计算机网络

实验报告

（2022学年秋季学期）

教学班级	计科二班	专业（方向）	计算机科学与技术
学号	20337263	姓名	俞泽斌

一、 实验题目

利用 wireshark 分析 ICMP 及相关 IP 数据包服务

1）在实验机器终端启动 wireshark 抓包,设置过滤显示 IP，ICMP，UDP 和 TCP 相关的信息；
2） 运行命令 ping 命令
ping www.ucdavis.edu
3） 截图显示网络层 IP、ICMP 协议,传输层协议的活动；观察期间数据传输；

利用 wireshark 分析 ICMP/UDP 及相关 IP 服务

1）在实验机器终端启动 wireshark 抓包,设置过滤显示 IP，ICMP，UDP 相关的信息；
2） 运行命令 traceroute 命令（windows 的是 tracert）
Tracert www.ucdavis.edu
3） 截图显示网络层 IP、ICMP 协议，传输层及 UDP 相关的信息；观察期间数据传输；
4） 分析并解释以上实验结果

二、 实验步骤

利用 wireshark 分析 ICMP 及相关 IP 数据包服务

首先启动wireshark抓包，并,设置过滤显示 IP，ICMP，UDP 和 TCP

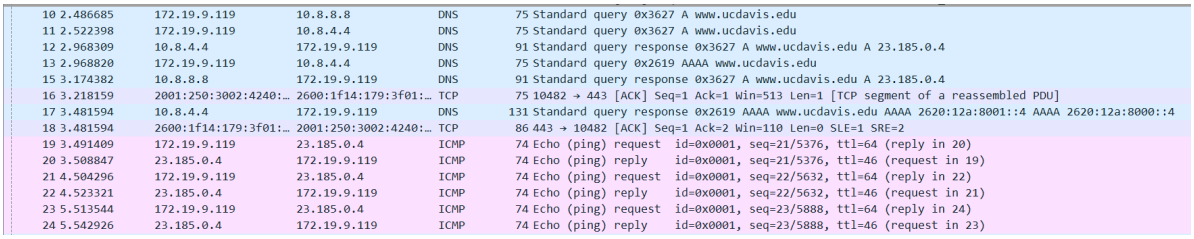
将过滤器设置为

```
icmp ||udp ||tcp ||ip
```

在命令行窗口运行

```
C:\Users\Aholic^y>ping www.ucdavis.edu

Pinging www.ucdavis.edu [23.185.0.4] with 32 bytes of data:
Reply from 23.185.0.4: bytes=32 time=17ms TTL=46
Reply from 23.185.0.4: bytes=32 time=19ms TTL=46
Reply from 23.185.0.4: bytes=32 time=29ms TTL=46
Reply from 23.185.0.4: bytes=32 time=16ms TTL=46

Ping statistics for 23.185.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 16ms, Maximum = 29ms, Average = 20ms
```

主要得到报文如下



10	2.486685	172.19.9.119	10.8.8.8	DNS	75 Standard query 0x3627 A www.ucdavis.edu
11	2.522398	172.19.9.119	10.8.4.4	DNS	75 Standard query 0x3627 A www.ucdavis.edu
12	2.968309	10.8.4.4	172.19.9.119	DNS	91 Standard query response 0x3627 A www.ucdavis.edu A 23.185.0.4
13	2.968820	172.19.9.119	10.8.4.4	DNS	75 Standard query 0x2619 AAAA www.ucdavis.edu
15	3.174382	10.8.8.8	172.19.9.119	DNS	91 Standard query response 0x3627 A www.ucdavis.edu A 23.185.0.4
16	3.218159	2001:250:3002:4240:…	2600:1f14:179:3f01:…	TCP	75 10482 → 443 [ACK] Seq=1 Ack=1 Win=513 Len=1 [TCP segment of a reassembled PDU]
17	3.481594	10.8.4.4	172.19.9.119	DNS	131 Standard query response 0x2619 AAAA www.ucdavis.edu AAAA 2620:12a:8001::4 AAAA 2620:12a:8000::4
18	3.481594	2600:1f14:179:3f01:…	2001:250:3002:4240:…	TCP	86 443 → 10482 [ACK] Seq=1 Ack=2 Win=110 Len=0 SLE=1 SRE=2
19	3.491409	172.19.9.119	23.185.0.4	ICMP	74 Echo (ping) request id=0x0001, seq=21/5376, ttl=64 (reply in 20)
20	3.508847	23.185.0.4	172.19.9.119	ICMP	74 Echo (ping) reply id=0x0001, seq=21/5376, ttl=46 (request in 19)
21	4.504296	172.19.9.119	23.185.0.4	ICMP	74 Echo (ping) request id=0x0001, seq=22/5632, ttl=64 (reply in 22)
22	4.523321	23.185.0.4	172.19.9.119	ICMP	74 Echo (ping) reply id=0x0001, seq=22/5632, ttl=46 (request in 21)
23	5.513544	172.19.9.119	23.185.0.4	ICMP	74 Echo (ping) request id=0x0001, seq=23/5888, ttl=64 (reply in 24)
24	5.542926	23.185.0.4	172.19.9.119	ICMP	74 Echo (ping) reply id=0x0001, seq=23/5888, ttl=46 (request in 23)

