

## Wireshark 实验指导书

### 一、实验目的

通过对 Wireshark 抓包实例进行分析，加强对 SMTP 协议的理解。

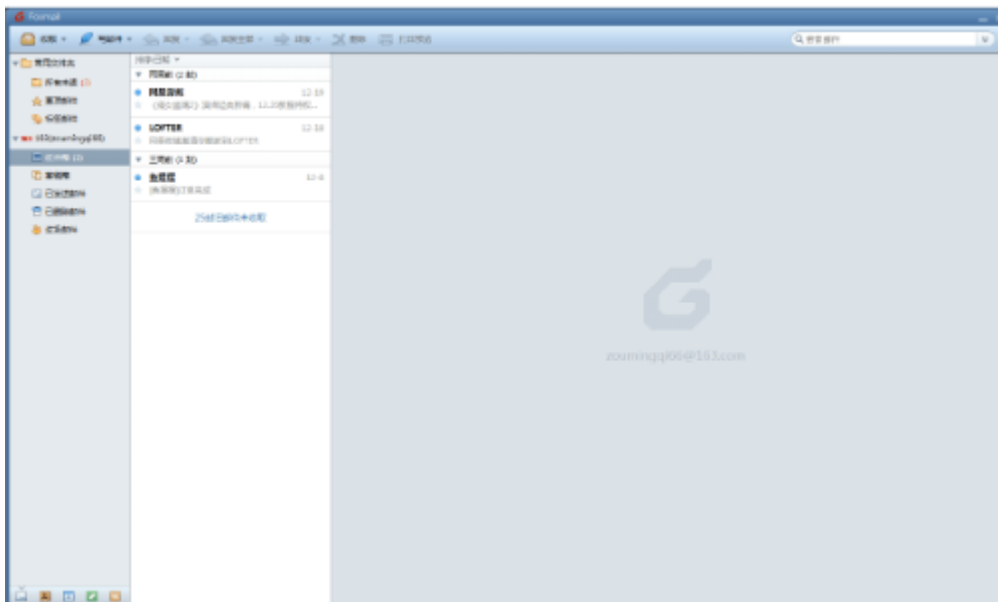
### 二、实验任务

#### 1. 邮箱登陆及接收过程（POP3 协议）

利用 Wireshark 软件抓包，得到邮箱登录的信息和发送邮件的信息，并根据所抓包对 SMTP 协议进行分析。打开 wireshark 开始抓包。然后打开 Foxmail 邮箱，输入用户名，密码，POP 服务器，SMTP 服务器，如下图：



点击创建，登录成功后点击收取，结果如下图：



打开 wireshark，停止捕获并对结果进行分析：

因为 POP3 协议默认的传输协议是 TCP 协议，因此连接服务器要先进行三次握手。请提供该过程的 Wireshark 屏幕截图并解释三次握手的过程。

## 2. 邮寄发送过程（SMTP 协议）

- 打开 foxmail 客户端，填写所发送邮件的相关信息。
- 打开 Wireshark 软件，选择正在上网的网卡，开始抓包。
- 在 Foxmail 客户端中对已编写好的邮件，点击左上角的发送按钮，邮件开始发送，发送成功以后，关闭界面。

