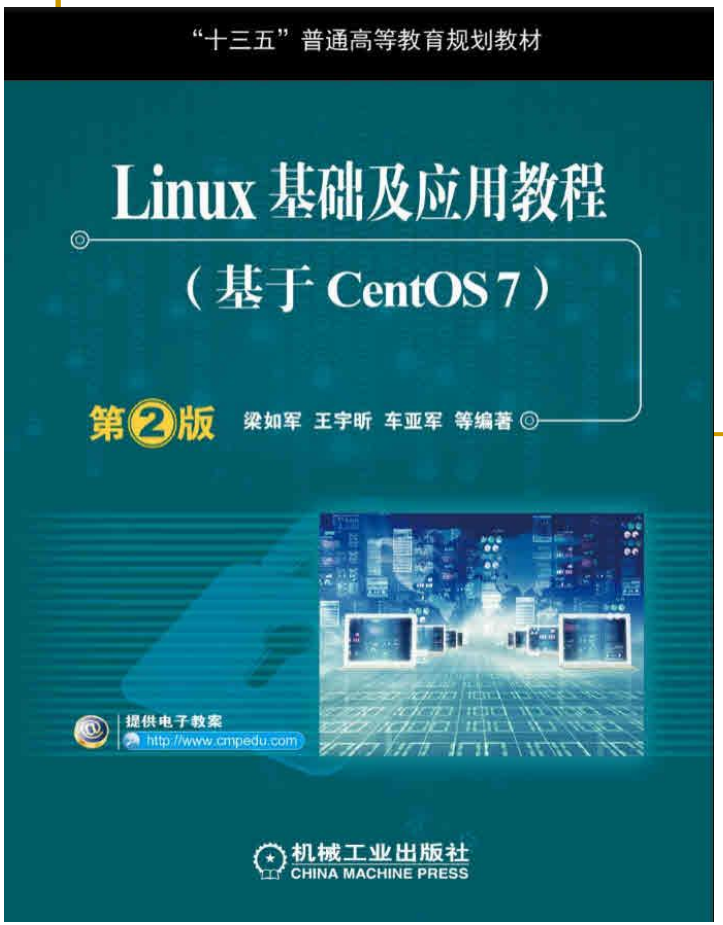


第15章 Apache进阶

主讲人： 梁如军

2015-05-05

本章内容要点

- Linux下的Web编程语言
- Linux下的关系数据库
- Linux下的内存键值数据库
- Apache与CGI
- Apache日志统计分析工具
- LAMP的环境配置及应用
- Apache与Tomcat

本章学习目标

- 熟悉常用的动态网站技术
- 掌握MariaDB (MySQL)的安装与配置
- 掌握Memcached/Redis的安装与配置
- 掌握 Apache 的 CGI 配置
- 掌握 AWStats 的安装和配置
- 掌握 LAMP（PHP模块） 的安装和配置
- 学会安装配置常用的LAMP应用
- 学会安装配置JDK和Tomcat
- 学会配置Aapche反向代理Tomcat

WEB编程语言

梁如军（linuxbooks@126.com）

Linux下的脚本语言

- 常用的脚本语言
  - Bash —— 系统必备
  - PHP —— 简明单纯
  - Perl —— 凝练晦涩
  - Python —— 优雅明晰
  - Ruby —— 精巧灵动
- 脚本语言的主要用途
  - 系统管理的自动化
  - 动态Web编程

梁如军（linuxbooks@126.com）

CentOS 下的脚本语言

- 脚本语言的安装
  - CentOS 官方仓库提供了 Perl/Python/PHP/Ruby
  - 可以使用 yum 安装
- 脚本语言的模块管理工具
  - PHP：**pear**、**pecl**
  - Perl：**cpan**
  - Python：**easy_install**或**pip**
  - Ruby：**gem**

