

计算机网络实验报告

实验一：tracert的使用

1、命令详解与原理

(1) traceroute/tracert

traceroute是Linux和Mac OS等系统默认提供的路由追踪小程序，Tracert是Windows系统默认提供的路由追踪小程序。二者的功能相同，都能探测数据包从源地址到目的地址经过的路由器的IP地址。Traceroute/Tracert的实现都借助了TTL：通过向目的地址发送一系列的探测包，设置探测包的TTL初始值分别为1,2,3...，根据返回的超时通知（ICMP Time Exceeded Message）得到源地址与目的地址之间的每一跳路由信息。虽然两者输出结果一致，但在实现原理上还有着显著的差别。

(2) traceroute的实现原理

- ① 源地址发出一个UDP探测包到目的地址，并将TTL设置为1
- ② 到达路由器时，将TTL减1
- ③ 当TTL变为0时，包被丢弃，路由器向源地址发回一个ICMP超时通知（ICMP Time Exceeded Message），内含发送IP包的源地址，IP包的所有内容及路由器的IP地址
- ④ 当源地址收到该ICMP包时，显示这一跳路由信息
- ⑤ 重复1~5，并每次设置TTL加1
- ⑥ 直至目标地址收到探测数据包，并返回端口不可达通知（ICMP Port Unreachable）
- ⑦ 当源地址收到ICMP Port Unreachable包时停止traceroute

(3) tracert 实验原理

- ① 从源地址发出一个ICMP请求回显（ICMP Echo Request）数据包到目的地址，并将TTL设置为1；
- ② 到达路由器时，将TTL减1；
- ③ 当TTL变为0时，包被丢弃，路由器向源地址发回一个ICMP超时通知（ICMP Time Exceeded Message），内含发送IP包的源地址，IP包的所有内容及路由器的IP地址；
- ④ 当源地址收到该ICMP包时，显示这一跳路由信息；
- ⑤ 重复1~5，并每次设置TTL加1；
- ⑥ 直至目标地址收到探测数据包，并返回ICMP回应答复（ICMPEcho Reply）；
- ⑦ 当源地址收到ICMP Echo Reply包时停止tracert。

2、实验内容

(1) 实验内容

序号	实验项目	学时	实验目的及主要内容	实验类型	实验教学目标
1	traceroute	2	1、实验目的：熟悉traceroute的使用2、实验内容：用traceroute测量到163网站（ www.163.com ）和到微软公司（ www.microsoft.com ）网站的路径。分析测量结果。的路径结构信息。	基本验证	目标3