你应该得到类似下图所示的捕获结果：

No.	Time	Source	Destination	Protocol	Length	Info
814	8.474711	220.181.12.13	10.24.2.190	SMTP	119	S: 220 163.com Anti-spam GT for Coremail System (163com[20141201])
815	8.475469	10.24.2.190	220.181.12.13	SMTP	76	C: EHLO WIN-24H0LB5CM6D
817	8.503247	220.181.12.13	10.24.2.190	SMTP	239	S: 250-mail   PIPELINING   AUTH LOGIN PLAIN   AUTH=LOGIN PLAIN   coremail 1Uxr2xKj7kG0xkL...
818	8.503456	10.24.2.190	220.181.12.13	SMTP	66	C: AUTH LOGIN
820	8.530785	220.181.12.13	10.24.2.190	SMTP	72	S: 334 dXN1cmShbWU6
821	8.531006	10.24.2.190	220.181.12.13	SMTP	84	C: User: MTUxMTYzNzE0NzhAMTYzLmNvbQ==
823	8.558225	220.181.12.13	10.24.2.190	SMTP	72	S: 334 UGFzc3dvcmQ6
824	8.558390	10.24.2.190	220.181.12.13	SMTP	76	C: Pass: ZGVuZ3NpamlhQDEyMTA=
833	8.756208	220.181.12.13	10.24.2.190	SMTP	85	S: 235 Authentication successful
834	8.761285	10.24.2.190	220.181.12.13	SMTP	88	C: MAIL FROM: <15116371478@163.com>
838	8.794659	220.181.12.13	10.24.2.190	SMTP	67	S: 250 Mail OK
839	8.794884	10.24.2.190	220.181.12.13	SMTP	94	C: RCPT TO: <51194501112@stu.ecnu.edu.cn>
841	8.823524	220.181.12.13	10.24.2.190	SMTP	67	S: 250 Mail OK
842	8.823748	10.24.2.190	220.181.12.13	SMTP	94	C: RCPT TO: <51194501112@stu.ecnu.edu.cn>
846	8.851810	220.181.12.13	10.24.2.190	SMTP	67	S: 250 Mail OK
847	8.852086	10.24.2.190	220.181.12.13	SMTP	60	C: DATA
852	8.880073	220.181.12.13	10.24.2.190	SMTP	91	S: 354 End data with <CR><LF>.<CR><LF>
853	8.880226	10.24.2.190	220.181.12.13	SMTP	520	C: DATA fragment, 466 bytes
857	8.947201	10.24.2.190	220.181.12.13	SMTP/L...	1003	from: "15116371478@163.com" <15116371478@163.com>, subject: test, (text/plain) (text/htm...
860	8.984725	220.181.12.13	10.24.2.190	SMTP	127	S: 250 Mail OK queued as smtp9,DcCOWAB3vnxBoxejA3jAg--.6381652 1587323613
861	8.985612	10.24.2.190	220.181.12.13	SMTP	60	C: QUIT
862	9.013656	220.181.12.13	10.24.2.190	SMTP	63	S: 221 Bye

> Frame 838: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface \Device\NPF\_{E05D9531-825A-4F8D-B349-1160782002E0}, id 0  
 > Ethernet II, Src: fa:ff:ff:ff:ff:ff (fa:ff:ff:ff:ff:ff), Dst: RealtekU\_a0:04:7e (52:54:00:a0:04:7e)  
 > Internet Protocol Version 4, Src: 220.181.12.13, Dst: 10.24.2.190

请从上面的屏幕截图中分析下面问题。

1. 在实验中，第多少帧可以看到 Foxmail 向服务器发送 EHLO 指令，表明身份？我们可以看到 Foxmail 客户端的主机名是什么？
2. 我们可以看到发送邮件的 User 和 Pass 吗？是以什么形式？（经过 base64 加密的，因为 SMTP 不接收明文）
3. 我们可以看到发送邮件的发送者和接受者吗？是以什么形式？（这个是明文的）
4. Foxmail 客户端发送的数据大小是多少？
5. 客户端向服务器发送什么命令表示释放服务器连接？服务器返回什么状态码表示通信过程结束？（“QUIT”）（服务器返回“221”表示同意双方释放 TCP 连接，通信过程结束）
6. 提供屏幕截图。

## 三、实验要求

以实验报告的形式把过程截图与答案依次陈列出来，要求独立完成。

## Wireshark 实验指导书

### 一、实验目的

域名系统(DNS)将主机名转换为 IP 地址，在互联网基础架构中发挥关键作用。在本实验中，我们将仔细查看 DNS 在客户端的细节。回想一下，客户端在 DNS 中的角色相对简单——客户端向其本地 DNS 服务器发送请求，并接收一个响应。如书中的图 2.21 和 2.22 所示，由于 DNS 分层服务器之间相互通信，可以递归地或迭代地解析客户端的 DNS 查询请求，而大多数操作是不可见的。然而，从 DNS 客户端的角度来看，协议非常简单——将查询指向为本地 DNS 服务器，并从该服务器接收到响应。