CentOS
The Community ENTerprise Operating System

关系数据库系统

梁如军（linuxbooks@126.com）

动态网站数据库

- 动态Web站点并非一定要有数据库支持，但大多数应用需要数据库支持
- 动态网站常用的开源数据库
  - 关系型数据库（Relational database）
    - **MySQL**：http://www.mysql.com
    - **PostgreSQL**：http://www.postgresql.org/
    - **SQLite**：http://sqlite.org/
  - 面向文档的数据库（Document-oriented database）
    - **mongoDB**：http://www.mongodb.org/
    - **CouchDB**：http://couchdb.apache.org/

梁如军（linuxbooks@126.com）

MySQL/MariaDB数据库简介

- MySQL是一个单进程多线程、支持多用户、基于客户机/服务器（Client/Server简称C/S）的关系数据库管理系统 。
  - 由一个服务器守护程序mysqld和很多不同的客户程序和库组成
  - 支持FreeBSD、Linux、MAC、Windows等多种操作系统平台
- MySQL由瑞典MySQL AB公司开发。
  - 2008年1月MySQL被美国的SUN公司收购。
  - 2009年4月SUN公司又被美国的甲骨文（Oracle）公司收购。

MySQL数据库特点

- 可以同时处理几乎不限数量的用户
- 可以处理拥有上千万条记录的大型数据
- 简单有效的用户特权系统
- 支持常见的SQL语句规范
- 可移植行高，安装简单，小巧
- 良好的运行效率，有丰富信息的网络支持
- 相对其他大型数据库而言调试、管理，优化简单
- 提供多种存储引擎支持，如（MyISAM、InnoDB等）。MySQL5.5默认使用高效的事务引擎InnoDB
- 支持复制功能（Replication）功能，为高可用的MySQL系统提供了可靠方案

MySQL的版本

- 根据MySQL的开发情况，可以将MySQL分为
  - Alpha、Beta、Gamma
  - 和Generally Available（GA）
- MySQL官方为Linux下的每一种GA版本提供了
  - RPM包、二进制包和源码包
  - 为RedHat系列发型版提供了YUM仓库
    - http://repo.mysql.com/yum/
  - 为Debian系列发型版提供了APT仓库
    - http://repo.mysql.com/apt/

MySQL的表类型和存储引擎

- MySQL的表类型由存储引擎（Storage Engines）决定，**针对不同的存储引擎可以指定相应不同的配置**
- MySQL 的表主要支持六种类型
  - 事务安全型(transaction-safe)：**InnoDB**和**BDB**
  - 非事务安全型(non-transaction-safe)：**MYISAM**、**HEAP**、**ISAM**、**MERGE**
- 显示当前数据库支持的存储引擎:
  - show engines;
- MySQL 5.5/5.6 的默认存储引擎是InnoDB

梁如军（linuxbooks@126.com）

MySQL的存储引擎比较

CentOS
The Community ENTerprise Operating System

特点	Myisam	InnoDB	BDB	Memory	Archive
批量插入的速度	高	低	高	高	非常高
事务安全		支持	支持		
全文索引	支持				
锁机制	表锁	行锁	页锁	表锁	行锁
存储限制	没有	64TB	没有	有	没有
B树索引	支持	支持	支持	支持	
哈希索引		支持		支持	
集群索引		支持			
数据缓存		支持		支持	
索引缓存	支持	支持		支持	
数据可压缩	支持				支持
空间使用	低	高	低	N/A	非常低
内存使用	低	高	低	中等	低
支持外键		支持			

梁如军（linuxbooks@126.com）

MyISAM vs InnoDB （1）——MyISAM 的特点

- 数据存储方式简单，使用 B+ Tree 进行索引
- 使用三个文件定义一个表：.MYI .MYD .frm
- 少碎片、支持大文件、能够进行索引压缩
- 二进制层次的文件可以移植 (Linux 、 Windows)
- 访问速度快，是所有MySQL文件引擎中速度最快的
- 不支持一些数据库特性，比如 事务、外键约束等
- 使用表级锁（Table level lock），性能稍差，更适合读取多的操作
- 表数据容量有限，一般建议单表数据量介于 50w–200w

