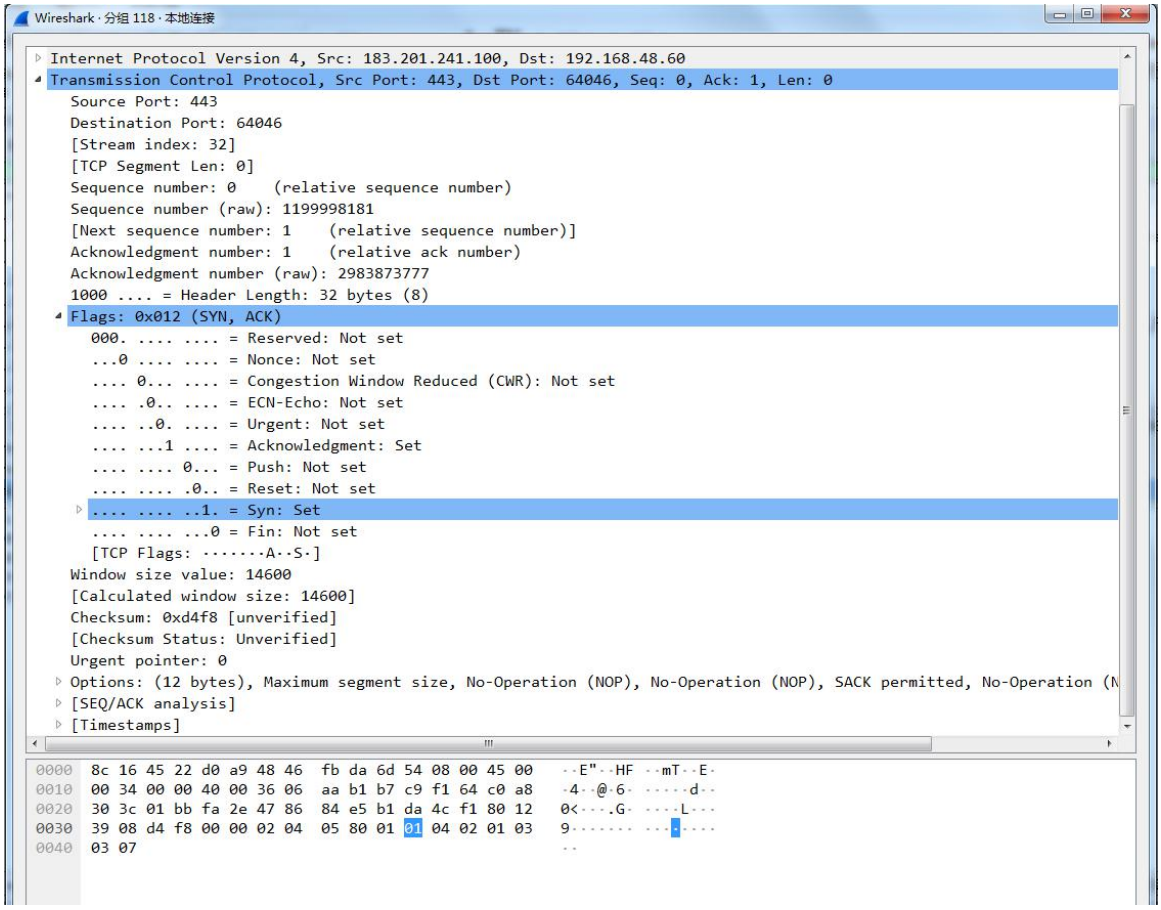
（1） 第一次握手

建立连接时，客户端发送 SYN 包到服务器，其中包含客户端的初始序号 Seq=x。（其中，SYN=1，ACK=0，表示这是一个 TCP 连接请求数据报文；序号 Seq=X，表明传输数据时的第一个数据字节的序号是 x）。