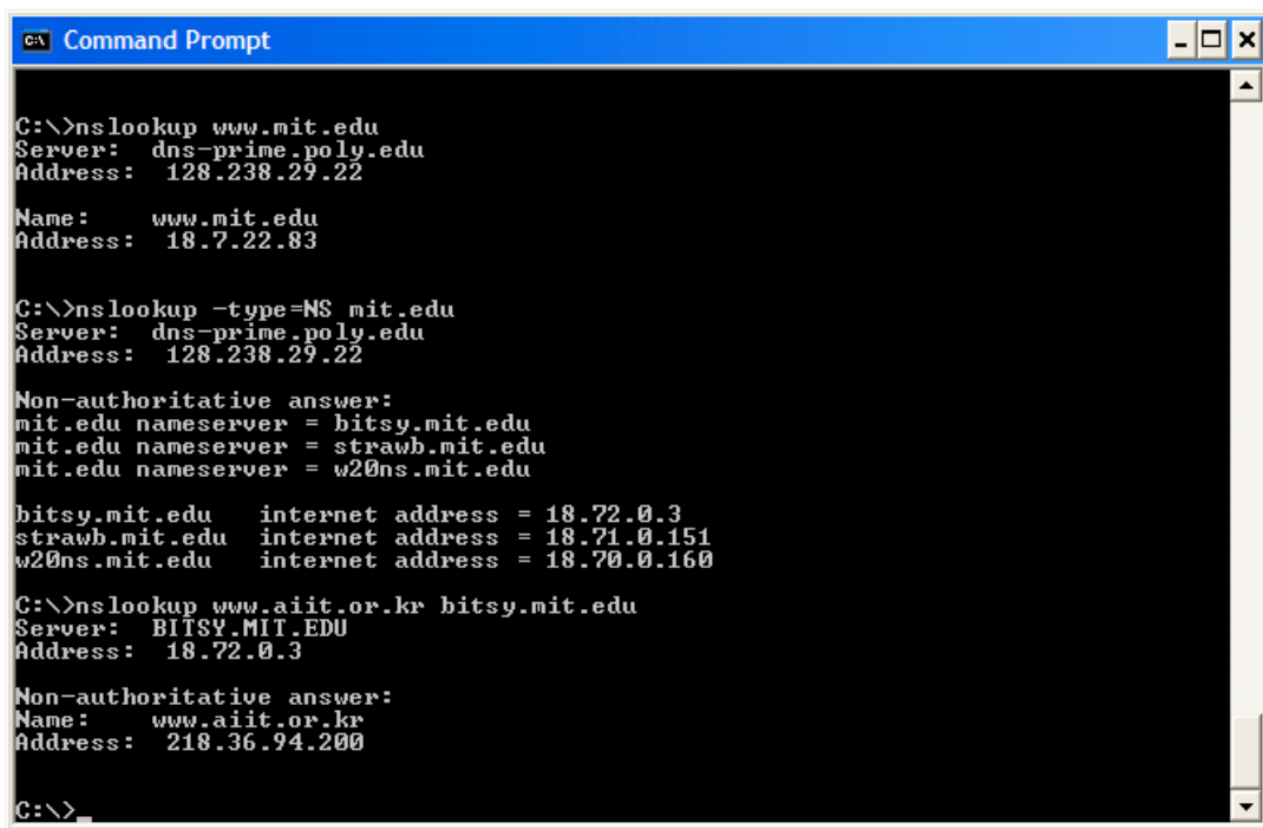
在开始本实验之前，您可能需要阅读书中相关章节来了解 DNS。另外，您可能需要查看关于本地 DNS 服务器，DNS 缓存，DNS 记录和信息，以及 DNS 记录中的 TYPE 字段的资料。

