**linux系统之Web基础**

**老男孩老师教学与培训核心思想：重目标、重思路、重方法、重实践、重习惯、重总结。**

# 1 HTTP协议

## 1.1 HTTP协议简介

* HTTP协议，全称Hyper Text Protocol，中文名称超文本传输协议，是互联网中最常用的一种网络协议，HTTP的重要应用就是WWW服务。设计HTTP协议的最初目的就是提供一种发布和接收HTML（一种页面标记语言）页面的方法。
* HTTP协议是互联网上的通信协议方案之一。它有很多的应用，但最流行的就是用于Web浏览器和Web服务器之间的通信应用，即WWW应用或Web应用。
* WWW，全称World Wide Web，简称为Web，中文译为“万维网”。它是目前互联网上最受用户欢迎的信息服务形式。HTTP协议的WWW服务应用的默认服务端口为80，另外的一个加密的WWW服务应用（https）的默认服务端口为443，用于网银、支付等和钱相关的业务。当今，http服务、www服务、Web服务，三者的概念已经混淆了，在本书中也视为相同，都是指当下最常见的网站服务应用。

## 1.2 HTTP协议版本

* HTTP协议从诞生之初到现在经历了几个主要版本：HTTP/0.9，HTTP/1.0，HTTP/1.1，其中，HTTP/1.0是第一个得到广泛使用的版本，而HTTP/1.1为当前使用的主流版本
* HTTP/1.0介绍
* HTTP/1.0是第一个得到广泛使用的HTTP版本。HTTP/1.0在HTTP/0.9的基础上增加了HTTP请求头，支持更多的请求方法，以及对多媒体对象的处理。HTTP/1.0使得包含生动图片的Web页面和交互式表格成为可能，而这些页面和表格促使互联网为人们广泛地接受。HTTP/1.0规定浏览器与服务器只保持短暂的连接，浏览器的每次请求都需要与服务器建立一个TCP连接，服务器完成请求处理后立即断开TCP连接，服务器不跟踪每个客户也不记录过去的请求。
* 2）HTTP/1.1介绍
* HTTP/1.1重点是修复HTTP设计中的缺陷，从可扩展性、缓存处理、带宽优化、持久 、HOST头、错误通知、消息传递、内容协商几个方面做了相关改进。HTTP/1.1是当前互联网主流的HTTP版本。
* 在连接方面，HTTP/1.1支持永久连接，在一个TCP连接上可以传送多个HTTP请求和响应，减少了建立和关闭连接的消耗和延迟。
* 在请求头方面，HTTP/1.1增加更多的请求头和响应头信息，来增强HTTP功能。这使得Web浏览器可以使用主机头名来明确表示要访问服务器上的哪个Web站点，这样就可以使用Web服务器在同一个IP地址和端口号上配置多个虚拟Web站点
* HTTP/1.1的持续连接，也需要增加新的请求头来帮助实现，例如，Connection请求头的值为Keep-Alive时，客户端通知服务器返回本次请求结果后保持连接；Connection请求头的值为close时，客户端通知服务器返回本次请求结果后关闭连接。HTTP/1.1还提供了与身份认证、状态管理和Cache缓存等机制相关的请求头和响应头。

## 1.3 HTTP请求方法

* 在HTTP通信中，每个HTTP请求保本都包含一个方法。用以告诉Web服务器端需要执行什么具体的操作，这些动作包括：获取指定Web页面、提交内容到服务器、删除服务器上资源文件等，这些HTTP请求报文中包含的方法被称为HTTP方法。
* 其中，常用的HTTP方法如下:
* GET 客户端请求指定资源信息，服务器返回指定资源
* HEAD 只请求响应报文中的HTTP首部
* POST 将客户端的数据提交到服务器，例：注册表单
* PUT 从客户端向服务器传送的数据取代指定的文档内容
* DELETE 请求服务器删除指定的资源
* MOVE 请求服务器讲指定的页面移至另一个网络地址

## 1.4 HTTP状态码

* HTTP状态码介绍
* HTTP状态码（HTTP Status Code）是用来表示Web服务器响应http请求状态的数字代码。每当Web客户端向Web服务器发送一个HTTP请求时，Web服务器都会返回一个状态响应代码。这个状态码是一个三位数字代码，作用是告知Web客户端此次的请求是否成功，或者是才去其他的动作方式
* HTTP状态码：
* 100-199 用于指定客户端相应的某些动作
* 200-299 用于表示请求成功
* 300-399 用于已经移动的文件并且常被包含在定位头信息中指定新的地址信息
* 400-499 用于指出客户端的错误
* 500-599 用于指出服务器的错误
* 我们知道，http响应的状态码种类很多，但是在实际工作场景中，经常遇到的状态码却不多。生产场景常见的重要状态码及对应的作用如下：
* 200 - OK 服务器成功返回网页，这是成功的http请求，返回的标准状态码
* 301 - Moved Permanently 永久跳转，所有请求的网页将永久跳转到被指定的新的位置，例如：从etiantian.org跳转到www.etiantian.org
* 403 - Forbidden 禁止访问，这个请求是合法的，但是服务器端因为匹配了预先设置的规则而拒绝响应客户端的请求，此类问题一般为服务器权限配置不当所致
* 404 - Not Found 服务器找不到客户端请求的指定页面，可能是客户端请求了服务器不存在的资源导致
* 500 - Internal Server Error 内部服务器错误，服务器遇到了意料不到的情况，不能完成客户的请求。这是一个比较笼统的报错，一般为服务器的设置或者内部程序问题所致。例如：SElinux开启，而又没有为http设置规则许可，客户访问就是500
* 502 - Bad Gateway 坏的网关，一般是代理服务器请求后端服务时，后端服务不可用或者没有完成响应网管服务器。一般为代理服务器下面的节点出问题导致
* 503 - Service Unavailable 服务当前不可用，可能因为服务器超载或停机维护导致，或者是代理服务器后面没有可以提供服务的节点
* 504 - Gateway Timeout 网关超时，一般是网关代理服务器请求后端服务时，后端服务没有在特定的时间内完成处理请求，一般是服务器过载导致没有在指定的时间内返回数据给代理服务器

[root@web02 php]# curl -I -s -w %{http\_code} -o /dev/null www.etiantian.org

200

#→200即为状态码

## 1.5 HTTP报文

* HTTP报文有很多行内容，不同行的字段都是一些ASCII码串，各个字段的长度也是不同的。
* HTTP报文有两种，一种是从Web客户端发往Web服务器的HTTP报文，称为请求报文（Request Message）。
* 另外一种是从Web服务器发往Web客户端的报文，成为响应报文（Response Message），HTTP请求和响应报文的格式类似

### 1.5.1 HTTP请求报文

* HTTP请求报文有请求行、请求头部（header）、空行和请求报文主体几个部分组成，里面报文信息如下:
* 请求行

请求方法URL协议版本

例：GET /index.html HTTP/1.1

* 请求头

字段名1：值1

字段名2：值2

......

