北 京 邮 电 大 学

计 算 机 科 学 与 技 术 学 院

《下一代Internet技术与协议》

实验报告

姓名：\_\_\_于海鑫\_\_\_\_

学号：\_\_2017211240\_

班级：\_\_2017211305\_

2020年04月

**实验报告一**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验名称 | | ICMPv6抓包分析 | | |
| 实验目的 | | 了解 ICMPv6 协议的报文 | | |
| 实验完成人 | | 于海鑫 | 完成时间 | 2020/04/30 |
| 实验环境 |  | | | |
| 实验步骤与结果分析 | | | | |
| 1.关闭IPv4协议 因为运营商的问题，直接使用宽带网络无法连接倒 IPv6，我们改用手机热点做实验。  在连接好手机提供的热点后，我们打开控制面板，暂时关闭到 Wi-Fi 连接的 IPv4 功能，只保留 IPv6。 2. 分析本机IPv6相关地址信息 在终端输入 `ifconfig` 之后，我们在输出的最后找到了如下几行信息：  无线局域网适配器 WLAN:  连接特定的 DNS 后缀 . . . . . . . :  IPv6 地址 . . . . . . . . . . . . : 2409:8918:3650:1c02:d9d5:92bb:cee9:85dc  临时 IPv6 地址. . . . . . . . . . : 2409:8918:3650:1c02:a07d:4eb7:5bec:4f3e  本地链接 IPv6 地址. . . . . . . . : fe80::d9d5:92bb:cee9:85dc%8  默认网关. . . . . . . . . . . . . : fe80::1075:d751:7a60:3ff0%8  我们来大致分析一下这几个地址：  2409:8918:3650:1c02:d9d5:92bb:cee9:85dc 根据该地址首部的 2409 可以确定，该地址的类型为“**可汇聚全球单播地址**”，用于唯一标识一个接口，和 ipv4 的 公网 IP 类似。该地址的前缀为 64 位，也就是“2409:8918:3650:1c02”。使用归属地查询可以发现该前缀来自吉林省移动，符合个人的位置以及移动网论运营商。  2409:8918:3650:1c02:a07d:4eb7:5bec:4f3e 与上一个类似，也是 “**可汇聚全球单播地址**”，该地址被标记为临时地址，表示其一般会在一定时间内进行更换，以此保护用户隐私，实际上网时候使用的 IP 地址为该地址。  fe80::d9d5:92bb:cee9:85dc%8 该地址是“**链路本地单播地址**”，主要用于与路由器（手机）或者同网络内的其余用户通信。  fe80::1075:d751:7a60:3ff0%8 该地址是“**链路本地单播地址**”，为路由器的地址。 3. 使用wireshark软件进行抓包 在这里，我们选定的网站为人民网([www.people.com.cn](http://www.people.com.cn))。  使用 `nslookup` 进行ip 查找的回应如下：  ```powershell  PS C:\Users\name1> nslookup www.people.com.cn  DNS request timed out.  timeout was 2 seconds.  服务器: UnKnown  Address: fe80::1075:d751:7a60:3ff0  非权威应答:  名称: hpcc-page-ipv6.cncssr.chinacache.net  Addresses: 2409:8c14:f1a:8a02::1:51  2409:8c14:f1a:8a02::1:56  120.201.106.51  120.201.106.56  Aliases: www.people.com.cn  [www.people.chinacache.net](http://www.people.chinacache.net)  ```  可以看到，实际返回的地址为一个 CDN 的 IP 地址，符合人民网的特点。  使用 `ping` 测试网站的应答如下：  ```powershell  PS C:\Users\name1> ping 2409:8c14:f1a:8a02::1:51  正在 Ping 2409:8c14:f1a:8a02::1:51 具有 32 字节的数据:  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=67ms  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=89ms  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=71ms  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=80ms  2409:8c14:f1a:8a02::1:51 的 Ping 统计信息:  数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，  往返行程的估计时间(以毫秒为单位):  最短 = 67ms，最长 = 89ms，平均 = 76ms  ```  使用 ` tracert` 溯源的结果如下：  ```powershell  PS C:\Users\name1> ping 2409:8c14:f1a:8a02::1:51  正在 Ping 2409:8c14:f1a:8a02::1:51 具有 32 字节的数据:  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=67ms  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=89ms  