### 二、实验任务

#### 1. nslookup

在本实验中，我们将大量使用 *nslookup* 工具，这个工具在现在的大多数 Linux/Unix 和 Microsoft 平台中都有。要在 Linux/Unix 中运行 *nslookup*，您只需在命令行中键入 *nslookup* 命令即可。要在 Windows 中运行，请打开命令提示符并在命令行上运行 *nslookup*。

在这是最基本的操作，*nslookup* 工具允许主机查询任何指定的 DNS 服务器的 DNS 记录。DNS 服务器可以是根 DNS 服务器，顶级域 DNS 服务器，权威 DNS 服务器或中间 DNS 服务器（有关这些术语的定义，请参阅书本）。要完成此任务，*nslookup* 将 DNS 查询发送到指定的 DNS 服务器，然后接收 DNS 回复，并显示结果。



```
C:\>nslookup www.mit.edu
Server:  dns-prime.poly.edu
Address:  128.238.29.22

Name:     www.mit.edu
Address:  18.7.22.83

C:\>nslookup -type=NS mit.edu
Server:  dns-prime.poly.edu
Address:  128.238.29.22

Non-authoritative answer:
mit.edu nameserver = bitsy.mit.edu
mit.edu nameserver = strawb.mit.edu
mit.edu nameserver = w20ns.mit.edu

bitsy.mit.edu    internet address = 18.72.0.3
strawb.mit.edu   internet address = 18.71.0.151
w20ns.mit.edu    internet address = 18.70.0.160

C:\>nslookup www.aiit.or.kr bitsy.mit.edu
Server:  BITSY.MIT.EDU
Address:  18.72.0.3

Non-authoritative answer:
Name:     www.aiit.or.kr
Address:  218.36.94.200

C:\>
```

上面的屏幕截图显示了三个不同 *nslookup* 命令的结果（显示在 Windows 命令提示符中）。在此示例中，客户端主机位于布鲁克林理工大学校园，默认本地 DNS 服务器为 dns-prime.poly.edu。运行

`nslookup` 时，如果没有指定 DNS 服务器，则 `nslookup` 会将查询发送到默认的 DNS 服务器（在这种情况下为 `dnsprime.poly.edu`）。来看第一个命令：

```
nslookup www.mit.edu
```

说这个命令是说，请告诉我主机 `www.mit.edu` 的 IP 地址。如屏幕截图所示，此命令的响应提供两条信息：（1）提供响应的 DNS 服务器的名称和 IP 地址；（2）响应本身，即 `www.mit.edu` 的主机名和 IP 地址。虽然响应来自理工大学的本地 DNS 服务器，但本地 DNS 服务器很可能会迭代地联系其他几个 DNS 服务器来获得结果。

现在来看第二个命令：

```
nslookup -type=NS mit.edu
```

在这个例子中，我们添加了选项“-type=NS”和域名“mit.edu”。这将使得 `nslookup` 将 NS 记录发送到默认的本地 DNS 服务器。换句话说，“请给我发送 mit.edu 的权威 DNS 的主机名”（当不使用-type选项时，`nslookup` 使用默认值，即查询 A 类记录。）上述屏幕截图中，首先显示了提供响应的 DNS 服务器（这是默认本地 DNS 服务器）以及三个 MIT 域名服务器。这些服务器中的每一个确实都是麻省理工学院校园主机的权威 DNS 服务器。然而，`nslookup` 也表明该响应是非权威的，这意味着这个响应来自某个服务器的缓存，而不是来自权威 MIT DNS 服务器。最后，响应结果还显示了麻省理工学院权威 DNS 服务器的 IP 地址。（即使 `nslookup` 生成的 NS 类型查询没有明确要求 IP 地址，本地 DNS 服务器依然”免费“返回了这些信息，然后被 `nslookup` 显示出来。）