例：

Accept：image/gif，image/jpeg

Accept-Language：zh-cn

......

* 空行

空白无内容

* 请求报文主体

Ps：GET方法没有请求报文主体，POST方法才有

* 下面对HTTP请求报文的每个部分逐一阐述
* 请求行

请求行是请求报文的第一行，用来说明客户端想要做什么。内容由请求方法字段、URL字段和HTTP协议版本字段弓3个字段组成。它们之间用空格分隔。

例如GET /index.html HTTP/1.1

GET：请求方法字段示例

/index.html URL字段示例

HTTP/1.1 HTTP协议版本

* 请求头部
* 空行

最后一个请求头部信息之后是一个空行，发送回车符和换行符，通知Web服务器空行以下不会有请求头部的信息了

* 请求报文主体

请求报文主体包括了要发送给Web服务器的数据信息；请求报文主体不会应用于HTTP的GET命令方法，而是应用于POST方法。POST方法适用于需要客户填写表单的场合。

### 1.5.2 HTTP响应报文

* HTTP响应报文（Response Message）有起始行、响应头部（header）、空行和响应报文主体几个部分组成，和HTTP请求报文格式类似。
* 起始行

协议及版本号 数字状态码 状态信息

例如：HTTP/1.1 200 OK

* 响应头部

字段名1：值1

字段名2：值2

.......

例：

Content-Type:text/html;charset=utf-8

Content-Length:78

.......

* 空行

空白无内容

* 响应报文主体

<html>

<head><title>oldboy's blog</title></head>

<body>

I am oldboy,mysql blog is http://oldboy.blog.51cto.com

</body>

</html>

* 下面对响应报文的每个部分逐一阐述：
* 起始行

响应报文的起始行，也叫状态行，用来说明服务器响应客户端请求的状况。一般为协议及版本号、数字状态码、状态情况。例：HTTP/1.1 200 OK

* 响应头部

和请求报文类似，接着起始行的后面一般有若干个头部字段。每个头部字段都包含一个名字和一个值，两者之间用冒号分隔。头部结尾也是以一个空行结束。常见的头部信息有：

Content-Type:text/html;charset=utf-8

Content-Length:78

....省略...

* 空行

最后一个响应头部信息之后是一个空行，发送回车符和换行符，通知客户端空行下文无头部信息了

* 响应报文主体

响应报文主体中装载了要返回给客户端的数据。这些数据可以是文本及二进制的（如图片、视频），下面是响应的html格式文本数据示例。

<html>

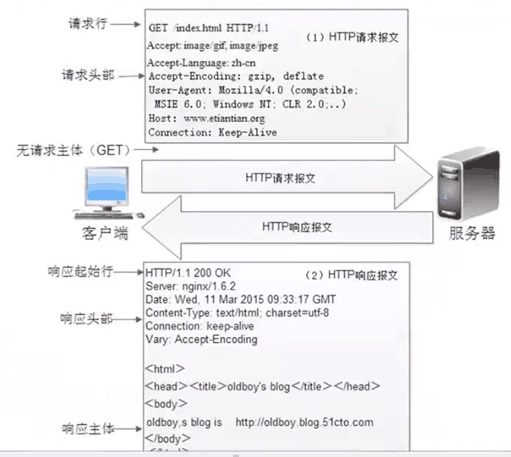
<head><title>oldboy's blog</title></head>

<body>

I am oldboy,mysql blog is http://oldboy.blog.51cto.com

</body>

</html>



## 1.6 HTTP协议原理

HTTP协议属于OSI模型中的第七层应用层协议，HTTP协议的重要应用就是www服务应用，下面就是以www服务应用为例介绍HTTP协议的通信原理，HTTP协议进行通信时，需要有客户端即终端用户，和服务器及web服务器，当web客户端向web服务器发送请求报文之前，先要通过TCP/IP协议在web客户端和服务器之间建立一个TCP/IP连接。整个http协议请求的共组流程步骤如下：

1、终端客户在web浏览器地址输入访问地址：<http://www.etiantian.org>

2、web浏览器请求DNS服务器把域名[www.etiantian.org转换成web服务器的IP](http://www.etiantian.org转换成web服务器的IP)地址，此处的解析过程就是DNS解析的原理流程，上文已详细讲过

3、web浏览器将端口号（默认80）从访问地址（URL）中解析出来

4、web浏览器通过解析后的IP地址及端口号与web服务器之间建立一条TCP连接

5、建立TCP连接后，web浏览器向web服务器发送一条HTTP请求报文，请求报文内容格式及信息细节，HTTP报文一节已经讲到

6、web服务器响应并读取浏览器的请求信息，并返回一条HTTP响应报文，响应报文内容格式及信息细节，前文已讲到

7、web服务器关闭http连接，关闭tcp连接，web浏览器显示访问的网站内容到屏幕

上述就是HTTP协议通信原理过程，整个通信原理的重要知识点有：

1）用户访问网站的流程

2）DNS解析流程细节

3）建立TCP连接后发送HTTP报文的过程

4）HTTP请求报文的细节

5）HTTP响应报文的细节

6）web服务器请求后端集群存储的细节

## 1.7动态web服务请求过程：

客户端发送请求→httpd得到请求→httpd解析请求的格式（html、css）→请求相应php解析→php解析程序执行完毕→db（数据库），返回结果（html）给httpd