梁如军（linuxbooks@126.com）

MyISAM vs InnoDB （2）——InnoDB 的特点

- 使用表空间（Table Space）的方式来进行数据存储 (ibdata1, ib_logfile0)
- 支持 事务、外键约束等数据库特性
- 使用行级锁（Rows level lock），读写性能都非常优秀
- 能够承载大数据量的存储和访问
- 拥有自己独立的缓冲池，能够缓存数据和索引
- 在关闭自动提交的情况下，与MyISAM引擎速度差异不大

MyISAM vs InnoDB （3）——存储引擎的选择

CentOS The Community ENTerprise Operating System

- 如果应用不需要事务，处理的只是基本的CRUD操作，那么MyISAM是不二选择
  - MyISAM不支持事务、也不支持外键，但其访问速度快
- 一般来说，如果需要事务支持，并且有较高的并发读写频率，InnoDB是不错的选择
  - InnoDB存储引擎提供了具有提交、回滚和崩溃恢复能力的事务安全。
  - 比起MyISAM存储引擎，InnoDB写的处理效率差一些并且会占用更多的磁盘空间以保留数据和索引。

梁如军（linuxbooks@126.com）

MariaDB

- 由原来 MySQL 的作者 Michael Widenius 创办的公司所开发的免费开源的数据库服务
- 是采用 Maria 存储引擎的 MySQL 分支版本
- 与 MySQL 相比较，MariaDB 更强的地方在于，二者支持的不同的引擎。通常可以通过show engines 命令来查看两种数据库服务器 支持的不同的引擎。
- CentOS 7已默认提供了MariaDB而非MySQL

安装MariaDB服务

- 安装

  # yum install mariadb mariadb-server

- 启动

  # systemctl start mariadb

  # systemctl enable mariadb

- 设置MySQL的root用户口令

  # yum install pwgen

  # pwgen -1 20

  Aed7ahBuu7ru2Wooyohg

  # mysqladmin -u root password 'Aed7ahBuu7ru2Wooyohg'

MariaDB服务概览

- 软件包： **mariadb-server**
- 服务类型：由Systemd启动的守护进程
- 配置单元：/usr/lib/systemd/system/mariadb.service
- 端口： 3306
- 配置： /etc/my.cnf
- 相关软件包： **mariadb**、php-mysql、perl-DBD-mysql

梁如军（linuxbooks@126.com）

MariaDB/MySQL的配置文件

- 配置文件为/etc/my.cnf

- MariaDB/MySQL的详细配置参数的解释请参考MariaDB/MySQL手册
- 建议DBA从头编制适合特定应用的配置文件

MariaDB/MySQL的配置原则

- 针对不同的服务器硬件进行合理配置（如CPU核数、内存大小等）
- 针对是否是独立的服务器进行合理配置（若Mysql服务器还同时运行其他服务，就该适当削减其资源占用）
- 针对 MyISAM 或 InnoDB 不同引擎进行不同定制性配置
- 针对不同的应用情况进行合理配置，尽量使应用本身达到最合理的情况

梁如军（linuxbooks@126.com）

MySQL常用的公共配置选项

选项	缺省值	推荐值	说明
max_connections	100	1024	MySQL服务器同时处理的数据库连接的最大数量
query_cache_size	0（不打开）	16M	查询缓存区的最大长度，按照当前需求，一倍一倍增加，本选项比较重要
sort_buffer_size	512K	16M	每个线程的排序缓存大小，一般按照内存可以设置为2M以上，推荐是16M，该选项对排序order by，group by起作用
record_buffer	128K	16M	每个进行一个顺序扫描的线程为其扫描的每张表分配这个大小的一个缓冲区，可以设置为2M以上
table_cache	64	512	为所有线程打开表的数量。增加该值能增加mysqld要求的文件描述符的数量。MySQL对每个唯一打开的表需要2个文件描述符。

梁如军（linuxbooks@126.com）

常用的 MyISAM 配置选项

选项	缺省值	推荐值	说明
key_buffer_size	8M	256M	用来存放索引区块的缓存值，建议128M以上，不要大于内存的30%
read_buffer_size	128K	16M	用来做MyISAM表全表扫描的缓冲大小.为从数据表顺序读取数据的读操作保留的缓存区的长度
myisam_sort_buffer_size	16M	128M	设置,恢复,修改表的时候使用的缓冲大小，值不要设的太大