最后来看第三个命令：

```
nslookup www.aiit.or.kr bitsy.mit.edu
```

在这个例子中，我们希望将查询请求发送到 DNS 服务器 `bitsy.mit.edu`，而不是默认的 DNS 服务器（`dns-prime.poly.edu`）。因此，查询和响应事务直接发生在我们的主机和 `bitsy.mit.edu` 之间。在这个例子中，DNS 服务器 `bitsy.mit.edu` 提供主机 `www.aiit.or.kr` 的 IP 地址，它是高级信息技术研究所（韩国）的 Web 服务器。

现在我们了解了一些示例，您可能想知道 `nslookup` 命令的一般语法。语法是：

```
nslookup -option1 -option2 host-to-find dns-server
```

一般来说，`nslookup` 可以不添加选项，或者添加一两个甚至更多选项。正如我们在上面的示例中看到的，`dns-server` 也是可选的；如果这项没有提供，查询将发送到默认的本地 DNS 服务器。

现在我们提供了总览了 `nslookup`，现在是你自己驾驭它的时候了。执行以下操作（并记下结果）：

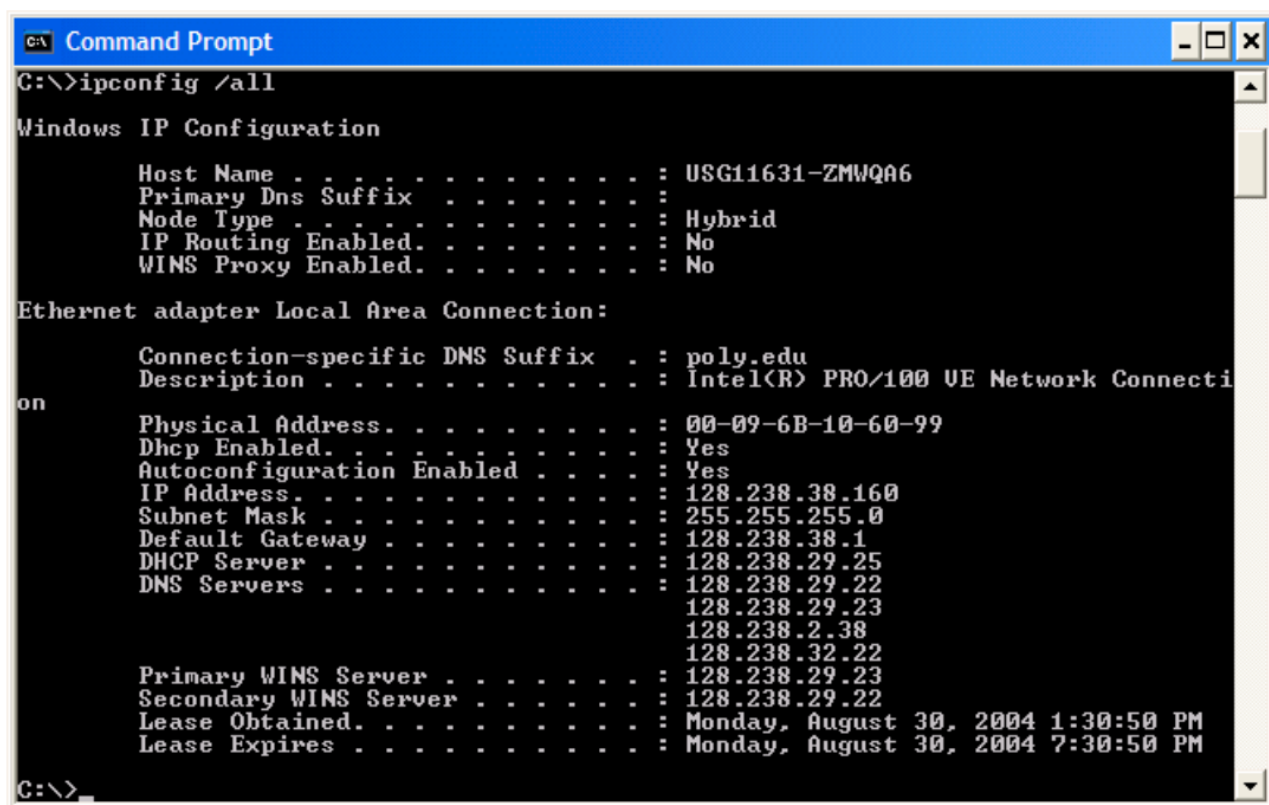
1. 运行 `nslookup` 以获取一个亚洲的 Web 服务器的 IP 地址。该服务器的 IP 地址是什么？（清华大学 <https://www.tsinghua.edu.cn>）
2. 运行 `nslookup` 来确定一个欧洲的大学的权威 DNS 服务器。（牛津大学 <http://www.ox.ac.uk>）
3. 运行 `nslookup`，使用问题 2 中一个已获得的 DNS 服务器，来查询 Yahoo!邮箱的邮件服务器。它的 IP 地址是什么？（可直接查询）

