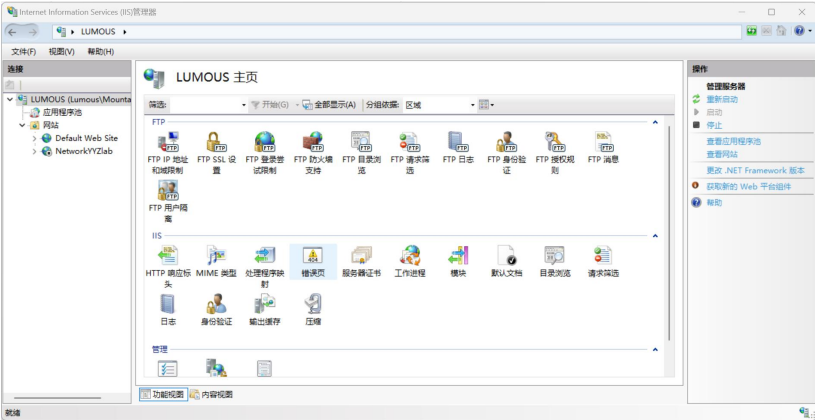
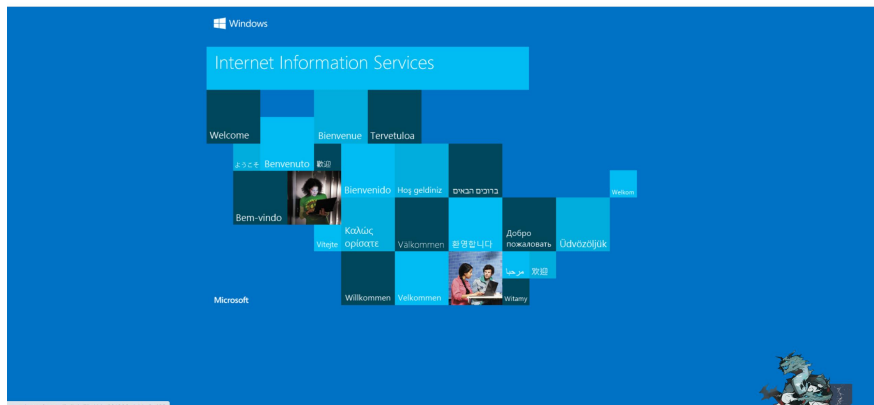


四川大学计算机学院、软件学院

实验报告

学号：2022141460176 姓名：杨一舟 专业：计算机科学与技术 第 6 周

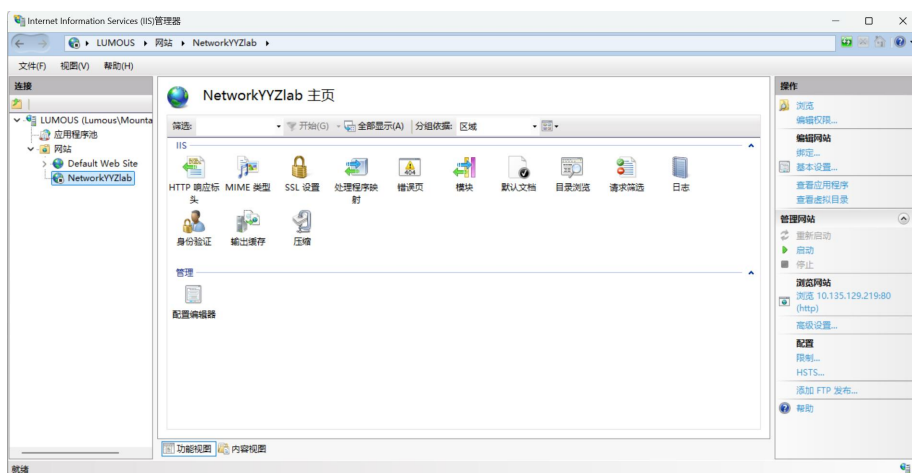
课程名称	计算机网络课程设计	实验课时	2 课时
实验项目	WEB 服务器搭建及 http 协议分析	实验时间	2023 年 10 月 11 日
实验目的	掌握 IIS 组件的安装及在 IIS 下 WEB 服务器的搭建；  Apache 的安装及在 Apache 下 WEB 服务器的搭建；		
实验环境	• Window Server 2008 R2 SP1; Win10, win11  • Apache HTTPd, IIS		
实验内容（算法、程序、步骤和方法）	<div>一、WEB 服务器的搭建</div> <div>1. 安装 IIS 并配置</div> <div></div> <div>这里直接安装并启用了 Windows 控制面板中的 Internet Information Services 功能</div>		