主要截取的是DNS和ICMP部分

然后我们首先来看DNS的网络层部分

```
Internet Protocol Version 4, Src: 172.19.9.119, Dst: 10.8.4.4
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 61
    Identification: 0x776d (30573)
  > Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: UDP (17)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 172.19.9.119
    Destination Address: 10.8.4.4
User Datagram Protocol, Src Port: 64416, Dst Port: 53
    Source Port: 64416
    Destination Port: 53
    Length: 41
    Checksum: 0xc3d0 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 1]
  > [Timestamps]
    UDP payload (33 bytes)
```

字段	值	具体字段值
版本	ipv4	Internet Protocol Version 4
首部长度	20 bytes	Header Length
区分服务	0x00	Differentiated Services Field
总长度	61	total length
标识	0x776d	Identification
标志	0x00	Flags
片偏移	0	Fragment offset
生存时间	64	Time to Live
协议	UDP	Protocol
检验和	0x0000	Header Checksum

字段	值	具体字段值
原IP地址	172.19.9.119	Src
目的IP地址	10.8.4.4	Destination

可以看到ping命令一开始是通过DNS服务，在运输层中使用UDP协议来得到具体的数据以及目标域名的IP地址

接下来是ICMP的协议部分

```
Internet Protocol Version 4, Src: 172.19.9.119, Dst: 23.185.0.4
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 60
    Identification: 0xaff8 (45048)
  > Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: ICMP (1)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 172.19.9.119
    Destination Address: 23.185.0.4
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x4d46 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 21 (0x0015)
    Sequence Number (LE): 5376 (0x1500)
    [Response frame: 20]
  > Data (32 bytes)
```

字段	值	具体字段值
版本	ipv4	Internet Protocol Version 4
首部长度	20 bytes	Header Length
区分服务	0x00	Differentiated Services Field
总长度	60	total length
标识	0xaff8	Identification
标志	0x00	Flags
片偏移	0	Fragment offset
生存时间	64	Time to Live
协议	ICMP	Protocol

字段	值	具体字段值
检验和	0x4d46	Header Checksum
原IP地址	172.19.9.119	Src
目的IP地址	23.185.0.4	Destination

大体的字段与上面的解释相同，就是协议改成了ICMP协议，来传达ping命令的request和reply，然后这里因为涉及了具体的数据传输有了校验和的值以及标识字段

利用 wireshark 分析 ICMP/UDP 及相关 IP 服务

分析ICMP

首先分析ICMP的操作，在Windows条件下运行

```
Tracert www.ucdavis.edu
```

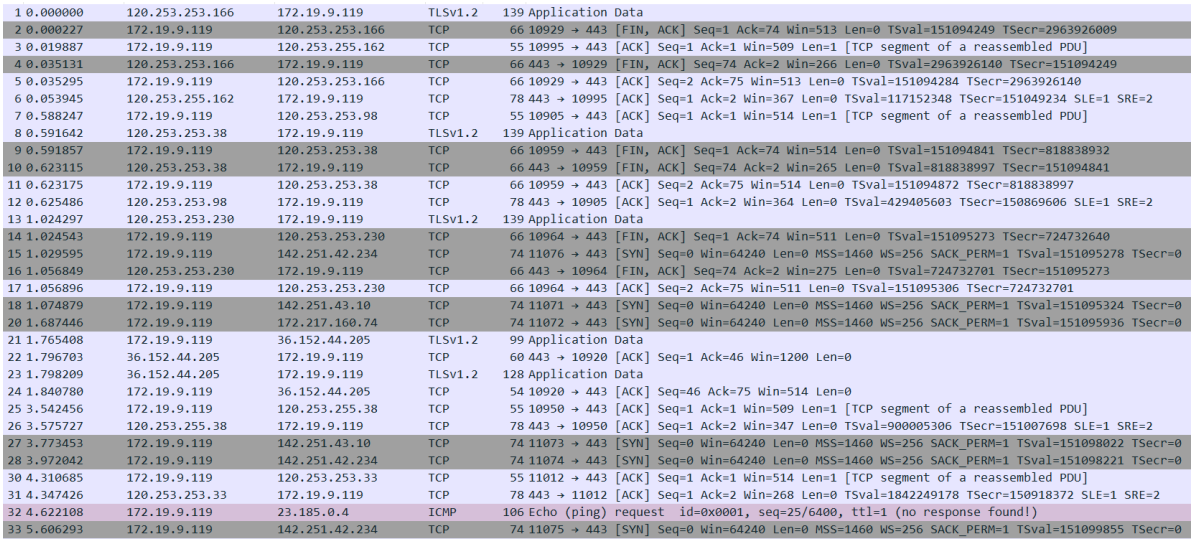
```
Tracing route to www.ucdavis.edu [23.185.0.4]
over a maximum of 30 hops:

  1     *        *        *     Request timed out.
  2     5 ms    39 ms    40 ms  10.44.36.201
  3     4 ms     4 ms     2 ms  10.44.16.201
  4     4 ms     1 ms     2 ms  10.10.1.42
  5     4 ms     2 ms     2 ms  120.236.174.129
  6     3 ms     5 ms     3 ms  120.197.11.5
  7     4 ms     3 ms     3 ms  183.233.109.81
  8     *        *        *     Request timed out.
  9    26 ms    26 ms    10 ms  111.24.5.21
 10    67 ms     7 ms     6 ms  111.24.4.246
 11     7 ms     5 ms     5 ms  221.183.68.137
 12    12 ms    13 ms    10 ms  221.183.52.86
 13    15 ms    11 ms    22 ms  221.183.55.81
 14     *        *       15 ms  223.120.2.81
 15     *        *        *     Request timed out.
 16    88 ms    85 ms    76 ms  63-217-16-189.static.pccwglobal.net [63.217.16.189]
 17    80 ms    83 ms    80 ms  BE46.clbr02.hkg12.pccwbtn.net [63.218.174.142]
 18     *       31 ms    28 ms  Fastly-Hu0-0-0-1-15.clbr02.hkg12.pccwbtn.net [63.217.237.106]
 19    16 ms    17 ms    17 ms  23.185.0.4
```