## 2. ipconfig

`ipconfig`（对于 Windows）和 `ifconfig`（对于 Linux / Unix）是主机中最实用的程序，尤其是用于调试网络问题时。这里我们只讨论 `ipconfig`，尽管 Linux / Unix 的 `ifconfig` 与其非常相似。`ipconfig` 可用于显示您当前的 TCP/IP 信息，包括您的地址，DNS 服务器地址，适配器类型等。例如，您只需进入命令提示符，输入

```
ipconfig /all
```

所有关于您的主机信息都类似如下面的屏幕截图所显示。



```
C:\>ipconfig /all

Windows IP Configuration

    Host Name . . . . . : USG11631-ZMWQA6
    Primary Dns Suffix . . . . . : 
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  : poly.edu
    Description . . . . . : Intel(R) PRO/100 UE Network Connection
    Physical Address. . . . . : 00-09-6B-10-60-99
    Dhcp Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 128.238.38.160
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 128.238.38.1
    DHCP Server . . . . . : 128.238.29.25
    DNS Servers . . . . . : 128.238.29.22
                           128.238.29.23
                           128.238.2.38
                           128.238.32.22
    Primary WINS Server . . . . . : 128.238.29.23
    Secondary WINS Server . . . . . : 128.238.29.22
    Lease Obtained. . . . . : Monday, August 30, 2004 1:30:50 PM
    Lease Expires . . . . . : Monday, August 30, 2004 7:30:50 PM

C:\>
```

`ipconfig` 对于管理主机中存储的 DNS 信息也非常有用。我们了解到主机可以缓存最近获得的 DNS 记录。要查看这些缓存记录，在 `C:\>` 提示符后输入以下命令：

```
ipconfig /displaydns
```

每个条目显示剩余的生存时间（TTL）（秒）。要清除缓存，请输入

```
ipconfig /flushdns
```

清除了所有条目并从 `hosts` 文件重新加载条目。

### 3. 使用 Wireshark 追踪 DNS

现在，我们熟悉 `nslookup` 和 `ipconfig`，我们准备好了一些正经的事情。首先让我们捕获一些由常规上网活动生成的 DNS 数据包。

- 使用 `ipconfig` 清空主机中的 DNS 缓存。
- 打开浏览器并清空浏览器缓存。（若使用 Internet Explorer，转到工具菜单并选择 **Internet 选项**；然后在常规选项卡中选择删除文件。）
- 打开 Wireshark，然后在过滤器中输入“`ip.addr==yourIPaddress`”，您可以先使用 `ipconfig` 获取你的 IP 地址。此过滤器将删除既从你主机不发出也不发往你主机的所有数据包。
- 在 Wireshark 中启动数据包捕获。
- 使用浏览器访问网页：<http://www.ietf.org>
- 停止数据包捕获。