在地址栏输入 `http://localhost/` 出现了以上界面, 说明 IIS 配置成功, 能够正常运行。

## 2. 添加新建网站

在“添加网站”配置界面中, 输入新建网站名称, 这里配置为“Network YYZ lab”; 配置内容目录存放的物理路径“`X:\www`”; 配置 IP 地址以及提供 WEB 服务的端口。



## 3. 配置网站主页

创建一个简单的网页文件并存放到 `X:\www` 路径下, 运行网站后可以看到自己创建的网页。

若要进行快速访问, 请将你的网址收藏放在此收藏夹上。 [立即管理收藏夹](#)

hello world yyz!

## 二、HTTP 协议分析

### 1. Apache 服务器的安装与配置

下载并安装好 Apache 后，在 httpd.conf 文件的 “DocumentRoot” 中修改网站所在的路径

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "X:\\www"
<Directory "X:\\www">|
| #
```

在 httpd.conf 文件的 “DirectoryIndex” 文件中修改网站显示的首页文件

```
#
# DirectoryIndex: sets the file that Apache will serve if a directory
# is requested.
#
<IfModule dir_module>
    DirectoryIndex index.html mainindex.htm|
</IfModule>
```

查找 listen 端口（这里未做修改）

```
...
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 80
```

## 2. 获取 HTTP 协议请求报文及应答报文并分析

- 1) 打开 Wireshark, 并启动 Wireshark 的分组捕获器;
- 2) 在 WEB 浏览器地址栏中输入  
`http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html`
- 3) 停止分组捕获;
- 4) 在过滤器中输入 “HTTP”, 只显示 HTTP 报文。

## 3. 分析第一次抓包的问题

- 1) 浏览器和服务端所运行的 HTTP 版本号是多少?

HTTP1.1

```
> Frame 254: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface \Device\NPF_{EDF25D4C-1883-446...
> Ethernet II, Src: Intel_a9:5d:85 (f4:26:79:a9:5d:85), Dst: RuijieNetwor_4c:47:53 (58:69:6c:4c:47:53)
> Internet Protocol Version 4, Src: 10.134.50.127, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54160, Dst Port: 80, Seq: 1, Ack: 1, Len: 472
< Hypertext Transfer Protocol
  < GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0 Saf
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicat
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    \r\n
    [Response in frame: 264]
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
```

- 2) 当前收到数据包浏览器的语言类型是什么?

支持中文

```
> Frame 254: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface \Device\NPF_{EDF25D4C-1883-446...
> Ethernet II, Src: Intel_a9:5d:85 (f4:26:79:a9:5d:85), Dst: RuijieNetwor_4c:47:53 (58:69:6c:4c:47:53)
> Internet Protocol Version 4, Src: 10.134.50.127, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54160, Dst Port: 80, Seq: 1, Ack: 1, Len: 472
< Hypertext Transfer Protocol
  < GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0 Saf
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicat
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    \r\n
    [Response in frame: 264]
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
```

3) 客户端和服务器的 IP 地址分别是多少?

客户端是 10.134.50.127

服务器是 128.119.245.12