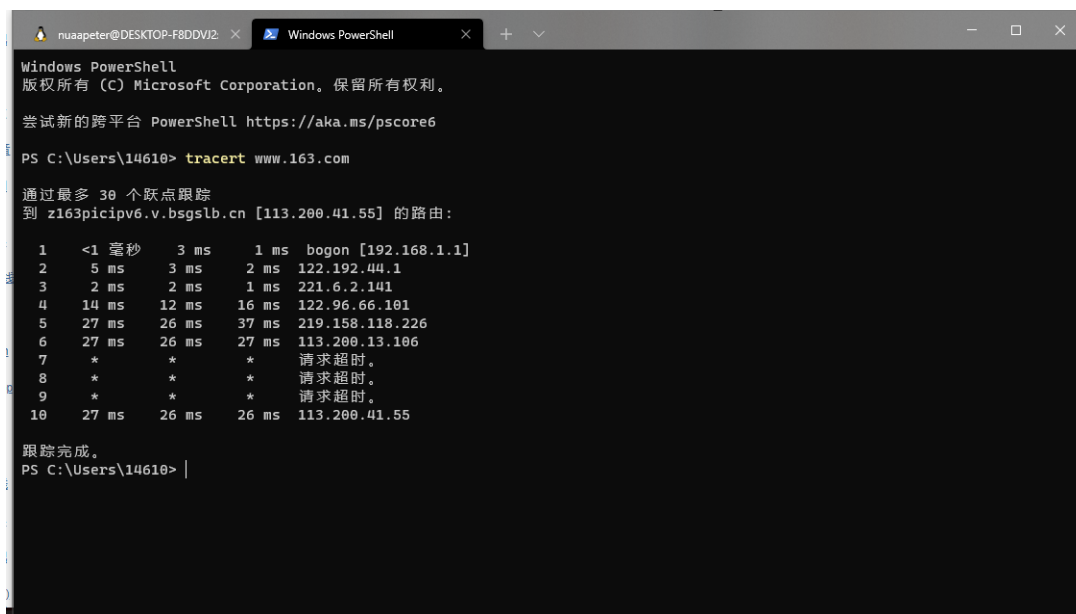
(2) tracert 命令格式

`tracert[-d][-hmaximum_hops][-j host - list][-w timeout][-R][-S srcaddr][-4][-6]target_name`

- ①、-d
表示不将地址解析成主机名。
- ②、-h maximum_hops
表示搜索目标的最大跃点数。
- ③、-j host-list
表示与主机列表一起的松散源路由（仅适用于IPv4）。
- ④、-w timeout
表示等待每个回复的超时间（以毫秒为单位）。
- ⑤、-R
表示跟踪往返行程路径（仅适用于IPv6）。
- ⑥、-S srcaddr
表示要使用的源地址（仅适用于IPv6）。
- ⑦、-4和-6
表示强制使用IPv4或者IPv6。
- ⑧、target_name
表示目标主机的名称或者IP地址。

3、实验结果分析

(1) 测试到www.163.com的路径



```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\14610> tracert www.163.com

通过最多 30 个跃点跟踪
到 z163picipv6.v.bsgslb.cn [113.200.41.55] 的路由:

  1  <1 毫秒    3 ms    1 ms  bogon [192.168.1.1]
  2   5 ms     3 ms    2 ms  122.192.44.1
  3   2 ms     2 ms    1 ms  221.6.2.141
  4  14 ms    12 ms   16 ms  122.96.66.101
  5  27 ms    26 ms   37 ms  219.158.118.226
  6  27 ms    26 ms   27 ms  113.200.13.106
  7   *         *      *      请求超时。
  8   *         *      *      请求超时。
  9   *         *      *      请求超时。
 10  27 ms    26 ms   26 ms  113.200.41.55

跟踪完成。
PS C:\Users\14610>
```

分析：

- tracert命令用于确定 IP数据包访问目标所采取的路径，显示从本地到目标网站所在网络服务器的一系列网络节点的访问速度，最多支持显示30个网络节点
- 最左侧的，1，2，3，4~10，表明在我使用的宽带上，经过9（不算自己本地的）个路由节点，可以到达www.163.com的服务
- 中间的三列，单位是ms，表示我们连接到每个路由节点的速度，返回速度和多次链接反馈的平均值
- 后面的IP，就是每个路由节点对应的IP
- 7,8,9 中出现了“*”号，表示这个路由节点和当前我们使用的宽带，是无法联通的，原因有：特意在路由上做了过滤限制，或者确实是路由的问题等，需要具体问题具体分析
- 综上 经过9个路由结点（不算自己的路由），可以到达www.163.com的服务

(2) 测试到www.microsoft.com的路径

```
PS C:\Users\14610> tracert www.microsoft.com

通过最多 30 个跃点跟踪
到 e13678.ca2.s.tl88.net [218.58.101.49] 的路由:

  1  <1 毫秒    4 ms    <1 毫秒 bogon [192.168.1.1]
  2  3 ms      9 ms    8 ms   122.192.44.1
  3  3 ms      2 ms    3 ms   122.96.66.73
  4  14 ms     13 ms   11 ms   122.96.66.97
  5  19 ms     18 ms   21 ms   219.158.17.33
  6  *         *      17 ms   60.210.14.194
  7  26 ms     27 ms   29 ms   221.1.4.238
  8  36 ms     22 ms   25 ms   61.133.95.166
  9  18 ms     17 ms   17 ms   218.58.101.49

跟踪完成。
PS C:\Users\14610> |
```