→httpd把数据返回给客户端（可能是压缩过的）→浏览器接到返回结果（解压）根据html绘出界面展示出来

## 1.8 HTTP协议重点知识小结

结论：DNS解析原理、http协议原理是高薪面试的重点，高级运维必会知识

1、http协议位于OSI模型中的第七层应用层

2、http协议的重要应用是www服务

3、用户上网流程，DNS解析原理流程

4、DNS解析获取到IP后，建立TCP连接，然后发送http请求细节和服务器响应细节

5、http请求报文与http响应报文知识

6、到达http服务后请求后端集群节点的流程，nginx→FastCGI→php→数据库或存储等

7、老男孩推荐书籍：

a、http权威指南

b、TCP/IP协议卷一

## 1.9媒体（MIME）类型

互联网上的数据有很多种不同的数据类型，web服务器会把通过web传输的每个对象都打上名为MIME类型（MIME type）的数据格式标签。最初设置MIME（Mutipurpose Internet Mail Extension，多用途因特网邮件扩展）是为了解决在不同的电子邮件系统之间搬移报文时存在的问题。MIME在电子邮件系统中工作得非常好，后来HTTP也支持了这个功能，用它来把数据描述并标记不同的数据内容类型。

当web服务器响应HTTP请求时，会为每一个HTTP对象数据加一个MIME类型。当web浏览器获取到服务器返回的对象时，会去查看相关的MIME类型，进行相应处理。

MIME类型存在于HTTP响应报文的响应头部信息里，它是一种文本标记，表示一种主要的对象类型和一个特定的子类型，中间由一条斜杠分割下表位。生产场景最常见的MIME类型：

MIME类型 文件类型

text/htm html htm shtml文本类型

text/css css文件类型

text/xml xml文本类型

image/gif gif图像类型

image/jpeg jpeg jpe图像类型

application/javascript js文本类型

text/plain txt文本类型

application/json json文本类型

video/mp4 mp4视频类型

video/quicktime mov视频类型

video/x-flv flv视频类型

video/x-ms-wmv wmv视频类型

video/x-msvideo avi视频类型

## 1.10 URL介绍

URL，全称Uniform Resource Location，中文翻译为统一资源定位符。也被称为网页地址（网址）。如同在网络上的门派，是因特网上标准的资源的唯一地址。通俗的说，URL是Internet上用来描述信息资源的字符串，主要是用在各种www客户端和服务端程序上。采用URL可以用一种统一的格式来描述各种信息资源，包括文件、服务器的地址和目录等。严格讲，每个URL都是一个URL，它标识一个互联网资源，并制定对其他进行操作或取得该资源的方法。

URL的格式（URL的格式由以下三部分组成）：

第一部分是协议，例：http

第二部分是主机资源服务器IP地址或域名（端口号），例：[www.etiantian.org](http://www.etiantian.org)

第三部分是主机资源和具体地址，如目录和文件名等，例：[www.etiantian.org/index.html](http://www.etiantian.org/index.html)

第一部分和第二部分之间用“://”符号隔开，第二部分和第三部分用/符号隔开。第一部分和第二部分是不可缺少的，第三部分可以省略

列举一个标准的URL及说明

协议 分隔符号 IP地址域名 分隔符号 资源目录地址

http :// [www.etiantian.org](http://www.etiantian.org) / index.html

## 1.11 URI介绍

URI，全称Uniform Resource identitifier，中文翻译为统一资源定位符。是一个用于标识某一互联网资源名称的字符串。这个字符串在世界范围内唯一标识并定位某一个信息资源。互联网上每个可用的数据资源html、图片、视频等皆可通过统一资源标示符进行定位。

## 1.12静态网页

### 1.12.1介绍

在网站设计中，纯粹的HTML格式的网页（可以包含图片、视频、JS（前端功能实现）、CSS等）通常备称为静态网页，早起（大约2002年前）的网站大多都是由静态网页制作的。静态网页是相对于动态网页而言的，是指没有后台数据库、不含程序（php、jsp、asp）和可交互的网页

### 1.12.2特点

静态网页的特点是，开发者编写的是什么，它显示的就是什么，一旦编写完成，就不会有任何的改变。静态网页的维护和更新相对比较麻烦。每个不同的网页都需要单独编辑更新。静态网页一般适用于更新较少的宣传展示型网站，是早起2002年以前多数中小网站展示的形式。

扩展名特征：静态网页资源的对应程序以及资源文件的常见扩展名为：

1）纯文本类程序或文件，html、htm、xml、shtml、js、css等

2）图片类文件或数据文档，jpg、gif、png、bmp、txt、doc、ppt等

3）视频类流媒体文件mp4、swf、avi、wmv、flv等

静态网页资源有几个重要的特征特点：

1）每个页面都有一个固定的URL地址，且URL一般以htm、html、shtml等常见形式为后缀，而且地址中不包含有问号？或&等特殊符号

2）网页内容一经发布到网站服务器上，无论是否有用户访问。每个页面的内容都是保存在网站服务器文件系统上的。也就是说，静态网页是实实在在保存在服务器上的文件实体，每个网页都是一个独立的文件

3）网页的内容相对固定的，因此，容易被搜索引擎收录

4）网页没有数据库的支持，在网站制作和维护方面工作量较大，因此当网站信息量很大时完全依靠静态网页制作的方式比较困难

5）网页的交互性较差，在程序功能实现方面有较大的限制

6）网页程序在用户浏览器端解析，如IE浏览器，这样程序解析效率更高，由于服务端不进行解析，并且不需要读取数据库，因此可以接受更多的并发访问。当客户端向服务器请求数据时，服务器直接把数据从磁盘上返回，当客户端拿到数据后，在浏览器端解析展示出来

常见的静态网页语言有html，js，css，xml，shtml等

静态网页的核心特点

1）客户浏览器端解析程序，不需要读取数据库，性能和效率较高

2）扩展名格式：

a 纯文本类程序或文件，htm、html、xml、shtml、js、css等

b 图片类文件或数据文档，jpg、gif、png、bmp、txt、doc、ppt等

c 视频类流媒体文件 mp4、swf、avi、wmv、flv等

## 1.13动态网页资源

### 1.13.1动态网页资源介绍

所谓的动态网页是与静态网页相对而说的，也就是说，动态网页的URL后缀不是.htm、.html、.shtml、.xml、js、css等静态网页的常见后缀扩展名形式，而是以asp、aspx、php、jsp、do、cgi等形式为后缀，并且一般在动态网页网址中经常会有标志性的符号？、&等，以及大多数情况后端需要数据库支持等。