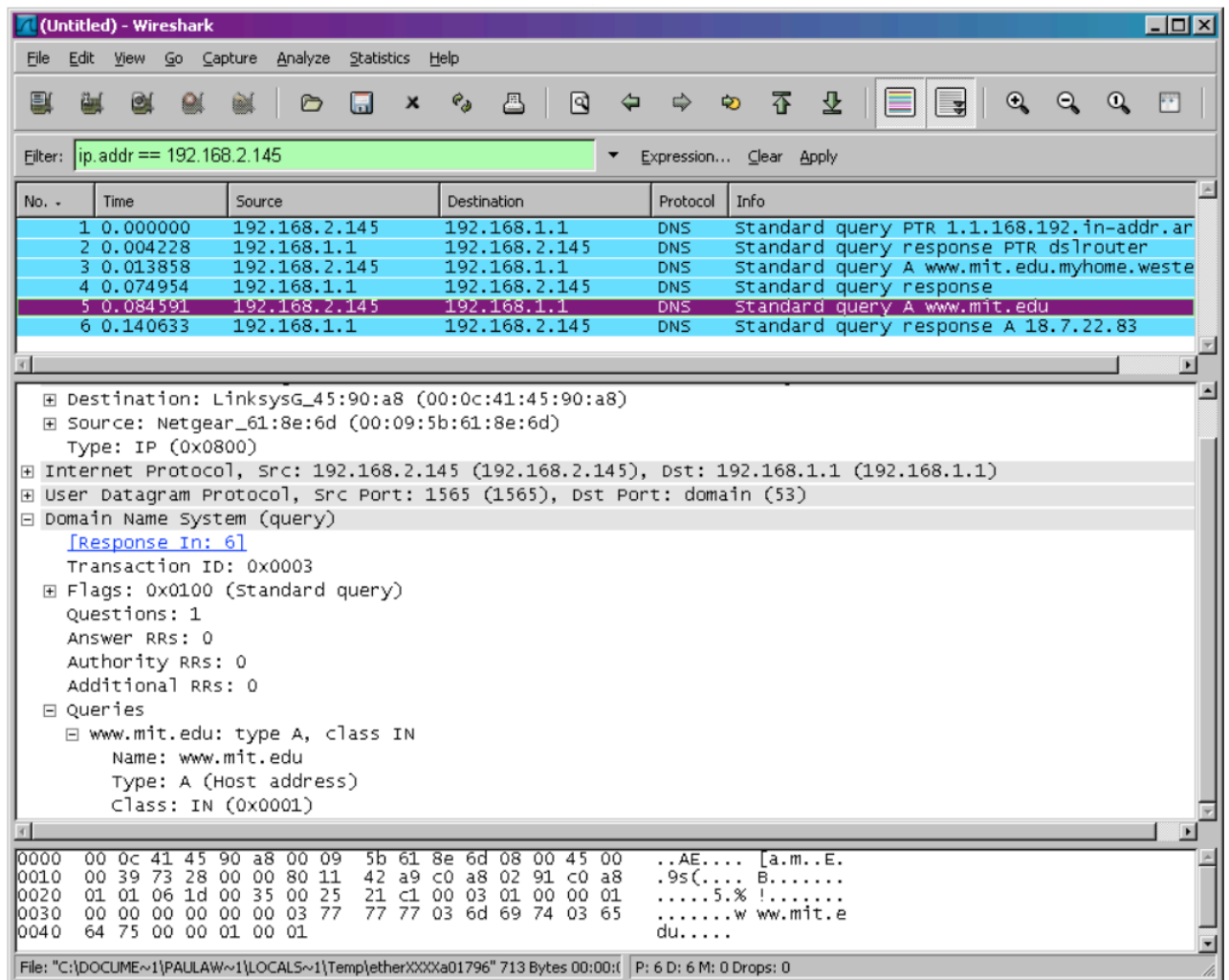
如果您无法在你的网络连接上运行 Wireshark，则可以下载一个捕获了数据包的文件（zip 文件 <http://gaia.cs.umass.edu/Wireshark-labs/Wireshark-traces.zip> 中的跟踪文件 `dns-ethereal-trace`），这个文件是本书作者在自己计算机上按照上述步骤捕获的。回答下列问题。您应该在解答中尽可能展示你使用了哪些你捕获到的数据包，并注释出来。若要打印数据包，请使用文件->打印，只勾选仅选中分组，和概要行，并选中你所需要用于解答问题的数据包。