分析

- 具体分析同上，经过8个路由结点（不算自己的路由）,可以到达www.microsoft.com

4、本实验小结

traceroute/tracert路由追踪程序是用来追踪数据包到达网络主机所经过的路由信息的重要工具，虽然路由追踪效果一致，但实现原理略有不同：前者借助UDP协议，后者借助ICMP协议。

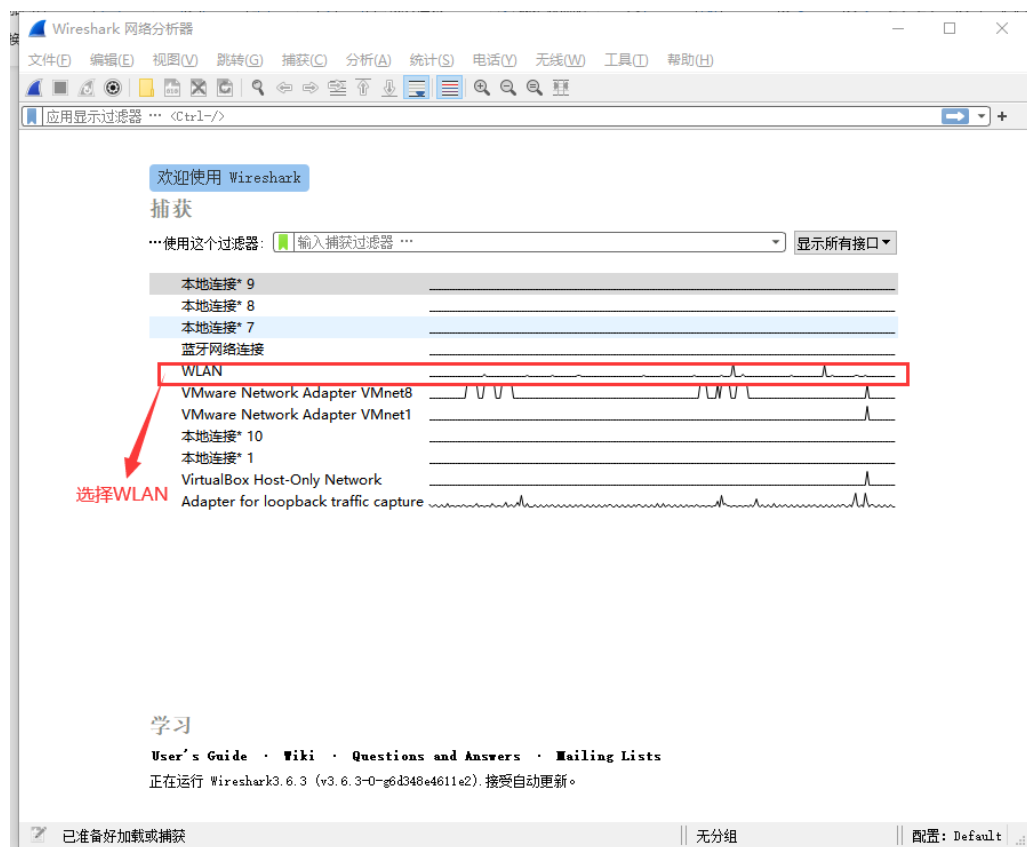
实验二：wireshark的使用

1、实验内容

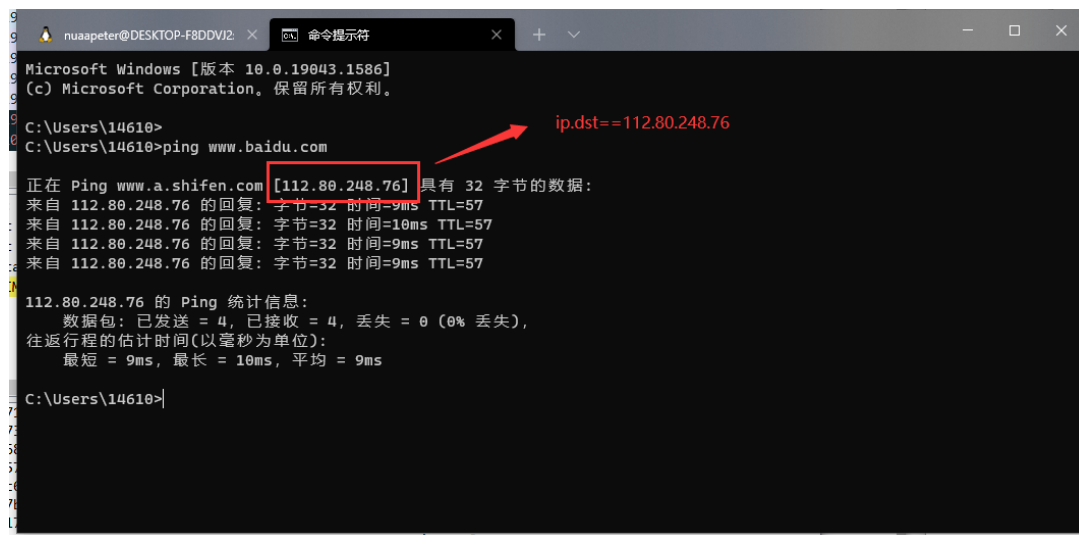