```
> Frame 254: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface \Device\NPF_{EDF25D4C-1883-444...
> Ethernet II, Src: Intel_a9:5d:85 (f4:26:79:a9:5d:85), Dst: RuijieNetwork_4c:47:53 (58:69:6c:4c:47:53)
> Internet Protocol Version 4, Src: 10.134.50.127, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54168, Dst Port: 80, Seq: 1, Ack: 1, Len: 472
> Hypertext Transfer Protocol
  > GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0 Saf
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicat
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    \r\n
    [Response in frame: 264]
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
```

4) 当前截获的数据包的浏览器所支持的压缩方式是什么?

支持的压缩方式是 gzip, deflate

```
> Frame 254: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface \Device\NPF_{EDF25D4C-1883-444...
> Ethernet II, Src: Intel_a9:5d:85 (f4:26:79:a9:5d:85), Dst: RuijieNetwork_4c:47:53 (58:69:6c:4c:47:53)
> Internet Protocol Version 4, Src: 10.134.50.127, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54168, Dst Port: 80, Seq: 1, Ack: 1, Len: 472
> Hypertext Transfer Protocol
  > GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0 Saf
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicat
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    \r\n
    [Response in frame: 264]
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
```

5) 浏览器支持的 MIME 的类型是什么?

支持 text、HTML、application、avif、webp、apng 等

```
> Frame 254: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface \Device\NPF_{EDF25D4C-1883-444...
> Ethernet II, Src: Intel_a9:5d:85 (f4:26:79:a9:5d:85), Dst: RuijieNetwork_4c:47:53 (58:69:6c:4c:47:53)
> Internet Protocol Version 4, Src: 10.134.50.127, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54168, Dst Port: 80, Seq: 1, Ack: 1, Len: 472
> Hypertext Transfer Protocol
  > GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/129.0.0.0 Saf
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicat
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    \r\n
    [Response in frame: 264]
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
```

6) 服务器返回对象最后修改的时间是多少?

服务器返回给浏览器的内容共多少字节?

最后修改时间是 2024 年 10 月 14 日 5:59, 内容共 540 字节

```
> Frame 264: 540 bytes on wire (4320 bits), 540 bytes captured (4320 bits) on interface \Device\NPF_{EDF25D4C-1883-...}
> Ethernet II, Src: RuijieNetwork_4c:47:53 (58:69:6c:4c:47:53), Dst: Intel_a9:5d:85 (f4:26:79:a9:5d:85)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.134.50.127
> Transmission Control Protocol, Src Port: 80, Dst Port: 54160, Seq: 1, Ack: 473, Len: 486
< Hypertext Transfer Protocol
  < HTTP/1.1 200 OK\r\n
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
    Date: Tue, 15 Oct 2024 03:00:09 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Mon, 14 Oct 2024 05:59:01 GMT\r\n
    ETag: "80-62469888dc107"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [Request in frame: 254]
```

7) 通过什么信息可以判断服务器是否成功返回客户端所需要的信息?

Status code: 200 表示服务器成功返回

```
> Frame 264: 540 bytes on wire (4320 bits), 540 bytes captured (4320 bits) on interface \Device\NPF_{EDF25D4C-1883-...}
> Ethernet II, Src: RuijieNetwork_4c:47:53 (58:69:6c:4c:47:53), Dst: Intel_a9:5d:85 (f4:26:79:a9:5d:85)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.134.50.127
> Transmission Control Protocol, Src Port: 80, Dst Port: 54160, Seq: 1, Ack: 473, Len: 486
< Hypertext Transfer Protocol
  < HTTP/1.1 200 OK\r\n
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
    Date: Tue, 15 Oct 2024 03:00:09 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Mon, 14 Oct 2024 05:59:01 GMT\r\n
    ETag: "80-62469888dc107"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [Request in frame: 254]
```

8) 浏览器和服务器之间采用持久连接还是非持久连接的方式工作?如何从截获的数据包中进行判断?

采用持久连接, 因为 connection 为 keep-alive

