

# 网站架构原理及网站服务搭建

## Web工作原理

客户端与Web服务器之间的通信是通过HTTP协议实现的。Web服务器有时也被称为HTTP服务器或Web容器。HTTP协议采用请求/响应模式，即客户端发起HTTP请求，Web服务器接收并处理该请求，然后将HTTP响应发送回客户端。

## Web服务构成

### 1. 客户端

Web客户端的主要作用是作为用户与Web服务器之间的桥梁。它通过HTTP（超文本传输协议）等网络协议，向服务器发送请求并接收响应，然后将这些响应展示给用户。这种模式被称为请求/响应模式。Web客户端可以是浏览器，如Chrome、Firefox等，也可以是任何使用HTTP协议与Web服务器通信的应用程序。

### 2. 服务器

服务器不间断地等待来自其他设备的请求，并对这些请求做出相应的响应。与个人计算机不同，服务器也具备IP地址，但安装了专门的服务器软件，用于决定如何正确处理和回应来自浏览器等客户端的请求。Web服务器的核心任务是存储、处理网页，并将这些网页发送给请求的客户端。除了Web服务器之外，还有其他种类的服务器，如数据库服务器、应用程序服务器等，它们分别承担不同的服务功能。

#### 2.1 Web服务器和应用程序服务器的区别

##### 1. 功能定位

- **Web服务器**：主要提供静态网页内容，如HTML、图片和CSS文件等。处理HTTP请求并返回相应内容，通常不涉及复杂的逻辑处理。常见的Web服务器有Apache、Nginx和IIS。
- **应用程序服务器**：提供动态网页内容，通过执行业务逻辑来生成内容。可以处理复杂的事务、数据库连接和在线用户会话管理。常见的应用服务器包括Tomcat、JBoss和WebLogic。

##### 2. 处理内容

- **Web服务器**：主要用于处理和响应静态内容请求，例如发送一个HTML页面或图片给客户端。能够进行简单的脚本执行（如CGI、SSI），但通常不用于复杂逻辑处理。
- **应用程序服务器**：执行服务器端脚本和程序以动态生成内容，例如从数据库获取数据并生成HTML。处理业务逻辑，如用户认证、表单提交和数据库操作。

##### 3. 协议支持

- **Web服务器**：主要支持HTTP协议，用于传输Web内容。可以通过扩展支持其他协议，但核心功能集中在HTTP上。
- **应用程序服务器**：不仅支持HTTP，还支持多种通信协议，如RMI/IIOP，用于远程方法调用和对象传输。适用于多种复杂的企业级应用需求。

##### 4. 集成能力

- **Web服务器**：通常具有简单的插件和扩展机制，用于增加特定功能。主要用于静态内容传递，较少涉及与其他服务的深度集成。
- **应用程序服务器**：提供丰富的集成机制，可以与数据库、消息队列和其他服务进行深度集成。支持企业级API，如Java EE的EJB、JPA等。

#### 5. 负载均衡

- **Web服务器**：通常具有负载均衡和反向代理的功能，用于提高系统的可扩展性和可靠性。处理静态内容的性能较高，但在处理高并发请求时可能需要额外的配置和管理。
- **应用程序服务器**：需要强大的并发处理能力，以支持多用户同时访问和数据操作。内置负载均衡和故障转移机制，确保企业应用的高可用性。

#### 6. 职责范围

- **Web服务器**：职责相对单一，专注于内容的分发和简单处理。配置和管理较为简单，维护成本低。
- **应用程序服务器**：职责范围广泛，涵盖从业务逻辑处理到数据持久化等多个方面。配置和管理较为复杂，但提供了更多的灵活性和扩展性。

## 2.2 url的组成

URL (Uniform Resource Locator, 统一资源定位符) 是互联网上标准资源的地址。URL作为网络世界中的基本元素之一，其作用类似于现实生活中的住址。每一个网络资源都有其唯一的URL，这就像每个住所有其独特的地址一样。通过这个"地址"，用户能够精确地找到并访问网络上的资源。URL不仅仅是简单的网络地址，它还包含了关于如何访问这些资源的具体指令。具体介绍如下：

#### 1. 协议

- **概念**：协议部分决定了浏览器与服务器之间的通信方式。例如，http://、https:// 和 ftp:// 都是常见的协议。
- **应用**：每种协议都有其特定的用途。HTTP (HyperText Transfer Protocol) 用于传输网页内容，HTTPS (Hyper Text Transfer Protocol over Secure Socket Layer) 是其加密版，而FTP (File Transfer Protocol) 则用于文件传输。