### 1.13.2动态网页资源特点

1）网页扩展名：常见为asp、aspx、php、jsp、dp、cgi等

2）网页一般以数据库技术为基础，大大降低了网站维护工作的工作量

3）采用动态网页技术的网站可以实现更多的功能，如用户注册、用户登录、在线调查、投票、用户管理、订单管理、发博文等等

4）动态网页并不是独立存在于服务器上的一个个网页文件（php/jsp），当用户请求服务器上的动态程序时，服务器解析程序读取数据库返回一个完整的网页内容

5）动态网页的？对搜索引擎的收录存在一定的问题，搜索引擎一般不会从一个网站的数据库访问全部网页，或者处于技术方面的考虑，搜索蜘蛛一般不会去抓取网址中？后面的内容，因此采用动态网页的网站在进行搜索引擎推广时需要做一定的技术处理（伪静态技术）才能适应搜索引擎的抓取的要求

6）程序在服务端解析，服务端如：php引擎，java容器（tomcat、resin、jboss、weblogic），相当于顾客点餐，饭店里做饭做菜，耗时长，效率低。由于程序在服务器端解析，因此，会消耗大量的CPU和内存等资源，并且多数还要读取数据库等服务，因此，访问效率远不如静态网页。

### 1.13.3动态网页资源核心特点

1、程序在服务端解析，加上后端需要数据库支持，因此访问效率较差

2、动态网站效率很差，并发能力也很低，高并发场景中，应尽可能转换成静态网页提供服务

3、动态转静态也要根据业务需求设计，例如，对于更新频繁的网站会产生数据不一致的情况，即用户看到的数据不会是最新的内容。

4、动态转静态，高并发网站必备的架构方案，是高级架构师的职责所在

## 1.14伪静态网页

### 1.14.1伪静态网页介绍

如标题字义，伪静态就是通过某些技术（如rewrite）把动态网页的URL地址，但实质上用户访问的还是动态网页，只不过给用户看起来URL地址是静态的，因此，用户及某些搜索引擎就会误认为是静态网页。

### 1.14.2伪静态网页特点

从网站URL地址看，伪静态表面上看起来是静态内容（如地址结尾带html），这是通过rewrite规则来实现的URL地址重写，改写后的URL地址规范、美观、有利于搜索引擎抓取，以及用户访问体验加强。如：<http://oldboy.blog.51cto.com/2561410/1712768>

### 1.14.3伪静态网页作用

伪静态网页作用：

由于搜索引擎无法正确读取带参数的动态网页内容，造成网页中很多具有丰富信息的页面无法备搜索引擎收录，伪静态技术的作用是把动态网页URL改写成静态网页的URL，同样是消除了动态网页中的参数，但是并不需要生成任何网页，仅仅是改变了网页地址路径，这样做的主要目的有两个：

一是让搜索引擎收录网站内容让更多的用户访问网站内容

二是提升用户访问体验，动态网页带的特殊符号的URL看起来不如静态网页舒服

### 1.14.4伪静态网页缺点

动态网页伪装静态网页，虽然可以让搜索引擎收录以及提升用户访问体验，但伪静态网页并不能提升网站的访问效率性能，从理论上说还会降低网站的行，没有动态转静态网页效率高，但是对于一些并发不是超大的情况以及硬件资源充足的情况，伪静态还是一个不错的实用功能，因为，要把动态网页真正的转成静态网页，不是一朝一夕的，需要技术团队的通力配合才能行。

现在很多大型的网站都采用了动态网页生成静态网页的技术，消除动态网页中的参数，使搜索引擎收录更多的内容，达到优化网站的效果。生成静态网页确实可以更好的解决搜索引擎收录的问题，并且可以提升网站的访问性能，因此，当网站遇到访问瓶颈时，并且有技术力量去做动态网页生成静态网页的架构设计，那就最好不过了，但是，有些更新频率很快的业务不适合或者做动态转静态有很大难度，可以考虑设置伪静态。

### 1.14.5伪静态网页小结：

1、利于rewrite技术将动态网页伪装成静态网页

2、便于搜索引擎搜索提升用户访问量以及用户体验

3、访问性能没有提升，转换伪静态还会消耗资源，因此性能反而会下降

4、尽可能将动态网页转换城真正的静态页面

5、并发量不是很大或者动态更新过于频繁的时候，用rewrite实现伪静态也是不错的

6、伪静态网页的实现过程，一般是由运营产品提出需求，开发和运维共同实现的

## 1.15生产web架构优化实战方案

由于静态网页程序在客户端解析，大大降低了服务器端的访问压力，因此解析效率更高，在实际高并发网站架构中，我们可以考虑把用户请求的数据解析后存成静态文件放于磁盘中或放于内存中，来降低动态服务器的压力，节约企业成本，提升用户体验。参考文章如下：

<http://oldboy.blog.51cto.com/2561410/736710> 浅谈千万级PV/IP规模高性能高并发网站架构

高并发网站架构从动态转静态不用业务产品实现的例子：

新闻网站的特点是一旦发布完成，几乎不会再改动网页内容，因此，对于新闻的业务内容静态化相对比较简单：

第一步：程序支持发布内容直接转成静态

第二步：运营编辑人员发布新闻网页后，后台程序立刻将动态网页生成静态文件

第三步：运维人员通过发布触发把生成的静态网页发布到事先搭建好的公司缓存集群服务器上，或者把静态内容同步到购买的全国所有CDN服务器节点上，然后，再提供给用户提供访问浏览