wireshark抓包如下

设置过滤器为ICMP时

32	4.622108	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=25/6400, ttl=1 (no response found!)
47	8.175016	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=26/6656, ttl=1 (no response found!)
74	12.181009	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=27/6912, ttl=1 (no response found!)
91	16.174795	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=28/7168, ttl=2 (no response found!)
92	16.180390	10.44.36.201	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
93	16.181201	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=29/7424, ttl=2 (no response found!)
94	16.220525	10.44.36.201	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
95	16.221951	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=30/7680, ttl=2 (no response found!)
96	16.262217	10.44.36.201	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
135	26.308438	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=31/7936, ttl=3 (no response found!)
136	26.312777	10.44.16.201	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
137	26.314094	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=32/8192, ttl=3 (no response found!)
138	26.318528	10.44.16.201	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
139	26.319072	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=33/8448, ttl=3 (no response found!)
140	26.321166	10.44.16.201	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
193	36.372652	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=34/8704, ttl=4 (no response found!)
194	36.376975	10.10.1.42	172.19.9.119	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
195	36.377910	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=35/8960, ttl=4 (no response found!)
196	36.379731	10.10.1.42	172.19.9.119	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
197	36.381105	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=36/9216, ttl=4 (no response found!)
198	36.383335	10.10.1.42	172.19.9.119	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
280	46.449061	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=37/9472, ttl=5 (no response found!)
281	46.453503	120.236.174.129	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
282	46.454319	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=38/9728, ttl=5 (no response found!)
283	46.456988	120.236.174.129	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
284	46.457999	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=39/9984, ttl=5 (no response found!)
285	46.460586	120.236.174.129	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
347	56.739075	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=40/10240, ttl=6 (no response found!)
348	56.742267	120.197.11.5	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
349	56.745141	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=41/10496, ttl=6 (no response found!)
350	56.750824	120.197.11.5	172.19.9.119	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
351	56.754588	172.19.9.119	23.185.0.4	ICMP	106	Echo (ping) request id=0x0001, seq=42/10752, ttl=6 (no response found!)