序号	实验项目	学时	实验目的及主要内容	实验类型	实验教学目标
2	wireshark	4	1、实验目的：熟悉wireshark的使用2、实验内容：下载安装wireshark软件，设置捕获条件，用wireshark捕获数据包，对以太网帧和IP数据包进行分析	基本验证	目标1

2、实验过程

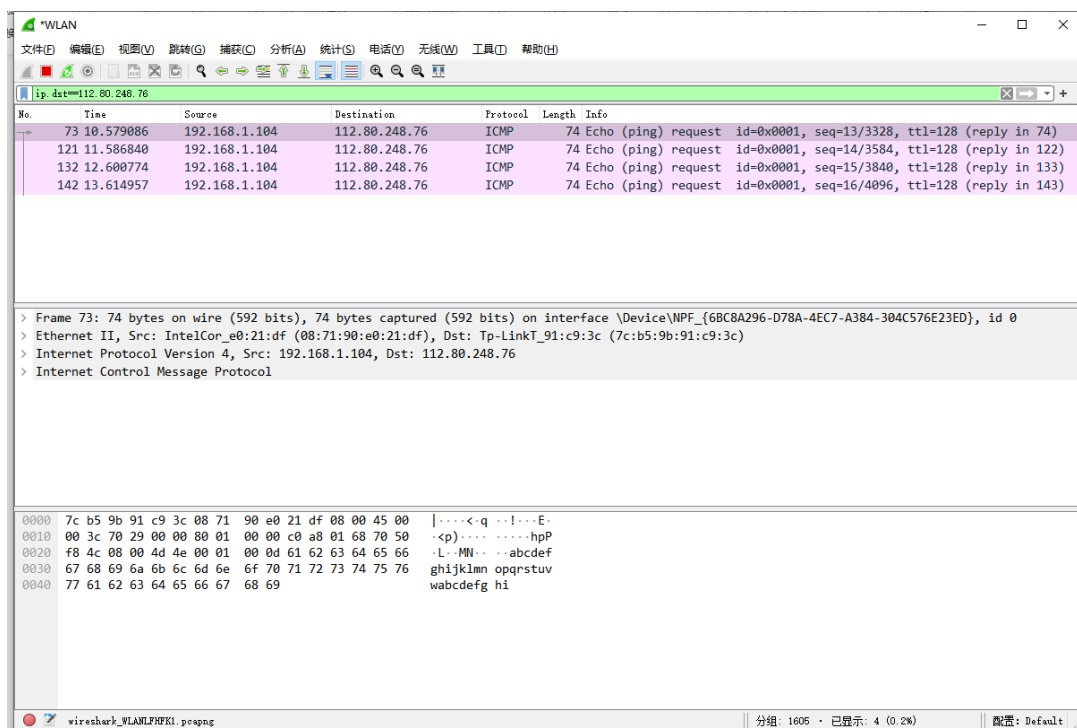
- ① 双击打开wireshark
- ② 选择WLAN，并点击左上角图标开始捕获