视频网站业务（一旦发布完成，几乎不会改动）

优酷视频网，用户上传视频，需要经历审核--转码的过程，大概1-2个小时，审核--转码--推送（同步到全国所有CDN服务器节点）

blog/bbs/sns/微博社区业务/电商（淘宝，京东）（用户发布完成，可能会随时更新）

异步方式，消息中间件、消息队列

## 1.16网站流量度量术语

### 1.16.1 IP（独立IP）

IP（独立IP）：即Internet Protocol，这里指独立IP数，独立IP数是指不同IP地址的计算机访问网站的被算计的总次数。独立IP数是衡量网站流量的一个重要指标。一般一天内相同IP地址的客户端访问网站页面只被计算为一次，记录独立IP的事件可为一天或一个月。目前通用的标准一般为一天。

国内几乎所有的公司上网都是局域网共享上网，即通过路由器NAT地址转换上网，每个计算机在局域网的私有IP是不同的，但是大家出外网上网，就必须都要被路由器把每个私网地址转换成了路由器拨号的公网IP了，所以说，对于网站来说一天内多个相同IP的客户端访问计算为**1个独立IP**。

请问一个客户机用户是通过ADSL等直接拨号上网，但是上网的时候偶尔掉线，一共重新拨号了3次，然后每次都继续老男孩的博客地址，请问此时，网站独立IP数是多少？答：**是3个独立IP**。

由此可见，通过独立IP数度量网站访问量，和实际访问情况不是很匹配，国内的企业、学校等上网多数都是NAT上网的，一个独立IP背后可能有数十上百个客户端访问，独立IP数虽然不是很准确，但却是IT技术人员比较关心的一个衡量网站的指标。

### 1.16.2 PV-Page View

PV（页面访问量）：即Page View，中文翻译即页面浏览，即页面浏览量或点击量，不管客户端是不是相同，也不管IP是不是相同，用户每次访问一个网站页面都会被计算一个PV。

PV的具体度量方法就是从客户浏览器发出一个对Web服务器的请求（Request），web服务器接到这个请求后，将该请求对应的一个网页（Page）发给浏览器，就产生了一个PV。这里有一个问题，就是只要这个请求发送给了浏览器，无论这个页面是否完全打开（或下载完成），那么都是会被计数为1个PV（服务器日志），一般为了防止用户快速刷PV，很多网站把PV的统计程序放在页面的最下面。

用PV衡量网站，PV数反映的是浏览器浏览网站的页面数量，每刷新一次页面也算一次。因此，可以说PV数与来访用户的数量成正比，但PV数并不是页面的来访者数量，二十网站备访问的页面数量，因为一个来访者可能产生多个PV。

问：如果一个用户访问赶集网或58同城，你觉得可能会产生多PV？解答：平均会10几个——几十个PV，一个来访者访问网站的PV数多少是和网站提供的业务直接相关的，例如：这些分类网站，用户浏览网站，可能是为了找房子，找工作，因此一个用户访问的页面会很多，因此PV就会很多。

PV（Page View）是网站备访问的页面数量的一个指标，但不能直接知道有多少人访问了这个网站。

一个来访者访问网站，PV数可能产生若干，但是独立IP数就是为1个，因此，如果对比一个网站的独立IP数和PV数，不难看出，PV数一定会大于独立IP数的，视网站的业务而定，例如这种分类门户可能要达到10：1，甚至更多。

### 1.16.3 UV（Unique Visitor）

UV（独立访客）：即Unique Visitor，同一台客户端（PC或移动端）访问网站备计算为一个访客。一天内相同的客户端访问同一个网站只被计算一次UV。UV一般是以客户端Cookie等技术作为统计依据的，实际统计会有误差。

考虑到一台客户端电脑可能会有多人使用情况，因此，UV实际上并不一定是独立的自然人访问

### 1.16.4企业网站对PV/IP/UV的度量

**IP的度量：**

1、分析所有Web服务器的访问日志信息，对IP地址段去重后计数，这是IT人员的基本计算手段。

2、在网站的每一个（所有）页面结尾，嵌入js等统计程序代码，当用户加载网页后，IP即传给统计IP的服务器，这种方法一般被第三方统计公司或者企业内部开发日志分析程序时使用。

3、用第三方大家比较信任的统计工具例如：谷歌的统计（GA）

IP的统计方法简单、易用，因此，成为了多数网站衡量网站流量的重要指标之一。

**PV的度量：**

1、分析web服务的访问日志（需要排除js、css、各种图片的日志信息），只计算html、php等页面URI。

2、在网站的每一个（所有）页面结尾，嵌入js等统计程序代码，当用户加载网页后，IP即传给统计IP的服务器，这种方法一般备第三方统计公司或者企业内部开发日志分析程序时使用。

3、用第三方大家比较新人的统计工具例如：谷歌的统计（GA）。

PV的统计访问也简单、易用，因此，也是多数网站衡量网站流量的重要指标之一。

提示：IP和PV也是Linux运维人员要掌握的重点

面试题：统计网站某一个时间段所有IP的对应的PV数排序，取前十。

**UV的度量：**

1、通过客户端HTTP请求报文分析

一个客户端会多次请求网站服务器，每次HTTP请求都会携带客户端自身的大量信息，比如：IP地址i，请求发出的时间，浏览器版本，操作系统版本等等。网站服务器对这些请求进行分析，如果这些请求满足一些共同特征，比如来自同一个IP地址，且浏览器版本，操作系统版本相同，请求时间又相近等等，如果满足这一系列的定义，那么就可以认为这些请求是来自同一个客户端，那么多个页面访问也只算一个UV。共同特征的定义是由服务器方决定的。通常，用IP地址+其他特征共同来定义的情况较多。此种度量方法无法解决以下问题，例如：1）多个人的电脑软硬件进程雷同，并且是一个学校或者公司的人；2）多个人共用一个电脑的情况。

2、通过Cookie鉴别

当客户端第一次访问某个网站服务器的时候，网站服务器会给这个客户端的电脑发出一个Cookie，通常放在这个客户端电脑的C盘当中。在这个Cookie中会分配一个独一无二的编号，这其中会记录一些访问服务器的信息，如访问事件，访问了哪些页面等等。当你下次再访问这个服务器的时候，服务器就可以直接从你的电脑中找到上一次放进去的Cookie文件，并且对其进行一些更新，但那个独一无二的编号是不会变的。如果一定时间内，服务器发现2个访者对应的是一个编号，那么我们自然可以认为这个来源于同一个来访者了，就计算1个UV。