设置过滤器为ICMP||IP时



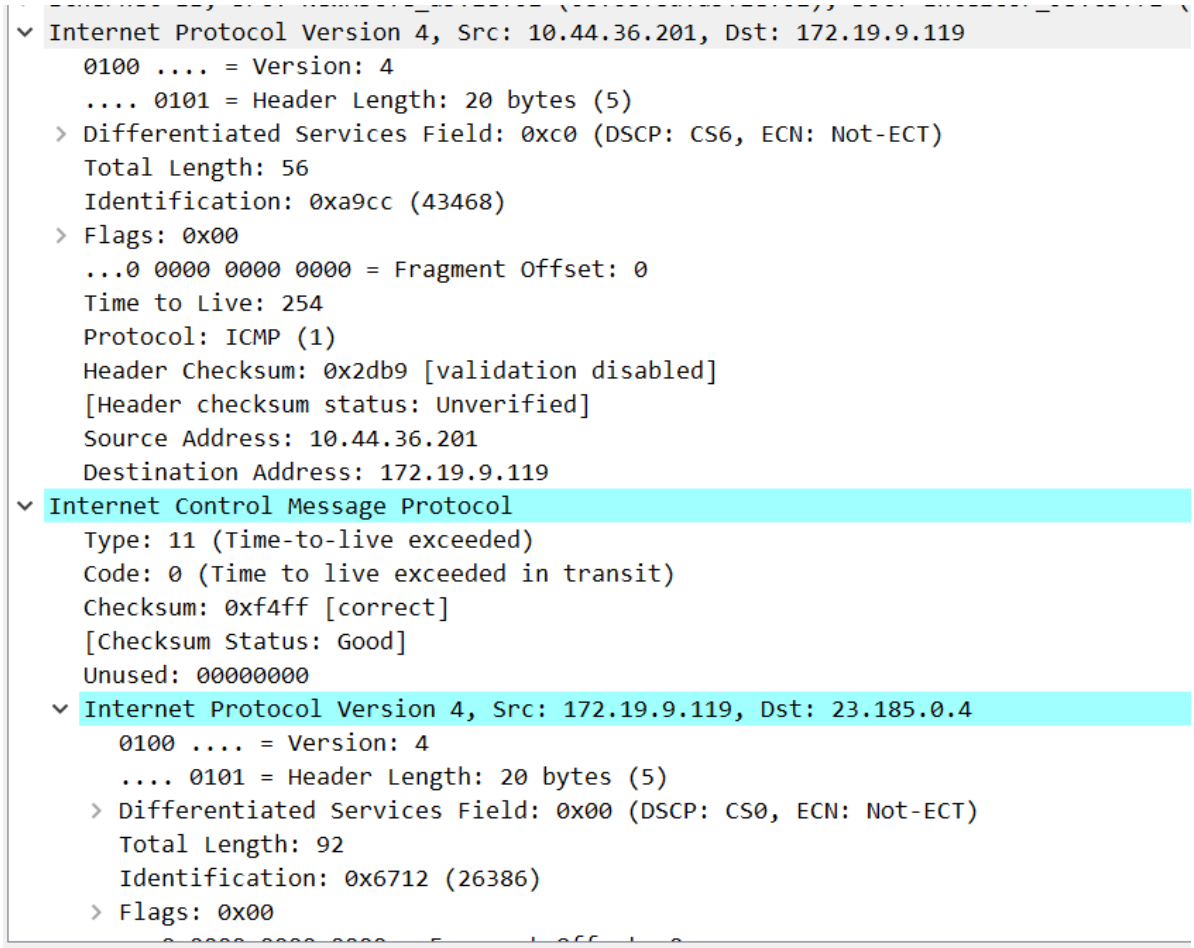
1	0.000000	120.253.253.166	172.19.9.119	TLSv1.2	139 Application Data
2	0.000227	172.19.9.119	120.253.253.166	TCP	66 10929 → 443 [FIN, ACK] Seq=1 Ack=74 Win=513 Len=0 TSval=151094249 TSecr=2963926009
3	0.019887	172.19.9.119	120.253.255.162	TCP	55 10995 → 443 [ACK] Seq=1 Ack=1 Win=509 Len=1 [TCP segment of a reassembled PDU]
4	0.035131	120.253.253.166	172.19.9.119	TCP	66 443 → 10929 [FIN, ACK] Seq=74 Ack=2 Win=266 Len=0 TSval=2963926140 TSecr=151094249
5	0.035295	172.19.9.119	120.253.253.166	TCP	66 10929 → 443 [ACK] Seq=2 Ack=75 Win=513 Len=0 TSval=151094284 TSecr=2963926140
6	0.053945	120.253.255.162	172.19.9.119	TCP	78 443 → 10995 [ACK] Seq=1 Ack=2 Win=367 Len=0 TSval=117152348 TSecr=151049234 SLE=1 SRE=2
7	0.588247	172.19.9.119	120.253.253.98	TCP	55 10905 → 443 [ACK] Seq=1 Ack=1 Win=514 Len=1 [TCP segment of a reassembled PDU]
8	0.591642	120.253.253.38	172.19.9.119	TLSv1.2	139 Application Data
9	0.591857	172.19.9.119	120.253.253.38	TCP	66 10959 → 443 [FIN, ACK] Seq=1 Ack=74 Win=514 Len=0 TSval=151094841 TSecr=818838932
10	0.623115	120.253.253.38	172.19.9.119	TCP	66 443 → 10959 [FIN, ACK] Seq=74 Ack=2 Win=265 Len=0 TSval=818838997 TSecr=151094841
11	0.623175	172.19.9.119	120.253.253.38	TCP	66 10959 → 443 [ACK] Seq=2 Ack=75 Win=514 Len=0 TSval=151094872 TSecr=818838997
12	0.625486	120.253.253.98	172.19.9.119	TCP	78 443 → 10905 [ACK] Seq=1 Ack=2 Win=364 Len=0 TSval=429405603 TSecr=150869606 SLE=1 SRE=2
13	1.024297	120.253.253.230	172.19.9.119	TLSv1.2	139 Application Data
14	1.024543	172.19.9.119	120.253.253.230	TCP	66 10964 → 443 [FIN, ACK] Seq=1 Ack=74 Win=511 Len=0 TSval=151095273 TSecr=724732640
15	1.029595	172.19.9.119	142.251.42.234	TCP	74 11076 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=151095278 TSecr=0
16	1.056849	120.253.253.230	172.19.9.119	TCP	66 443 → 10964 [FIN, ACK] Seq=74 Ack=2 Win=275 Len=0 TSval=724732701 TSecr=151095273
17	1.056896	172.19.9.119	120.253.253.230	TCP	66 10964 → 443 [ACK] Seq=2 Ack=75 Win=511 Len=0 TSval=151095306 TSecr=724732701
18	1.074879	172.19.9.119	142.251.43.10	TCP	74 11071 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=151095324 TSecr=0
20	1.687446	172.19.9.119	172.217.160.74	TCP	74 11072 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=151095936 TSecr=0
21	1.765408	172.19.9.119	36.152.44.205	TLSv1.2	99 Application Data
22	1.796703	36.152.44.205	172.19.9.119	TCP	60 443 → 10920 [ACK] Seq=1 Ack=46 Win=1200 Len=0
23	1.798209	36.152.44.205	172.19.9.119	TLSv1.2	128 Application Data
24	1.840780	172.19.9.119	36.152.44.205	TCP	54 10920 → 443 [ACK] Seq=46 Ack=75 Win=514 Len=0
25	3.542456	172.19.9.119	120.253.255.38	TCP	55 10950 → 443 [ACK] Seq=1 Ack=1 Win=509 Len=1 [TCP segment of a reassembled PDU]
26	3.575727	120.253.255.38	172.19.9.119	TCP	78 443 → 10950 [ACK] Seq=1 Ack=2 Win=347 Len=0 TSval=900005306 TSecr=151007698 SLE=1 SRE=2
27	3.773453	172.19.9.119	142.251.43.10	TCP	74 11073 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=151098022 TSecr=0
28	3.972042	172.19.9.119	142.251.42.234	TCP	74 11074 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=151098221 TSecr=0
30	4.310685	172.19.9.119	120.253.253.33	TCP	55 11012 → 443 [ACK] Seq=1 Ack=1 Win=514 Len=1 [TCP segment of a reassembled PDU]
31	4.347426	120.253.253.33	172.19.9.119	TCP	78 443 → 11012 [ACK] Seq=1 Ack=2 Win=268 Len=0 TSval=1842249178 TSecr=150918372 SLE=1 SRE=2
32	4.622108	172.19.9.119	23.185.0.4	ICMP	106 Echo (ping) request id=0x0001, seq=25/6400, ttl=1 (no response found!)
33	5.606293	172.19.9.119	142.251.42.234	TCP	74 11075 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=151099855 TSecr=0

