

The top corners of the slide feature abstract geometric patterns composed of various colored triangles and polygons in shades of blue, green, and purple. 

# 计算机网络课程设计



# 实验1： WEB服务器的搭建及HTTP协议分析

## 1.实验目标

在Windows环境下，通过使用IIS和Apache两种不同的WEB服务器应用系统搭建WEB服务器，让学生了解服务器的搭建方法。通过对HTTP报文的分析，掌握协议的原理及工作过程。实验过程中，需要掌握以下知识点：

- 1) IIS组件的安装及在IIS下WEB服务器的搭建；
- 2) Apache的安装及在Apache下WEB服务器的搭建；

## 2.实验平台

- Window Server 2008 R2 SP1; Win10, win11

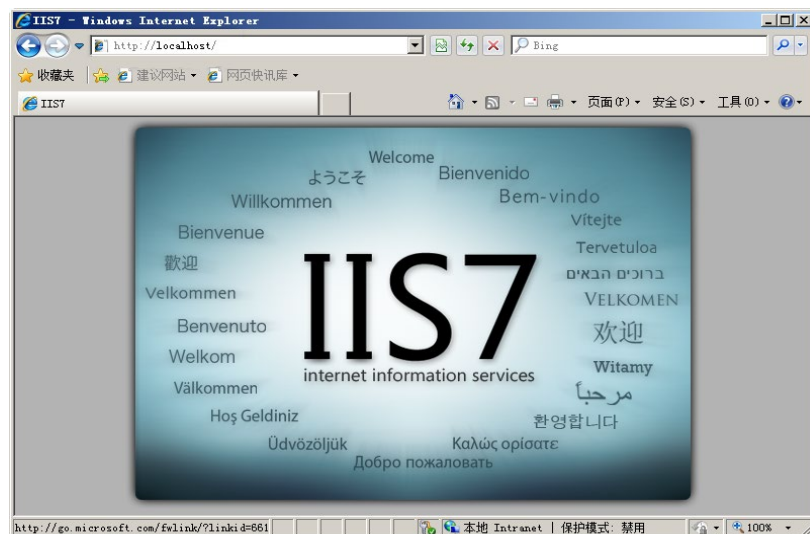