使用Cookie的方法要比分析客户端HTTP请求头部信息分析更准确些。但也存在一些问题，比如：有的客户端为保证更高级别的安全，关闭了Cookie的功能；或者有些客户端设置了在退出页面时自动删除Cookie，抑或你经常自己去手动删除Cookie，那么这个方法就不那么精确了。

因此，以上两种方法都只能得到近似的UV，而不是绝对精确的。

UV的度量相对IP和PV来说，不但方法麻烦，而且要开发比较复杂的程序系统计算才能得到期望的结果，因此，Linux运维领域，大家提及的较少，一般企业市场及运维人员可能会关注网站的UV。

### 1.16.5 IP/PV/UV的区别

通过一个访问例子来理解。

例如：某城市的一个网吧里有10个人都进入了[www.etiantian.org的网站，假设每个人平均访问了5](http://www.etiantian.org的网站，假设每个人平均访问了5)个页面，但是这个网吧对外出口是一个公网IP（注意：也可以配置多个IP出口，此处不计特殊情况），所以对于etiantian网站来说，只会就算一个独立IP访问，但是因为网吧里有10人在访问www.etiantian.org的网站，并且平均都访问了5次，因此对于etiantian网站来说，PV数就是10\*5=50个PV，而因为有10个人在访问，就是10个不同的客户端访问，因此，UV（独立访客）为10。

即上述例子中：**网站独立IP数1个，PV数50个，UV数10个。**

通过上述结果，我们能得出一个结论，一个网站的独立IP数量要比网站实际访问的PV数量小的多。通常情况下（国内互联网环境），网站的UV数也会大于独立IP数。

PV数高说明访问的页面数多，但是不一定就代表来访者多；但PV数一定与来访者数量成正比，但是PV并不直接决定页面的真实访者数量。比如一个网站就你一个人进来，通过不断刷新页面，也可以制造出非常高的PV。PV数多，用户访问网站页面的总数量多，通常服务器的压力会大一些。

### 1.16.6并发连接

**网站并发连接**

在面试过程中Linux运维人员经常会被问到：你的公司网站最大并发是多少这样的问题

那么到底什么是并发？怎么理解并发呢？

虽然查阅整个互联网，但还是么有查到让人信服的确切说法：下面是网上的一些说法：

A种理解：网站服务器每秒能够接收的最大用户请求数

B种理解：网站服务器每秒能够响应的最大用户请求数

C种理解：网站服务器在单位时间内能够处理的最大连接数

虽然AB的理解占据了很多IT人员的大多数，但是，按照老男孩的理解，C种理解更为准确一些

举个单位时间段的例子说明：

我们去餐馆吃饭：餐馆里一共有10张桌子，每张桌子最多坐4个人同时吃饭，那么一般人的理解，这个餐馆能够接收的吃饭的并发人数为10\*4，即40个并发，这里就没有考虑到时间问题，1秒并发可以是40个，10分钟内并发可能也是40个。因为这里还有一个因素，就是每个人吃饭时长的问题，如果平均每个人10分钟吃完，那么可以说10分钟内，这个餐馆的并发为40个，而不是每秒钟并发40个，因为第一秒可以是40个人同时进来，但是第二秒就无人可进了（满员了），如果说10分钟并发是40个，下一个十分钟还能是40个，第三个10分钟还可

以是40个。这就是老男孩对并发的理解，即网站服务器在单位时间内能够处理的最大连接数。

对于网站服务器来说，所谓的并发就是单位时间内，服务器能够同时处理的最大连接数，因为有的请求1秒结束，有的请求10秒才结束（业务程序及配置不同），因此，网站并发不是客户端每秒的并发请求数，而是服务器在一段时间内（1秒或者数秒内）可以处理的最大连接数，这个连接包含正在建立的连接，也包含已经建立的连接。

例如：某网站的并发是5000

意味着：单位时间内（理解为1秒或数秒内），正在处理的连接数、正在建立的连接数，加起来一共是5000个。

**其他服务并发连接**

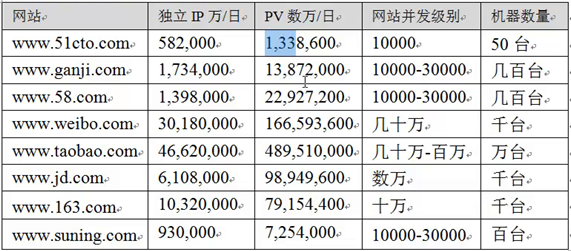
1、QPS每秒查询率（Query Per Second）

每秒查询；率QPS是对一个特定的查询服务器在规定时间内所处理流量多少的衡量标准，运维工作中，DNS系统以及数据库等服务的查询性能经常用每秒查询率来衡量。

2、IOPS（Input/Output Operations Per Second），即每秒进行读写（I/O）操作的次数，多用于数据库场合，衡量随即访问的性能。存储段的IOPS性能和主机端的IO是不同的，IOPS是指存储每秒可接受多少次主机发出的访问，主机一次IO需要多次访问存储才可以完成。例如，主机写入一个最小的数据块，也要经过“发送写入请求、写入数据、收到写入确认”等三个步骤，也就是3个存储端访问。

### 1.16.7常见企业网站排名及PV/IP访问量

表：知名网站访问量信息参考



提示：以上数据来自第三方<http://alexa.chinaz.com/alexa_more.aspx> 网站数据，仅供参考，不同的统计程序差别也很大，有一定的误差，实际访问量要比此表大

### 1.16.8有关网站度量Linux企业运维常见面试题

1、你如何理解并发？网站并发是多少？

2、你们公司网站访问量是多少？怎么计算的？

老男孩的学生一定要解答：IP、PV、并发量3个点，这三个点的多少决定面试时说多大的架构，不能几万PV，还说数十台的集群架构，这样就乌龙了。

**访问量的计算：**

1）运维部分日志分析

2）开发在页面潜入JS程序统计，运营市场通过第三方公司提供的工具程序统计，例如：GA统计。

## 1.17当前互联网主流web服务说明

当前互联网主流web服务软件有

**常见提供静态web服务的软件：**

1）Apache #→这是中小型web服务的主流，web服务器中的老大哥

2）nginx #→大型网站web服务主流，曾经web服务器中的初生牛犊，现已长大。nginx的分支tengine目前也在飞速发展

3）lighttpd #→这是一个不温不火的优秀的web软件，社区不活跃，静态解析效率很高。在nginx流行前，大并发静态业务的首选，国内百度贴吧、豆瓣等众多网站都用过lighttpd