梁如军（linuxbooks@126.com）

常用的 InnoDB 配置选项

选项	缺省值	推荐值	说明
innodb_buffer_pool_size	32M	1G	InnoDB使用一个缓冲池来保存索引和原始数据，这里你设置越大,你在存取表里面数据时所需要的磁盘I/O越少，一般是内存的一半，不超过2G，否则系统会崩溃，这个参数非常重要
innodb_additional_mem_pool_size	2M	128M	InnoDB用来保存 metadata 信息，如果内存是4G，最好本值超过200M
innodb_flush_log_at_trx_commit	1	0	0 代表日志只大约每秒写入日志文件并且日志文件刷新到磁盘； 1 为执行完没执行一条SQL马上commit； 2 代表日志写入日志文件在每次提交后，但是日志文件只有大约每秒才会刷新到磁盘上. 对速度影响比较大，同时也关系数据完整性
innodb_log_file_size	8M	256M	在日志组中每个日志文件的大小，一般是innodb_buffer_pool_size的25%，官方推荐是innodb_buffer_pool_size 的 40-50%，设置大一点来避免在日志文件覆写上不必要的缓冲池刷新行为
innodb_log_buffer_size	128K	8M	用来缓冲日志数据的缓冲区的大小. 推荐是8M，官方推荐该值小于16M，最好是 1M-8M 之间

梁如军（linuxbooks@126.com）

配置MySQL（/etc/my.cnf）

CentOS
The Community ENTerprise Operating System

```
[mysqld]
# GENERAL
datadir = /var/lib/mysql
socket = /var/lib/mysql/mysql.sock
pid_file = /var/lib/mysql/mysql.pid
user = mysql
port = 3306
storage_engine = InnoDB
# INNODB
innodb_buffer_pool_size = <value>
innodb_log_file_size = <value>
innodb_file_per_table = 1
innodb_flush_method = O_DIRECT
# MyISAM
key_buffer_size = <value>
# LOGGING
log_error = /var/lib/mysql/mysql-error.log
log_slow_queries = /var/lib/mysql/mysql-slow.log
# OTHER
tmp_table_size = 32M
max_heap_table_size = 32M
query_cache_type = 0
query_cache_size = 0
max_connections = <value>
thread_cache_size = <value>
table_cache_size = <value>
open_files_limit = 65535

[client]
socket = /var/lib/mysql/mysql.sock
port = 3306
```

MySQL参考资源

- 书籍
  - 高性能MySQL（第3版）
    - http://www.highperfmysql.com/
  - Effective MySQL 系列（3册，有中译本）
    - http://effectivemysql.com/
- 配置
  - 在线MySQL配置向导
    - https://tools.percona.com/wizard

键值缓存系统

梁如军（linuxbooks@126.com）

键值缓存系统简介

CentOS
The Community ENTerprise Operating System

- 通常内存键值缓存系统与数据库相结合来使用
  - 通过在内存中缓存数据和对象来减少读取关系数据库的次数，从而提高了动态数据库驱动的网站速度。
  - 若被访问的对象已在缓存中，则直接读取缓存中的数据返还给浏览器；
  - 若未在缓存中，则访问后端数据库查询获取数据并返还给浏览器，同时将查询结果置于缓存系统中以便加快后续访问。

梁如军（linuxbooks@126.com）

键值缓存系统简介（续）

- 键值缓存系统通常占用固定大小的内存来运行
  - 当内存中的缓存被占满后会使用最近最少用（LRU）算法自动移除一些缓存对象。
- 内存键值缓存系统通常是基于C/S模型设计的，即包含服务器端和客户端
  - 服务器端是以守护进程形式运行的
  - 提供基于不同语言（如：PHP、Python、Ruby、Java等）的多种客户端