## 3.实验工具

- Apache HTTPd, IIS

# 实验1： WEB服务器的搭建及HTTP协议分析

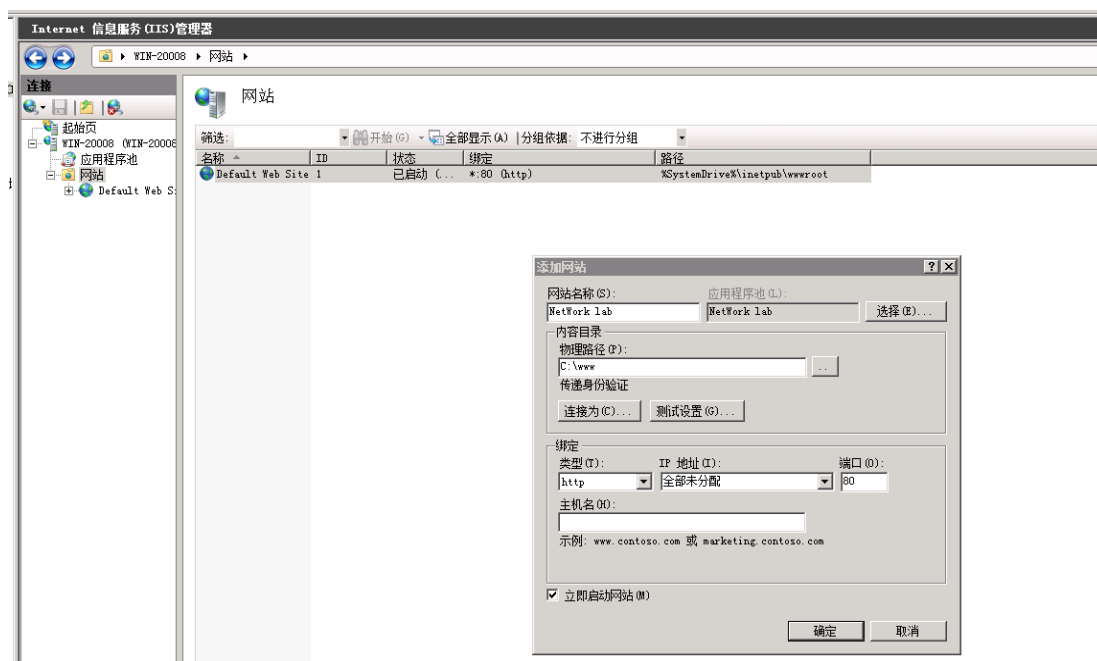
## 4. 实验步骤

- **第一步： IIS下WEB服务器的安装与WEB服务的配置；**
  - 1. 安装IIS，可以通过浏览器搜索IIS的安装方法进行安装。
  - 2. WEB服务安装成功以后，WEB服务器会自动启动一个默认的站点供用户测试。打开浏览器，在地址栏中输入http://localhost，则表明IIS服务器可正常运行，能够提供WEB服务。



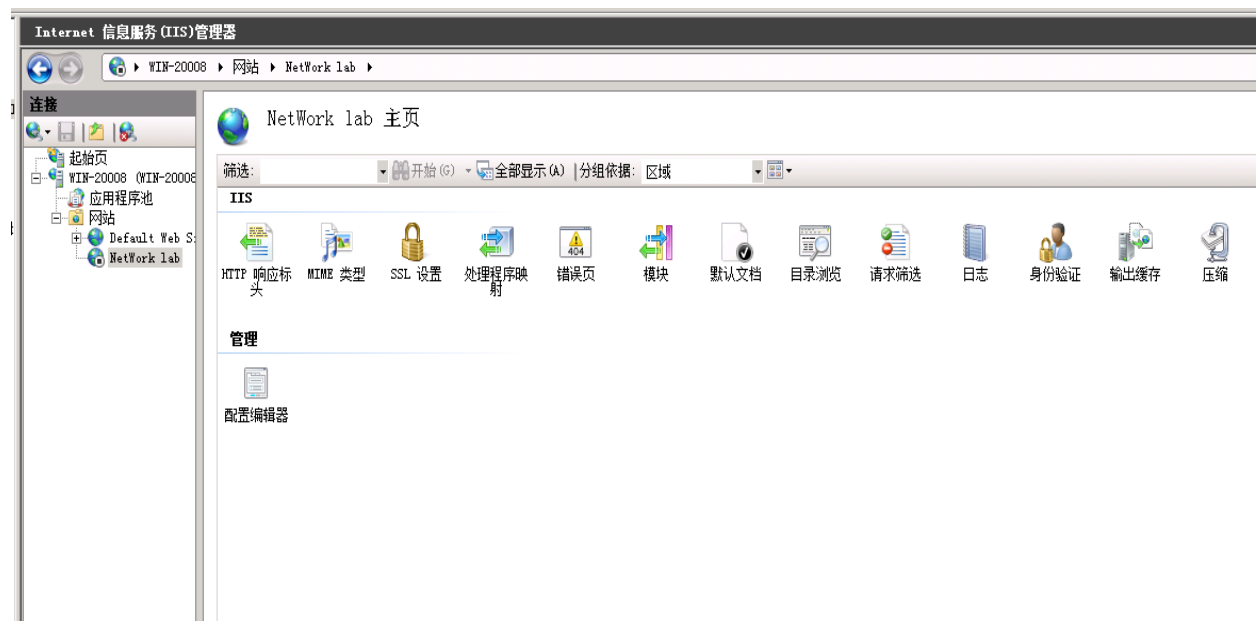
# 实验1： WEB服务器的搭建及HTTP协议分析

- 3.如果要发布自己的网站，可以新建一个WEB站点。在Internet 信息服务（IIS）管理器面板中选择“网站”选项， 点击鼠标右键， 在弹出菜单中选择“添加网站”选项