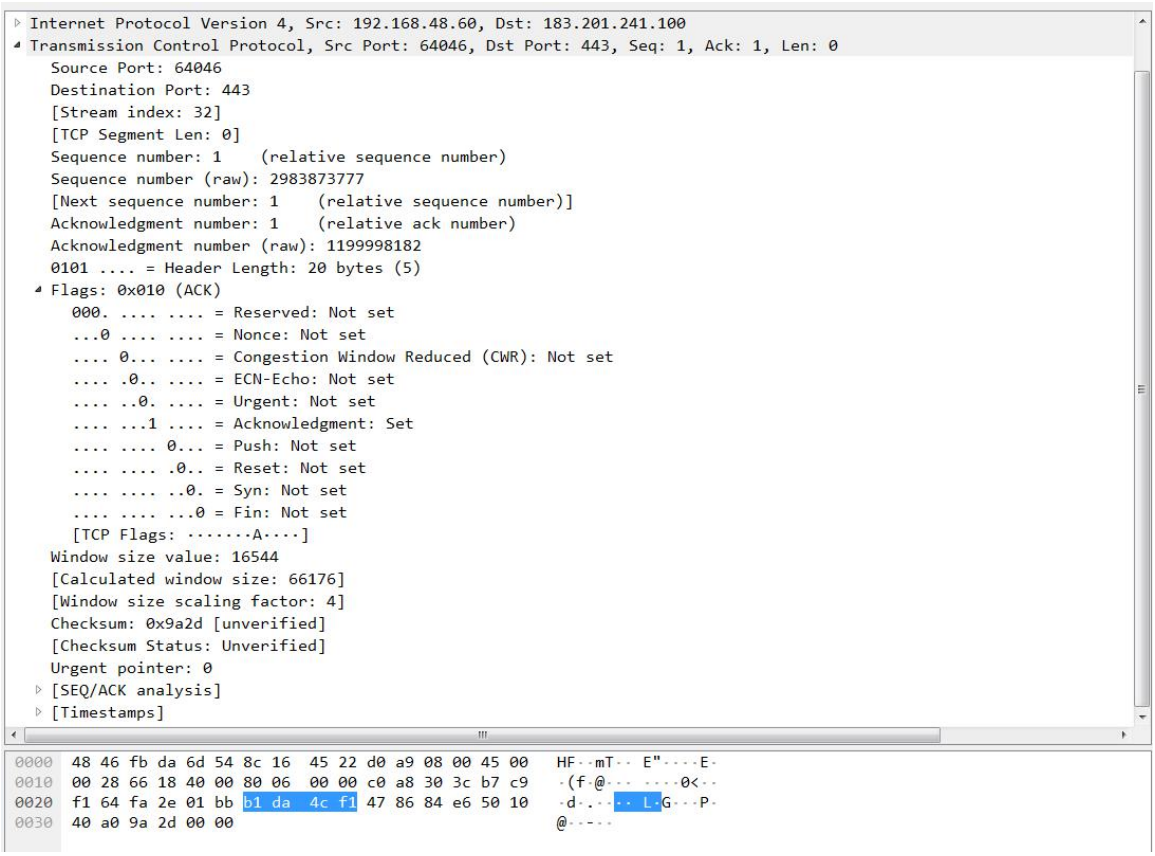
（2） 第二次握手

服务器收到请求后，必须确认客户的数据包。同时自己也发送一个 SYN 包，即 SYN+ACK 包。（标识位 SYN=1，ACK=1，表示这是一个 TCP 连接响应数据报文，并含服务端的初始序号 Seq(服务器)=Y，以及服务器对客户端初始序号的确认号 Ack(服务器)=Seq(客户端)+1=X+1）。



（3） 第三次握手

客户端收到服务器的 SYN+ACK 包，向服务器发送一个序列号(Seq=X+1)，确认号为 Ack(客户端)=Y+1，此包发送完毕，客户端和服务器进入 ESTAB_LISHED(TCP 连接成功)状态，完成三次握手。


Internet Protocol Version 4, Src: 192.168.48.60, Dst: 183.201.241.100
Transmission Control Protocol, Src Port: 64046, Dst Port: 443, Seq: 1, Ack: 1, Len: 0
Source Port: 64046
Destination Port: 443
[Stream index: 32]
[TCP Segment Len: 0]
Sequence number: 1 (relative sequence number)
Sequence number (raw): 2983873777
[Next sequence number: 1 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
Acknowledgment number (raw): 1199998182
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
.... 0... .... = Congestion Window Reduced (CWR): Not set
.... .0.. .... = ECN-Echo: Not set
.... ..0. .... = Urgent: Not set
....1 .... = Acknowledgment: Set
.... 0... = Push: Not set
.... .0.. = Reset: Not set
....0. = Syn: Not set
....0 = Fin: Not set
[TCP Flags: .....A....]
Window size value: 16544
[Calculated window size: 66176]
[Window size scaling factor: 4]
Checksum: 0x9a2d [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
[SEQ/ACK analysis]
[Timestamps]
0000 48 46 fb da 6d 54 8c 16 45 22 d0 a9 08 00 45 00
0010 00 28 66 18 40 00 80 06 00 00 c0 a8 30 3c b7 c9
0020 f1 64 fa 2e 01 bb b1 da 4c f1 47 86 84 e6 50 10
0030 40 a0 9a 2d 00 00

五、实验总结