```
> Frame 264: 540 bytes on wire (4320 bits), 540 bytes captured (4320 bits) on interface \Device\NPF_{EDF25D4C-1883-...}
> Ethernet II, Src: RuijieNetwork_4c:47:53 (58:69:6c:4c:47:53), Dst: Intel_a9:5d:85 (f4:26:79:a9:5d:85)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.134.50.127
> Transmission Control Protocol, Src Port: 80, Dst Port: 54160, Seq: 1, Ack: 473, Len: 486
< Hypertext Transfer Protocol
  < HTTP/1.1 200 OK\r\n
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
    Date: Tue, 15 Oct 2024 03:00:09 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Mon, 14 Oct 2024 05:59:01 GMT\r\n
    ETag: "80-62469888dc107"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [Request in frame: 254]
```

(接上) 实验内容 (算法、程序、步骤和方法)

4. 第二次抓包的实验步骤

- 1) 打开 Wireshark, 启动 Wireshark 分组俘获器;
- 2) 在浏览器地址栏中输入目标网址
- 3) 重新在浏览器地址栏中输入相同的 URL, 再次按回车或点击浏览器中的“刷新”按钮;
- 4) 停止 wireshark 的分组捕获;
- 5) 在过滤器中输入 “HTTP”, 只显示 HTTP 数据包。

5. 分析第二次抓包的问题

- 1) 浏览器向服务器发出的第一个 HTTP GET 请求的内容, 在该请求报文中, 是否有 If-Modified-Since 标题行? 为什么?  
  
第一次 HTTP 请求中没有 If-Modified-Since, 因为此时 GET 请求直接从服务器获取, 不需要判断是否有修改

```
> Frame 264: 540 bytes on wire (4320 bits), 540 bytes captured (4320 bits) on interface \Device\NPF_{EDF25D4C-1883-...}
> Ethernet II, Src: RuijieNetwor_4c:47:53 (58:69:6c:4c:47:53), Dst: Intel_a9:5d:85 (f4:26:79:a9:5d:85)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.134.50.127
> Transmission Control Protocol, Src Port: 80, Dst Port: 54160, Seq: 1, Ack: 473, Len: 486
< Hypertext Transfer Protocol
  < HTTP/1.1 200 OK\r\n
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
    Date: Tue, 15 Oct 2024 03:00:09 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Mon, 14 Oct 2024 05:59:01 GMT\r\n
    ETag: "80-62469888dc107"\r\n
    Accept-Ranges: bytes\r\n
  > Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [Request in frame: 254]
```



2) 浏览器第二次向服务器发出的 HTTP GET 请求的报文, 在该请求报文中, 是否有 If-Modified-Since 标题行? 如果有, 那么在 “IF-MODIFIED-SINCE” 头部之后是什么信息?

第二次请求的报文中含有 If-Modified-Since, 因为它从缓存中读取此页面, 需要判断是否修改过。If-Modified-Since 是表示该网页上次修改的时间

```
> Frame 44: 637 bytes on wire (5096 bits), 637 bytes captured (5096 bits) on interface \Device\NPF...
> Ethernet II, Src: Intel_a9:5d:85 (f4:26:79:a9:5d:85), Dst: RuijieNetwor_4c:47:53 (58:69:6c:4c:47:53)
> Internet Protocol Version 4, Src: 10.134.50.127, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54927, Dst Port: 80, Seq: 1, Ack: 1, Len: 583
< Hypertext Transfer Protocol
  < GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/svg+xml
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    If-None-Match: "80-62469888dc107"\r\n
    If-Modified-Since: Mon, 14 Oct 2024 05:59:01 GMT\r\n
  \r\n
  [Response in frame: 57]
```

3) 在对第二个 HTTP GET 的响应中, 服务器返回的 HTTP 状态码和短语是什么? 服务器是否明确返回了文件的内容?

返回的是 304 Not Modified, 并没有明确返回文件内容, 因为文件内容没有修改, 可以直接从本地缓存中读取