**常见提供动态服务的软件：**

1）php（FastCGI）大中小型网站都会使用，动态网页语言php程序的解析容器。

a、配合Apache解析动态程序，这里的php不是FastCGI守护进程模式，而是mod\_php5.so（module）

b、配合nginx解析动态程序，这里的php常用FastCGI守护进程模式提供服务

2）Tomcat 中小企业动态web服务主流，互联网Java容器主流（jsp，do）

3）resin 大型动态web服务主流，互联网java容器主流（java，do）

4）IIS（Internet information services） 微软的Windows下的web服务软件（asp，aspx）

### 1.17.1 www静态程序服务软件Apache

Apache软件有几个重要的版本系列，分别为：1.3、2.0、2.2、2.4系列，其中1.3和2.0系列已经成为过去时了，官方的网站也看不见其踪影了，目前主流的Apache为2.2系列，正在向2.4系列过度阶段。如果没有特别的要求，建议使用2.2系列。

官方地址：[www.apache.org](http://www.apache.org)

### 1.17.2 www静态服务软件nginx

nginx的版本只有一个系列，但是版本更新很快，仅仅半年就有数个版本，这也看出来社区的活跃程度，具体内容参考官方文档：[www.nginx.org](http://www.nginx.org) 以及 [www.nginx.org/en/docs](http://www.nginx.org/en/docs)，目前的最稳定版本为1.6.3

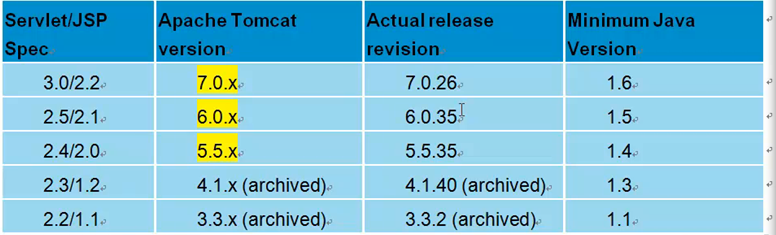
### 1.17.3 www动态服务软件resin

Resin官方号称是世界上最快的web服务，是大型动态web服务主流，互联网java程序的解析器容器，百度、人人都曾经用过Resin。

其发展的主要版本：3.0、3.1、4.0，目前企业较多的使用3.1系列，并正在向4.0过渡。

### 1.17.4 www动态服务软件tomcat

tomcat一直是中小企业动态web服务的主流，常用来作为解析java程序，其版本发展变化如下表：



目前企业使用的主流版本有6系列和7系列，官方也已经推出了更新的8.0系列。

### 1.17.5 www动态服务软件PHP

php软件是中大小型网站程序前台页面开发的首选，存世开源软件众多，也是中小企业网站开发的首选，它是动态网页语言php程序的解析容器。PHP的版本系列有5.2、5.3、5.4、5.5、5.6，其中最经典的版本为5.2系列，企业应用的主流版本可以说是百花争艳。

值得注意的是PHP提供解析的方式，配合Apache解析动态程序，是mod\_php5.so（module）模块方式，配合nginx解析动态程序，常用FastCGI守护进程模式提供服务