此时可以看到本地主机 172.19.9.119向其他地址发送请求数据包，然后收到路径上的部分数据包，从而在命令行中得到

主机172.26.5.18开始想域名对应的ip地址发一个TTL=1的UDP数据包，而经过的第一个路由器收到这个数据包以后，就自动把TTL减1，而TTL变为0以后，路由器就把这个包给抛弃了，并同时产生 一个主机不可达的ICMP数据报给主机。

主机收到这个数据报以后再发一个TTL=2的UDP数据报给目的主机，然后刺激第二个路由器给主机发ICMP数据报。如此往复直到到达目的主机。

接下来具体看一个icmp 的数据包



```
Internet Protocol Version 4, Src: 10.44.36.201, Dst: 172.19.9.119
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
    Total Length: 56
    Identification: 0xa9cc (43468)
  > Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 254
    Protocol: ICMP (1)
    Header Checksum: 0x2db9 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 10.44.36.201
    Destination Address: 172.19.9.119
Internet Control Message Protocol
    Type: 11 (Time-to-live exceeded)
    Code: 0 (Time to live exceeded in transit)
    Checksum: 0xf4ff [correct]
    [Checksum Status: Good]
    Unused: 00000000
  Internet Protocol Version 4, Src: 172.19.9.119, Dst: 23.185.0.4
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 92
      Identification: 0x6712 (26386)
    > Flags: 0x00
```