```
> Frame 57: 293 bytes on wire (2344 bits), 293 bytes captured (2344 bits) on interface \Device\NPF...
> Ethernet II, Src: RuijieNetwor_4c:47:53 (58:69:6c:4c:47:53), Dst: Intel_a9:5d:85 (f4:26:79:a9:5d:85)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.134.50.127
> Transmission Control Protocol, Src Port: 80, Dst Port: 54927, Seq: 1, Ack: 584, Len: 239
< Hypertext Transfer Protocol
  < HTTP/1.1 304 Not Modified\r\n
    Response Version: HTTP/1.1
    Status Code: 304
    [Status Code Description: Not Modified]
    Response Phrase: Not Modified
    Date: Tue, 15 Oct 2024 03:21:21 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Connection: Keep-Alive\r\n
    Keep-Alive: timeout=5, max=100\r\n
    ETag: "80-62469888dc107"\r\n
  \r\n
  [Request in frame: 44]
  [Time since request: 0.965134000 seconds]
  [Request URI: /wireshark-labs/HTTP-wireshark-file1.html]
  [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
```



数据记录 和计算	<p>通过本次实验，我们不仅掌握了两种主流 Web 服务器——IIS 和 Apache 的安装与配置方法，还深入研究了 HTTP 协议的工作机制。通过捕获和分析 HTTP 请求与响应报文，我们对 Web 通信的过程有了更直观的认识。</p> <p>在捕获 HTTP 报文的过程中，我们使用了网络嗅探工具 Wireshark，这让我们能够看到客户端与服务器之间的数据交换细节。</p> <p>通过分析这些报文，我们可以理解服务器是如何处理请求的，以及它如何构造响应来发送给客户端。这对于调试 Web 应用和优化网络性能都非常重要。</p>
结论 （结果）	<p>在本次实验中，我们成功安装并配置了 IIS，创建了一个可访问的网站；同样完成了 Apache HTTP Server 的安装与配置，并通过浏览器验证了其功能。通过使用网络嗅探工具，我们捕获到了 HTTP 请求与响应报文，并对其进行详细分析，明确了请求的方法、URI、协议版本以及响应的状态码、头部信息和主体内容等信息。这些结果证实了我们的配置正确无误，并且展示了 HTTP 协议的实际工作流程，达到了预期的学习目标。</p>
小结	<p>本次实验通过安装配置 IIS 和 Apache Web 服务器，并成功搭建了 Web 服务，加深了我们对这两种服务器软件的理解。实验中，我们还使用网络嗅探工具捕获并分析了 HTTP 请求与响应报文，揭示了 Web 通信的具体细节。这不仅巩固了理论知识，也提高了实际操作能力和问题解决技巧。通过这一系列步骤，我们验证了服务器配置的有效性，并且学会了如何利用报文分析来优化和调试 Web 服务。整个过程强调了实践的重要性，并为我们未来的技术应用打下了坚实基础。</p>
指导老师 评议	<div>成绩评定：</div> <div>指导教师签名：</div>

# 实验报告说明

专业实验中心

**实验名称** 要用最简练的语言反映实验的内容。如验证某程序、定律、算法，可写成“验证×××”；分析×××。

**实验目的** 目的要明确，要抓住重点，可以从理论和实践两个方面考虑。在理论上，验证定理、公式、算法，并使实验者获得深刻和系统的理解，在实践上，掌握使用实验设备的技能技巧和程序的调试方法。一般需说明是验证型实验还是设计型实验，是创新型实验还是综合型实验。

**实验环境** 实验用的软硬件环境（配置）。

**实验内容（算法、程序、步骤和方法）** 这是实验报告极其重要的内容。这部分要写明依据何种原理、定律算法、或操作方法进行实验，要写明经过哪几个步骤。还应该画出流程图（实验装置的结构示意图），再配以相应的文字说明，这样既可以节省许多文字说明，又能使实验报告简明扼要，清楚明白。

**数据记录和计算** 指从实验中测出的数据以及计算结果。

**结论（结果）** 即根据实验过程中所见到的现象和测得的数据，作出结论。

**小结** 对本次实验的体会、思考和建议。

**备注或说明** 可写上实验成功或失败的原因，实验后的心得体会、建议等。

**注意：**

- 实验报告将记入实验成绩；
- 每次实验开始时，交上一次的实验报告，否则将扣除此次实验成绩。