- ③ 打开 cmd 输入 ping www.baidu.com 观察 wireshark 里面的变化



- ④ 设置过滤器内的捕获条件为 ip.dst==112.80.248.76



- ⑤ 在框内可以得到实验结果

```

Ethernet II, Src: IntelCor_e0:21:df (08:71:90:e0:21:df), Dst: Tp-LinkT_91:c9:3c (7c:b5:9b:91:c9:3c)
  Destination: Tp-LinkT_91:c9:3c (7c:b5:9b:91:c9:3c)
  Source: IntelCor_e0:21:df (08:71:90:e0:21:df)
  Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.1.104, Dst: 112.80.248.76
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 60
  Identification: 0x7029 (28713)
  Flags: 0x00
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 128
  Protocol: ICMP (1)
  Header Checksum: 0x0000 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 192.168.1.104
  Destination Address: 112.80.248.76

```

3、实验结果分析

(1)以太网帧分析

MAC帧 = 6字节源mac地址 + 6字节目标mac地址 + 2字节类型 + 4字节帧检验序列FCS + 数据部分 (2)

No.	Time	Source	Destination	Protocol	Length	Info
73	10.579086	192.168.1.104	112.80.248.76	ICMP	74	Echo (ping) request id=0x0001, seq=13/3328, ttl=128 (reply in 74)
121	11.586840	192.168.1.104	112.80.248.76	ICMP	74	Echo (ping) request id=0x0001, seq=14/3584, ttl=128 (reply in 122)
132	12.600774	192.168.1.104	112.80.248.76	ICMP	74	Echo (ping) request id=0x0001, seq=15/3840, ttl=128 (reply in 133)

```

Ethernet II, Src: IntelCor_e0:21:df (08:71:90:e0:21:df), Dst: Tp-LinkT_91:c9:3c (7c:b5:9b:91:c9:3c)
  Destination: Tp-LinkT_91:c9:3c (7c:b5:9b:91:c9:3c)
    Address: Tp-LinkT_91:c9:3c (7c:b5:9b:91:c9:3c)
    .... 00. .... = LG bit: Globally unique address (factory default)
    .... 00. .... = IG bit: Individual address (unicast)
  Source: IntelCor_e0:21:df (08:71:90:e0:21:df)
    Address: IntelCor_e0:21:df (08:71:90:e0:21:df)
    .... 00. .... = LG bit: Globally unique address (factory default)
    .... 00. .... = IG bit: Individual address (unicast)
  Type: IPv4 (0x0800)

```

源地址 (Source) 为本机的MAC地址，从上图中可以看到的是192.168.1.104，在cmd中输入 ipconfig 验证，如下图，可以看出是匹配的

```
无线局域网适配器 WLAN:

    连接特定的 DNS 后缀 . . . . . : 
    本地链接 IPv6 地址. . . . . : fe80::fd99:789f:4a81:6b4f%10
    IPv4 地址 . . . . . : 192.168.1.104
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . : 192.168.1.1
```

目的地址 (Destination) 是112.90.248.76, 即最终的百度的服务器的地址

类型 (Type) 为IPv4类型即数据包类型为IPv4

(2) IP数据报分析

```
Internet Protocol Version 4, Src: 192.168.1.104, Dst: 112.80.248.76
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 60
    Identification: 0x7029 (28713)
  > Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: ICMP (1)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.1.104
    Destination Address: 112.80.248.76
```

根据上图我们分析出

- 版本 (Internet Protocol Version) : IPv4
- 首部长度 (Header Length) : 20 byte
- 服务类型(区分服务): DSCP: CS0, ECN: Not-ECT
- 总长度 (Total Length) : 60字节
- 标识 (Identification) : 0x7029 (28713)
- 标志 (Flags) : 0x00
- 片偏移 (Fragment Offset) : 0字节
- 生存时间 (TTL) : 128跳
- 协议 (Protocol) : ICMP(1)
- 首部校验和 (Header Checksum) : 0x0000
- 源地址 (Source Address) : 192.168.1.104
- 目的地址 (Destination Address) : 112.80.248.76