这是从目标地址向本地地址发送的错误信息，因为TTL结束了，说明所定义的TTL无法到达具体的目标地址，需要再加一个TTL来重新进行

type=11， code =0，说明TTL expired ，这就是目标发来的报错信息

分析UDP协议

（1）因为windows下的tracert命令发送的是ICMP的包，所以这次实验采用在unbuntu虚拟机的环境下进行，

输入命令

```
traceroute ucdavis.edu
```

控制台就结果如图

```
yzb@yzb-virtual-machine:~$ traceroute ucdavis.edu
traceroute to ucdavis.edu (23.185.0.4), 30 hops max, 60 byte packets
 1  _gateway (172.26.127.254)  3.395 ms  3.303 ms  3.257 ms
 2  10.44.37.201 (10.44.37.201)  9.335 ms  9.280 ms  9.230 ms
 3  10.44.16.201 (10.44.16.201)  4.574 ms  4.526 ms  4.477 ms
 4  10.10.1.42 (10.10.1.42)  4.432 ms  4.383 ms  4.333 ms
 5  120.236.174.129 (120.236.174.129)  5.286 ms  5.227 ms  5.179 ms
 6  120.197.11.5 (120.197.11.5)  5.136 ms  8.019 ms  6.864 ms
 7  183.233.109.85 (183.233.109.85)  9.364 ms 183.233.109.81 (183.233.109.81)  4.154 ms  6.365 ms
 8  * * *
 9  111.24.5.209 (111.24.5.209)  10.038 ms  8.858 ms 111.24.5.21 (111.24.5.21)  11.297 ms
10  111.24.5.174 (111.24.5.174)  8.983 ms 111.24.5.170 (111.24.5.170)  18.097 ms 111.24.4.250 (11
1.24.4.250)  13.962 ms
11  221.183.68.145 (221.183.68.145)  9.732 ms 221.183.68.141 (221.183.68.141)  9.681 ms 221.183.6
8.145 (221.183.68.145)  7.448 ms
12  221.183.25.117 (221.183.25.117)  24.333 ms  24.285 ms 221.183.25.121 (221.183.25.121)  9.972
ms
13  221.183.55.81 (221.183.55.81)  9.699 ms 221.183.68.130 (221.183.68.130)  15.259 ms 221.183.55
.81 (221.183.55.81)  10.909 ms
14  223.120.2.81 (223.120.2.81)  17.061 ms 223.120.2.101 (223.120.2.101)  17.248 ms 223.120.2.85
(223.120.2.85)  18.339 ms
15  223.120.2.118 (223.120.2.118)  16.880 ms  16.776 ms  16.729 ms
16  63-217-16-189.static.pccwglobal.net (63.217.16.189)  41.102 ms  31.517 ms 63-217-103-25.stati
c.pccwglobal.net (63.217.103.25)  229.709 ms
17  BE46.clbr02.hkg12.pccwbtn.net (63.218.174.142)  95.507 ms BE45.clbr02.hkg12.pccwbtn.net (63.2
18.174.130)  94.329 ms  69.789 ms
18  * * *
```