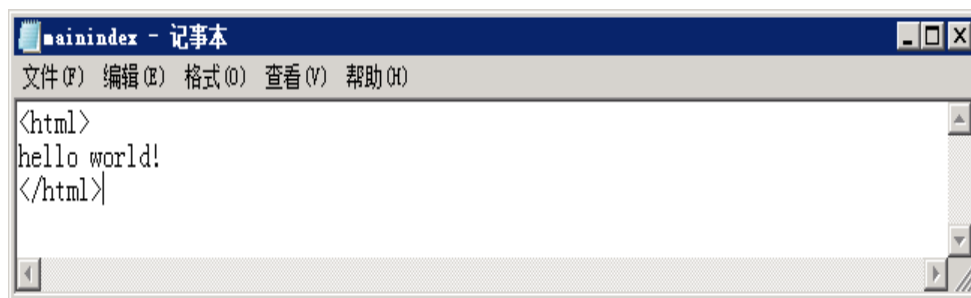
# 实验1： WEB服务器的搭建及HTTP协议分析

- 4. 在“添加网站”配置界面中，输入网站名称“NetWork lab”（实验者可以任意取名）；配置内容目录存放的物理路径“c:\www”（要求网站的所有信息需要存放在该目录下）；配置IP地址以及提供WEB服务的端口。端口可以自行设置，但是一个端口只能提供一个服务。配置了网站的默认基本信息，有的时候还需要配置网站访问时默认的首页文件。双击新建的网站“NetWork lab”



# 实验1： WEB服务器的搭建及HTTP协议分析

- 5. WEB服务的配置完成以后，可以进行WEB服务测试。打开记事本创建一个简单的网页，在记事本中输入如图5-11所示的信息，将文件保存为mainindex.htm，并存放到c:\www目录下。



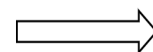
# 实验1： WEB服务器的搭建及HTTP协议分析

## • 第二步： Apache下WEB服务器的安装和WEB服务的配置；

### 1. 下载安装Apache服务器

2. 设置网站根目录：在httpd.conf文件中查找“DocumentRoot”，将其修改为网站所在的路径“c:/www”

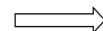
```
DocumentRoot "${SRVROOT}/htdocs"  
<Directory "${SRVROOT}/htdocs">
```



```
DocumentRoot "c:/www"  
<Directory "c:/www">
```

3. 设置网站的首页文件：在httpd.conf文件中查找“DirectoryIndex”所在位置，配置网站的首页文件，多个首页文件可以以半角空格隔开，服务器会根据从左至右的顺序来优先显示

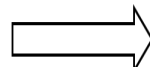
```
# DirectoryIndex: sets the file that Apache will serve if a directory  
# is requested.  
#  
<IfModule dir_module>  
    DirectoryIndex index.html  
</IfModule>
```



```
<IfModule dir_module>  
    DirectoryIndex index.html mainindex.htm  
</IfModule>
```

4. 设置服务器的端口号在httpd.conf查找Listen端口（一般为80端口），也可以将其改为别的端口（如改为8080端口）

```
#Listen 12.34.56.78:80  
Listen 80
```



```
#Listen 12.34.56.78:80  
Listen 8080
```

# 实验1： WEB服务器的搭建及HTTP协议分析

- 第三步： Apache下WEB服务器的安装和WEB服务的配置；
  - (1) 获取HTTP协议请求报文（以GET命令为例）及应答报文并进行分析，具体步骤如下：
    - 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`并回车；
    - 3) 停止分组捕获；
    - 4) 在过滤器中输入“HTTP”，只显示HTTP报文。



# 实验1： WEB服务器的搭建及HTTP协议分析

- 根据捕获的数据报回答问题，实验报告中必须附抓包的截图：
  - 1) 浏览器和服务端所运行的HTTP版本号是多少？
  - 2) 当前收到数据包的浏览器所支持的语言类型是什么？
  - 3) 客户端和服务器的IP地址分别是多少？
  - 3) 当前截获的数据包的浏览器所支持的压缩方式是什么？
  - 4) 浏览器支持的MIME的类型是什么？
  - 5) 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？
  - 6) 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？
  - 7) 浏览器和服务端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？

# 实验1： WEB服务器的搭建及HTTP协议分析

- (2) 获取HTTP协议中Conditional GET报文并分析工作原理

为了获得完整的实验数据，我们在进行WEB数据包跟踪工作之前，必须清空当前主机浏览器的高速缓存。这样做的目的是确保首次捕获的WEB网页数据是从网络传输的，而不是从本地高速缓存中读取的。

- 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（例如：<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html>），然后按回车；
- 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。

# 实验1： WEB服务器的搭建及HTTP协议分析

- 根据捕获的数据报回答问题，实验报告中必须附抓包的截图：

- 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？

- 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？

- 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？