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=71ms  来自 2409:8c14:f1a:8a02::1:51 的回复: 时间=80ms  2409:8c14:f1a:8a02::1:51 的 Ping 统计信息:  数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，  往返行程的估计时间(以毫秒为单位):  最短 = 67ms，最长 = 89ms，平均 = 76ms  PS C:\Users\name1> tracert -d 2409:8c14:f1a:8a02::1:51  通过最多 30 个跃点跟踪到 2409:8c14:f1a:8a02::1:51 的路由  1 5 ms 8 ms 4 ms 2409:8918:3650:1c02:2c5a:3eb6:6db9:3d80  2 \* \* \* 请求超时。  3 61 ms 60 ms 47 ms 2409:8018:2800::54  4 50 ms 53 ms 39 ms 2409:8018:2800::45  5 \* \* \* 请求超时。  6 \* \* \* 请求超时。  7 \* \* \* 请求超时。  8 \* \* \* 请求超时。  9 \* \* \* 请求超时。  10 \* \* \* 请求超时。  11 \* \* \* 请求超时。  12 99 ms 78 ms 75 ms 2409:8c14:f1a:8a02::1:51  跟踪完成。  ```  使用 ` ping 2409:8c14:f1a:8a02::1:51 -l 3000` 发长包的回应如下：  ```powershell  PS C:\Users\name1> ping 2409:8c14:f1a:8a02::1:51 -l 3000  正在 Ping 2409:8c14:f1a:8a02::1:51 具有 3000 字节的数据:  请求超时。  请求超时。  请求超时。  请求超时。  2409:8c14:f1a:8a02::1:51 的 Ping 统计信息:  数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，  ```  可见该网站十分保守，注重安全性。对于长的 ping 包从未回应。 4. 对抓包内容进行分析4.1 nslookup   这个是 `nslookup` 产生的查找人民网的 IPv6 地址记录的查询以及结果。下面，我们分析以下其具体构造：    可以发现，该包为基于 IPv6 协议的 UDP 包：   1. 其 IP 头为 `0b0110`，指明其协议为 IPv6。 2. 之后是 Traffic Class，共有 16 个级别，我们的包为最低级别，表示当网络拥塞时可以将包丢弃掉。 3. 之后是流标签字段，该字段是 IPv6 数据报中新增的一个字段，共 20 位，可用来标记报文的数据流类型，以便在网络层区分不同的报文。流标签字段有源节点分配，通过流标签、源地址、目的地址三元组方式就可以唯一标识一条通信流。本数据报的流标签就为 0xae5b8。 4. 之后是有效载荷长度，指明该数据报除基本首部以外的字节数。 5. 之后是下一个首部，相当于 IPv4 的协议字段或可选字段。下一个首部字段的作用和 IPv4 的协议字段一样，它的值指出了基本首部后 面的数据应交付给 IP 上面的哪一个高层协议(如 6 表示 TCP，17 表示 UDP)。当出现扩展首部时，下一个首部字段的值就标识后面第一个扩展首部的类型 6. 在之后是跳数限制，用来防止数据报在网络中无限期地存在。 7. 最后是源地址跟目的地址，不再叙述。   之后的 UDP 协议头与 DNS 包体与 IPv4 时代差距不大， 唯一的不同就是 DNS 查询的是 AAAA 记录，而非 IPv4 时代的 A 记录。  打开其回复包，可以看到结果已正确返回：   4.2 ping 再上一节我们已经看到了 IP 包的构造，这里我们直接查看 ICMPv6 协议的部分：    可以发现当前的 ICMP 协议依然保持着简洁的特性。下面我们大致介绍下各个字段：   1. Type：类型字段，128 表示该 ICMP 报文类型为回显请求 2. Code：代码字段，如果 Type 字段为 1-4 的话，代码字段才会配合类型字段 进行更详细的数据报类型说明，但因为本数据报的类型字段为 128，所以 Code 字段并没有实际的意义，整个 ICMP 数据报的类型就是 Type 表示的回显请求 3. Checksum：校验和字段，该字段用于保证数据接收的完整性 4. Identifier：标识，占两字节，用于标识本 ICMP 进程，但仅适用于回显请求和应答 ICMP 报文 5. Sequence：序列号，用于表示该数据报的发送序号，发送的数据报和对方回 复的数据报二者的序列号是一致的。 6. Data: 请求数据区   下面是该包的回复：    大致同请求包。不过值得一提的是 ICMP 的包的 IP 的优先级高了一点，是 0x4 而非 nslookup 的 0。 4.3 tracert Tracert 依赖于 ICMP 协议，其发出、收到的包的截图如下（部分）：    大致的协议我们在上一节已经说过了，在此不再赘述。我们只看一下 ICMP 中的出错的包的效果：    可以发现，其使用 Type 标识包的作用，之后通过 Code 表达了错误的具体信息。 4.4 ping -l 使用 ping -l 时候，会出现在 IPv4 时代也常见的分段：    图中可见为我们发送出去的三个分段。这部分大家都十分熟悉，不再展开描述。 | | | | |
| 分析与思考 | | | | |
| 本次实验其实还是比较顺利的，不过由于运营商（中国联通）的问题，我们无法直接通过宽带做实验，仍然需要使用手机热点，希望几年后能有所改善。  实验中遇到的问题为人民网的 IP 不接受长度为 3000 的 ICMP 包，推测是安全设置，或者是 MTU 的不同导致了这一问题。 | | | | |