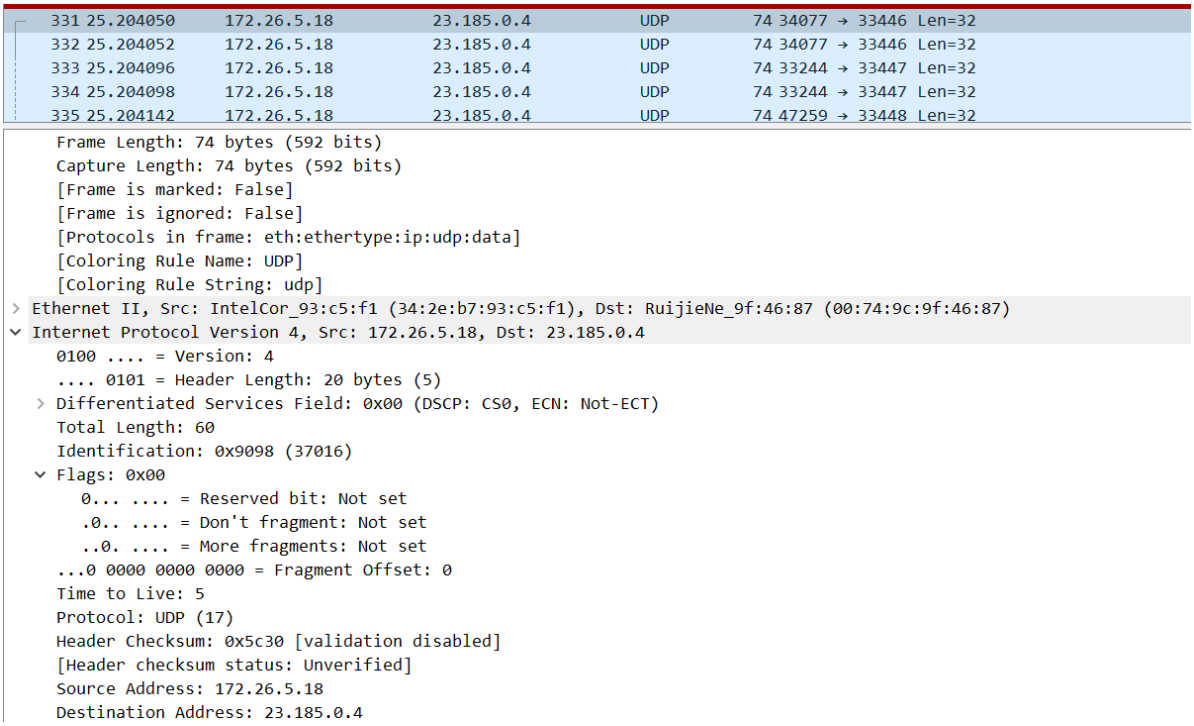
wireshark抓包得到结果如下

155	10.386463	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0xdcd5e088
235	22.228363	172.26.91.215	10.8.8.8	DNS	89	Standard query 0xf9a4 A v10.events.data.microsoft.com
236	22.228368	172.26.91.215	10.8.8.8	DNS	89	Standard query 0xf9a4 A v10.events.data.microsoft.com
237	22.231387	10.8.8.8	172.26.91.215	DNS	216	Standard query response 0xf9a4 A v10.events.data.microsoft.com CN
275	23.344061	172.26.50.14	224.0.0.251	MDNS	571	Standard query response 0x0000 TXT, cache flush PTR _mi-connect._
276	23.344749	fe80::ec84:efb9:da2…	ff02::fb	MDNS	591	Standard query response 0x0000 TXT, cache flush PTR _mi-connect._
301	25.154165	172.26.5.18	10.8.8.8	DNS	82	Standard query 0x45a1 A ucdavis.edu OPT
302	25.154169	172.26.5.18	10.8.8.8	DNS	82	Standard query 0x45a1 A ucdavis.edu OPT
303	25.154278	172.26.5.18	10.8.8.8	DNS	82	Standard query 0xf668 AAAA ucdavis.edu OPT
304	25.154279	172.26.5.18	10.8.8.8	DNS	82	Standard query 0xf668 AAAA ucdavis.edu OPT
305	25.184546	10.8.8.8	172.26.5.18	DNS	138	Standard query response 0xf668 AAAA ucdavis.edu AAAA 2620:12a:800
306	25.202702	10.8.8.8	172.26.5.18	DNS	98	Standard query response 0x45a1 A ucdavis.edu A 23.185.0.4 OPT
307	25.203342	172.26.5.18	23.185.0.4	UDP	74	58266 → 33434 Len=32
308	25.203345	172.26.5.18	23.185.0.4	UDP	74	58266 → 33434 Len=32
309	25.203410	172.26.5.18	23.185.0.4	UDP	74	41176 → 33435 Len=32
310	25.203412	172.26.5.18	23.185.0.4	UDP	74	41176 → 33435 Len=32
311	25.203459	172.26.5.18	23.185.0.4	UDP	74	54045 → 33436 Len=32
312	25.203461	172.26.5.18	23.185.0.4	UDP	74	54045 → 33436 Len=32
313	25.203554	172.26.5.18	23.185.0.4	UDP	74	55719 → 33437 Len=32
314	25.203557	172.26.5.18	23.185.0.4	UDP	74	55719 → 33437 Len=32
315	25.203607	172.26.5.18	23.185.0.4	UDP	74	50995 → 33438 Len=32
316	25.203609	172.26.5.18	23.185.0.4	UDP	74	50995 → 33438 Len=32
317	25.203656	172.26.5.18	23.185.0.4	UDP	74	46201 → 33439 Len=32
318	25.203658	172.26.5.18	23.185.0.4	UDP	74	46201 → 33439 Len=32
319	25.203705	172.26.5.18	23.185.0.4	UDP	74	54428 → 33440 Len=32
320	25.203707	172.26.5.18	23.185.0.4	UDP	74	54428 → 33440 Len=32
321	25.203754	172.26.5.18	23.185.0.4	UDP	74	58159 → 33441 Len=32

下面来对信息具体分析

235	22.228363	172.26.91.215	10.8.8.8	DNS	89	Standard query 0xf9a4 A v10.events.data.microsoft.com
236	22.228368	172.26.91.215	10.8.8.8	DNS	89	Standard query 0xf9a4 A v10.events.data.microsoft.com
237	22.231387	10.8.8.8	172.26.91.215	DNS	216	Standard query response 0xf9a4 A v10.events.data.microsoft.com CNAME global.asimov.events.data.trafficmanager…
275	23.344061	172.26.50.14	224.0.0.251	MDNS	571	Standard query response 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"zhonghc","as":"[8193, 81…
276	23.344749	fe80::ec84:efb9:da2…	ff02::fb	MDNS	591	Standard query response 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"zhonghc","as":"[8193, 81…
301	25.154165	172.26.5.18	10.8.8.8	DNS	82	Standard query 0x45a1 A ucdavis.edu OPT
302	25.154169	172.26.5.18	10.8.8.8	DNS	82	Standard query 0x45a1 A ucdavis.edu OPT
303	25.154278	172.26.5.18	10.8.8.8	DNS	82	Standard query 0xf668 AAAA ucdavis.edu OPT
304	25.154279	172.26.5.18	10.8.8.8	DNS	82	Standard query 0xf668 AAAA ucdavis.edu OPT
305	25.184546	10.8.8.8	172.26.5.18	DNS	138	Standard query response 0xf668 AAAA ucdavis.edu AAAA 2620:12a:8000::4 AAAA 2620:12a:8001::4 OPT
306	25.202702	10.8.8.8	172.26.5.18	DNS	98	Standard query response 0x45a1 A ucdavis.edu A 23.185.0.4 OPT