#### 2. 域名

- **解析**：DNS (Domain Name System) 服务将域名翻译成IP地址，使得用户无需记忆复杂的数字IP地址即可访问网站。

#### 3. 端口号

- **范围**：端口号的范围从0到65535。如果未指定端口号，则使用协议的默认端口，如HTTP的80端口和HTTPS的443端口。
- **功能**：不同的端口可以在同一台服务器上提供不同的服务，从而实现一台服务器上运行多个不同的网络服务。

#### 4. 路径

- **定义**：路径是指向服务器上特定文件或目录的路线图。以斜杠 (/) 开头，依次指示资源所在的文件夹和文件名。
- **应用**：路径可以是物理路径，也可以是经过重写和优化的虚拟路径，后者在现代网站中更为常见。

#### 5. 查询参数

- **格式**：查询参数以问号 (?) 开始，参数之间用和号 (&) 连接。例如：  
param1=value1&param2=value2。

- **用途：** 查询参数常用于传递动态数据给服务器，在动态网页生成和数据库查询中发挥重要作用。

## 6. 锚点

- **标识：** 锚点以井号（#）开头，后跟指定的锚点名称。例如，在网页内跳转时使用，如#section1。
- **应用：** 锚点常用于网页内的快速导航，使用户可以直接跳转到页面的特定位置。

## 7. 可以在URL中使用的字符

- **限制：** 为确保URL的正确解析和传输，大部分URL只能包含ASCII字符集中的字符。但某些特殊字符可以通过编码来使用。
- **编码：** 不安全的字符可以通过百分比编码（Percent-encoding），亦称URL编码进行转义，以确保URL的有效传输。

# 3. 网站搭建

### (1) 基于IP地址

适用于一台服务器有多个IP的情况，但是互联网IP成本比较高，一般不用这种，且目前阿里云的云服务器（ECS）只允许绑定一个公网IP和一个私网IP。

### (2) 基于IP+路径

每个网站放在一个文件目录里面，在浏览的时候，浏览器输入<http://ip/filename>（例如<http://192.168.96.132/pikachu-master>）

### (3) 基于IP+端口号

这种方法使用不同的端口号来识别不同的网站，实际访问时使用网址加端口号的方式来实现，如<http://ip:port>（例如<http://192.168.96.132:8081>）该方式适用于网站域名短缺但服务器的端口号充足的情况，缺点是网站后需要加上端口号，不利于用户访问。

### (4) 基于主机名

这种方法使用不同的域名来区分不同的网站，所有的域名解析都指向同一个IP，Apache通过在HTTP头中附带的host参数来判断用户需要访问哪一个网站，多数情况下多个网站架在一台服务器上大多使用该方法。

## 3.1 非集成工具搭建pikachu环境

### a. 解压pikachu靶场压缩包

```
cd /home/root/Desktop
unzip pikachu-master.zip
```

### b. 移动到apache网站根目录下 (/var/www/html)

```
mv pikachu-master /var/www/html
```

### c.配置数据库

```
//开启mysql数据库服务
service mysql start;

//创建数据库pikachu
create database pikachu;

//创建数据库用户
create user pikachu@localhost identified by 'pikachu';
grant all on pikachu.* to pikachu@localhost;
flush privileges
```

d.修改config.inc.php配置文件，将新建的数据库用户名密码写入配置文件中

e.开启apache服务并初始化靶场

```
//开启apache服务
service apache2 start

//访问127.0.0.1/pikachu-master/install.php进行靶场初始化
```

## 3.2 小皮面板搭建靶场dvwa

a.centos安装小皮面板

```
yum install -y wget && sudo wget -O install.sh
https://dl.xp.cn/dl/xp/install.sh && sudo bash install.sh
```

```
root@localhost:~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
xp面板已启动
小皮面板启动中
=====
小皮面板安装成功
=====面板状态=====
面板状态：运行中
队列状态：运行中
=====面板账户信息=====

面板版本：v1.1.3
外网面板地址：http://219.238.2.102:35216/e2db25
内网面板地址：http://192.168.219.130:35216/e2db25
username: 834a5db2
password: bc283ee0
=====面板注意事项=====

【云服务器】请在安全组放行 35216 端口
面板工具命令：xp
=====
安装花费时间：122 秒
安装日志路径：/root/xp-install.log
[root@localhost ~]#
```

- b.安装nginx和mysql
- C.将网站压缩包上传至网站根目录
- d.修改网站配置