官方网站：<http://php.net>

## 1.18 本章重点回顾

1）用户访问网站的基本流程

2）DNS系统的解析原理

3）HTTP协议通信原理

HTTP协议、请求报文、响应报文、状态报文、状态码

4）动态、静态概念特点以及伪静态技术

5）动态转静态web优化方案

6）IP、PV、UV的概念和区别

7）了解常用WWW服务软件特点

Apache、nginx、php（Fastcgi）、tomcat、resin

## 1.19本章知识相关面试考试题

1）请描述DNS系统的解析原理？ #→运维人员必会

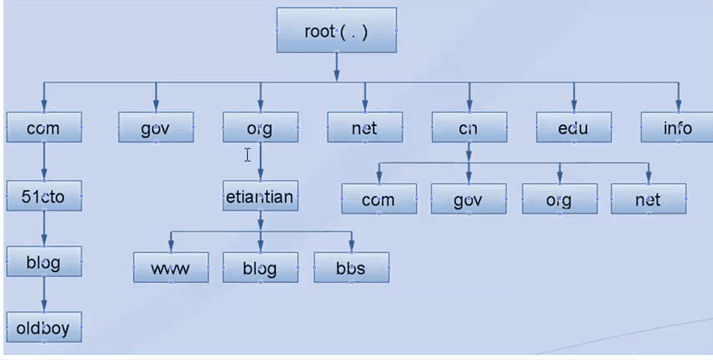
2）请描述HTTP协议的原理？ #→运维人员必会

3）请问你的公司的网站访问量是多少？ #→运维人员必会

# 2 DNS

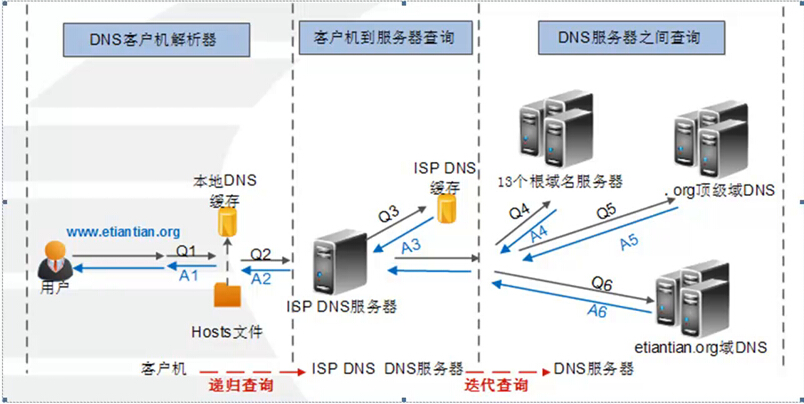
## 2.1 DNS简介

* DNS，全称Domain Name System，DNS在网站运行中气到了至关重要的作用，它的主要作用是负责把网站运行中起到了至关重要的作用，它的主要作用是负责把网站域名解析为对应的IP地址，例如：把www.etiantian.org解析为对应的地址记录如1.1.1.1，这个从域名到IP的解析过程，被称为A记录，即Address Record。
* DNS系统除了负责这个最重要的A记录解析外，还有很多功能，例如：
* 设置CNAME别名记录，这个别名解析功能常被CDN加速服务商应用。
* 设置MX邮件记录，这个MX记录功能，在购买或搭建邮件服务时会被用到
* 设置PTR记录，反向解析，即把IP地址解析为对应的域名，和A记录的解析相反，邮件服务等业务中会用到
* DNS系统的架构很类似于一颗倒挂着的树（和linux系统目录结构类似），他的顶点也是根（“.”），只不过这个根式点（.）来表示，不是目录的跟斜线（/）。
* 整个DNS系统的树状结构如下图所示：



## 2.2 DNS解析流程说明

* DNS的整个解析流程实际上就是从客户端浏览器输入网站地址回车开始，到获取到域名对应IP解析记录的过程，整个过程可以分为如下几个步骤，具体说明如下：
* 第一步：客户端用户从浏览器输入www.etiantian.org网站地址后回车，系统首先查找本地的host文件以及DNS缓存信息，查找是否存在www.etiantian.org对应的IP解析记录，如果有就直接获取到IP地址，然后去访问这个IP地址对应的www.etiantian.org服务器，一般第一次请求时，DNS缓存是没有解析记录的。
* 第二步：如果客户端本地换成或hosts没有对应www.etiantian.org域名的解析记录，那么，系统会把浏览器的解析请求，交给客户端本地设置的DNS服务器地址（通常称此DNS为LDNS，即Local DNS）解析，如果LDNS服务器的本地缓存有对应的解析记录就会直接返回IP地址，如果没有，LDNS会负责继续请求其他的DNS服务器
* 第三步：LDNS会从DNS系统的根（.）开始请求www.etiantian.org域名的解析，根DNS服务器在全球一共有13台服务器，根下面是没有www.etiantian.org域名解析的，但是根下面有www.etiantian.org对应的顶级域.org的解析记录，因此，根会把.org对应的DNS服务器地址返回给LDNS。
* 第四步：LDNS获取到.org对应的DNS服务器地址后，就回去.org服务器请求www.etiantian.org域名的解析，.org服务器下面也没有www.etiantian.org域名的对应的解析记录，但是有etiantian.org域名的解析记录，因此.org服务器会把etiantian.org对应的服务器地址返回给LDNS。
* 第五步：同理，LDNS获取到etiantian.org对应的DNS服务器地址后，就会请求etiantian.org服务器www.etiantian.org域名的解析，etiantian.org域名对应的DNS服务器是该域名授权DNS服务器，这个域名授权DNS服务器正是企业购买域名时管理解析所在的服务器（或者是自建的DNS服务器），这个服务器会有www.etiantian.org对应的IP解析记录，如果此时没有解析记录，就表示企业没有提前为www.etiantian.org
* 第六步：etiantian.org域名对应的授权DNS服务器会把www.etiantian.org对应的IP解析记录发给LDNS
* 第七部：LDNS把收到的来自授权DNS服务器www.etiantian.org对应的IP解析记录发给客户端浏览器，并且在LDNS本地把域名和IP的对应解析缓存起来，以便下一次更快的返回相同解析请求的记录，至此，整个DNS的解析流程就完成了



[root@oldboy ~]# dig +trace www.etiantian.org  
; <<>> DiG 9.3.6-P1-RedHat-9.3.6-20.P1.el5\_8.2 <<>> +trace www.etiantian.org  
;; global options:  printcmd  
.                       25991   IN      NS      m.root-servers.net.  
.                       25991   IN      NS      b.root-servers.net.  
.                       25991   IN      NS      g.root-servers.net.  
.                       25991   IN      NS      k.root-servers.net.  
.                       25991   IN      NS      f.root-servers.net.  
.                       25991   IN      NS      l.root-servers.net.  
.                       25991   IN      NS      d.root-servers.net.  
.                       25991   IN      NS      c.root-servers.net.  
.                       25991   IN      NS      i.root-servers.net.  
.                       25991   IN      NS      h.root-servers.net.  
.                       25991   IN      NS      j.root-servers.net.  
.                       25991   IN      NS      e.root-servers.net.  
.                       25991   IN      NS      a.root-servers.net.  
;; Received 436 bytes from 192.168.120.13#53(192.168.120.13) in 1 ms  
# a-m一共13个root DNS服务器，m.root-servers.net根DNS负责了此次的解析  
org.                    172800  IN      NS      b0.org.afilias-nst.org.  
org.                    172800  IN      NS      d0.org.afilias-nst.org.  
org.                    172800  IN      NS      b2.org.afilias-nst.org.  
org.                    172800  IN      NS      c0.org.afilias-nst.info.  
org.                    172800  IN      NS      a0.org.afilias-nst.info.  
org.                    172800  IN      NS      a2.org.afilias-nst.info.  
;; Received 437 bytes from 202.12.27.33#53(m.root-servers.net) in 136 ms  
#解析org的一共6个DNS服务器，.org的b0.org.afilias-nst.org. DNS负责了此次的解析  
etiantian.org.          86400   IN      NS      dns.bizcn.com.  
etiantian.org.          86400   IN      NS      dns.cnmsn.net.  
;; Received 89 bytes from 199.19.54.1#53(b0.org.afilias-nst.org) in 331 ms  
#解析etiantian.org的一共2个DNS服务器，授权DNS dns.bizcn.com负责了最终的A记录解析  
www.etiantian.org.      600     IN      A       118.26.204.147  
etiantian.org.          600     IN      NS      dns.cnmsn.net.  
etiantian.org.          600     IN      NS      dns.bizcn.com.  
;; Received 169 bytes from 218.93.205.110#53(dns.bizcn.com) in 39 ms

# 3用户访问网站基本流程