开始是对ucdavis.edu的域名解析，是用了DNS的协议来进行，具体的解析过程前面几个实验中也涉及到了，就是通过一层一层的DNS服务器来向本地返回所需要访问域名的ip地址



具体分析其中一条udp协议请求，先截取了网络层的报文内容

字段	值	意义
Version	4	代码ipv4协议
Differentiated Services Field	0x00	区分服务
header length	20 bytes	头部字段长度
total length	60	总长度
Identification	0x9098	标识符
Time to live	5	生存时间
Protocol	UDP	UDP协议
Source Address	172.26.5.18	主机ip地址
Destination Address	23.185.0.4	目标ip地址

传输层/UDP层

```
User Datagram Protocol, Src Port: 34077, Dst Port: 33446
    Source Port: 34077
    Destination Port: 33446
    [Expert Info (Chat/Sequence): Possible traceroute: hop #4, attempt #3]
        [Possible traceroute: hop #4, attempt #3]
        [Severity level: Chat]
        [Group: Sequence]
    Length: 40
    Checksum: 0x39ec [unverified]
    [Checksum Status: Unverified]
    [Stream index: 22]
    [Timestamps]
        [Time since first frame: 0.000000000 seconds]
        [Time since previous frame: 0.000000000 seconds]
    UDP payload (32 bytes)
Data (32 bytes)
    Data: 404142434445464748494a4b4c4d4e4f505152535455565758595a5b5c5d5e5f
    [Length: 32]
```

字段	值	意义
Src Port	34077	主机端口
Dst Port	33446	目标端口
checksum	0x39ec	数据校验和

所以traceroute ucdavis.edu的流程主要有以下几个方面

首先，本地开始解析ucdavis.edu这个域名的ip地址，向上一级DNS服务器进行请求，逐级请求下得到ucdavis.edu的ip地址为23.185.0.4

然后主机172.26.5.18开始想域名对应的ip地址发一个TTL=1的UDP数据包，而经过的第一个路由器收到这个数据包以后，就自动把TTL减1，而TTL变为0以后，路由器就把这个包给抛弃了，并同时产生 一个主机不可达的ICMP数据报给主机。

主机收到这个数据报以后再发一个TTL=2的UDP数据报给目的主机，然后刺激第二个路由器给主机发ICMP数据报。如此往复直到到达目的主机。

```
339 25.206553   172.26.127.254  172.26.5.18     ICMP    102 Time-to-live exceeded (Time to live exceeded in transit)
340 25.206553   172.26.127.254  172.26.5.18     ICMP    102 Time-to-live exceeded (Time to live exceeded in transit)
341 25.206553   172.26.127.254  172.26.5.18     ICMP    102 Time-to-live exceeded (Time to live exceeded in transit)
342 25.206870   172.26.5.18     10.8.8.8        DNS      98 Standard query 0x5681 PTR 254.127.26.172.in-addr.arpa OPT
343 25.206873   172.26.5.18     10.8.8.8        DNS      98 Standard query 0x5681 PTR 254.127.26.172.in-addr.arpa OPT
344 25.208025   10.44.16.201    172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
345 25.208025   10.44.16.201    172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
346 25.208025   10.44.16.201    172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
347 25.208258   10.10.1.42      172.26.5.18     ICMP    102 Time-to-live exceeded (Time to live exceeded in transit)
348 25.208258   10.10.1.42      172.26.5.18     ICMP    102 Time-to-live exceeded (Time to live exceeded in transit)
349 25.208258   10.10.1.42      172.26.5.18     ICMP    102 Time-to-live exceeded (Time to live exceeded in transit)
350 25.209225   120.236.174.129 172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
351 25.209225   120.236.174.129 172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
352 25.209225   120.236.174.129 172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
353 25.209225   120.197.11.5    172.26.5.18     ICMP     70 Time-to-live exceeded (Time to live exceeded in transit)
```

ip数据包的内容为udp传输层下的data数据，如下图

```
Data (32 bytes)
    Data: 404142434445464748494a4b4c4d4e4f505152535455565758595a5b5c5d5e5f
    [Length: 32]
```