1. 找到 DNS 查询和响应消息。它们是否通过 UDP 或 TCP 发送？（通过 UDP 发送）



2. DNS 查询消息的目标端口是什么？ DNS 响应消息的源端口是什么？（53）
  3. DNS 查询消息发送到哪个 IP 地址？使用 `ipconfig` 来确定本地 DNS 服务器的 IP 地址。这两个 IP 地址是否相同？
  4. 检查 DNS 查询消息。DNS 查询是什么"Type"的？查询消息是否包含任何"answers"？
  5. 检查 DNS 响应消息。提供了多少个"answers"？这些答案具体包含什么？
  6. 考虑从您主机发送的后续 TCP SYN 数据包。 SYN 数据包的目的 IP 地址是否与 DNS 响应消息中提供的任何 IP 地址相对应？
  7. 这个网页包含一些图片。在获取每个图片前，您的主机是否都发出了新的 DNS 查询？（并不是，只是部分重新发出了新的 DNS 查询）
- 启动数据包捕获。
  - 使用 `nslookup` 查询 `www.mit.edu`
  - 停止数据包捕获。

你应该得到类似下图所示的捕获结果：



我们从上面的屏幕截图看到，`nslookup` 实际上发送了三个 DNS 查询，并收到了三个 DNS 响应。只考虑本次实验相关结果，在回答以下问题时，请忽略前两组查询/响应，因为 `nslookup` 的一些特殊性，这些查询通常不是由标准网络应用程序生成的。您应该专注于最后一个查询和响应消息。

1. DNS 查询消息的目标端口是什么？ DNS 响应消息的源端口是什么？
2. DNS 查询消息的目标 IP 地址是什么？这是你的默认本地 DNS 服务器的 IP 地址吗？
3. 检查 DNS 查询消息。DNS 查询是什么"Type"的？查询消息是否包含任何"answers"？
4. 检查 DNS 响应消息。提供了多少个"answers"？这些答案包含什么？

5. 提供屏幕截图。

现在重复上一个实验，但换成以下命令：

```
nslookup -type=NS mit.edu
```

回答下列问题：

1. DNS 查询消息发送到的 IP 地址是什么？这是您的默认本地 DNS 服务器的 IP 地址吗？
2. 检查 DNS 查询消息。DNS 查询是什么"Type"的？查询消息是否包含任何"answers"？
3. 检查 DNS 响应消息。响应消息提供的 MIT 域名服务器是什么？此响应消息还提供了 MIT 域名服务器的 IP 地址吗？
4. 提供屏幕截图。

现在重复上一个实验，但换成以下命令：

```
nslookup www.aiit.or.kr bitsy.mit.edu
```

回答下列问题：

1. DNS 查询消息发送到的 IP 地址是什么？这是您的默认本地 DNS 服务器的 IP 地址吗？如果不是，这个 IP 地址是什么？（DNS 第一次查询消息发送的 IP 地址是默认的本地域名服务器，查询到 bitsy.mit.edu 的 IP 地址：18.72.0.3，之后向这个 IP 地址发送查询消息，但失败了，因为 MIT 的这个 DNS 服务器已[停用](#)，可直接分析作者的抓包结果）
2. 检查 DNS 查询消息。DNS 查询是什么"Type"的？查询消息是否包含任何"answers"？
3. 检查 DNS 响应消息。提供了多少个"answers"？这些答案包含什么？
4. 提供屏幕截图。

### 三、实验要求

以实验报告的形式把过程截图与答案依次陈列出来，要求独立完成。