梁如军（linuxbooks@126.com）

为什么使用键值缓存系统

- 为了加快Web站点的响应速度，通常可以使用基于内存的键值缓存系统：
  - 缓存经常被访问的静态HTML页面、CSS、Javascript、图片
  - 缓存用于生成动态页面的，渲染后的网页模板（Renderred Templates）
  - 缓存登录Cookie/Session、购物车
  - 缓存动态热点数据（从数据库获得的查询结果）等

梁如军（linuxbooks@126.com）

Memcached简介

- 类型
  - 内存键值缓存
- 数据存储
  - 将键直接映射为值
- 操作方法
  - 创建、读取，更新，删除等
- 其他特性
  - 使用多线程为服务器提供额外的性能

梁如军（linuxbooks@126.com）

Memcached服务概览

- 软件包： memcached
- 服务类型：由Systemd启动的守护进程
- 配置单元：/usr/lib/systemd/system/memcached.service
- 守护进程： /usr/bin/memcached
- 端口： 11211
- 配置： /etc/sysconfig/memcached
- 工具： /usr/bin/memcached-tool
- 相关软件包： php-pecl-memcached，……

梁如军（linuxbooks@126.com）

安装、配置Memcached服务

- 安装Memcached（使用Remi仓库中的新版）

  # yum install memcached

- 配置Memcached

  # vi /etc/sysconfig/memcached

```
PORT="11211“        # 定义监听端口变量，作为 –p 参数的值使用
USER="memcached" # 定义守护进程的执行用户变量，作为 –u 参数的值使用
MAXCONN="5000"  # 定义最大并行连接数变量，作为 –c 参数的值使用
CACHESIZE="1024" # 定义缓存最大尺寸的变量，作为 –m 参数的值使用（单位为MB）
OPTIONS=""          # 定义其他memcached命令行上可用的参数及其值
```


梁如军（linuxbooks@126.com）

启动Memcached服务

- 启动Memcached

  # systemctl start memcached

  # systemctl enable memcached

- 使用Memcached的管理工具

  # man memcached-tool

- 显示服务器的当前状态信息

  # memcached-tool 127.0.0.1:11211 status

梁如军（linuxbooks@126.com）

Redis简介

- 类型
  - 内存非关系数据库
- 数据存储
  - 将键映射为字符串(String)、哈希(Map)、列表(list)、集合(sets)和有序集合(sorted sets)等类型的值
- 操作方法
  - 提供了对于每种数据类型的通用访问模式，为每种数据类型的处理提供大量命令以及部分事务支持
- 其他特性
  - 发布/订阅、主/从复制、持久性（disk-backed）、脚本（存储过程）

梁如军（linuxbooks@126.com）

Redis服务概览

- 软件包： redis
- 服务类型：由Systemd启动的守护进程
- 配置单元： /usr/lib/systemd/system/redis.service
- 守护进程： /usr/sbin/redis-server
- 端口： 6379
- 配置： /etc/redis.conf
- 工具： /usr/bin/{redis-cli, redis-benchmark}
- 相关软件包： php-pecl-redis，……

梁如军（linuxbooks@126.com）

安装、配置Redis服务

- 安装Redis（使用Remi仓库中的新版）

  # yum install redis

- 配置Redis

  # vi /etc/redis.conf

```
port 6379                          # 指定监听端口
maxclients 5000                    # 指定最大并行连接数
maxmemory 1gb                      # 指定缓存最大尺寸
maxmemory-policy allkeys-lru       # 指定缓存满后的剔除策略
```


梁如军（linuxbooks@126.com）

启动Redis服务

- 启动Redis

 # systemctl start redis

 # systemctl enable redis

- 使用Redis工具
  - 查看redis的当前状态信息

 # redis-cli -h localhost -p 6379 info:

  - 监视redis的读写操作（<Ctrl+C>退出）

 # redis-cli -h localhost -p 6379 monitor

  - 性能测试（100个并发连接，10000个请求）

 # redis-benchmark -h localhost -p 6379 -c 100 -n 10000

Apache的动态网站技术

- **CGI**
- **Apache的第三方脚本语言模块**
- FastCGI

CGI 简介

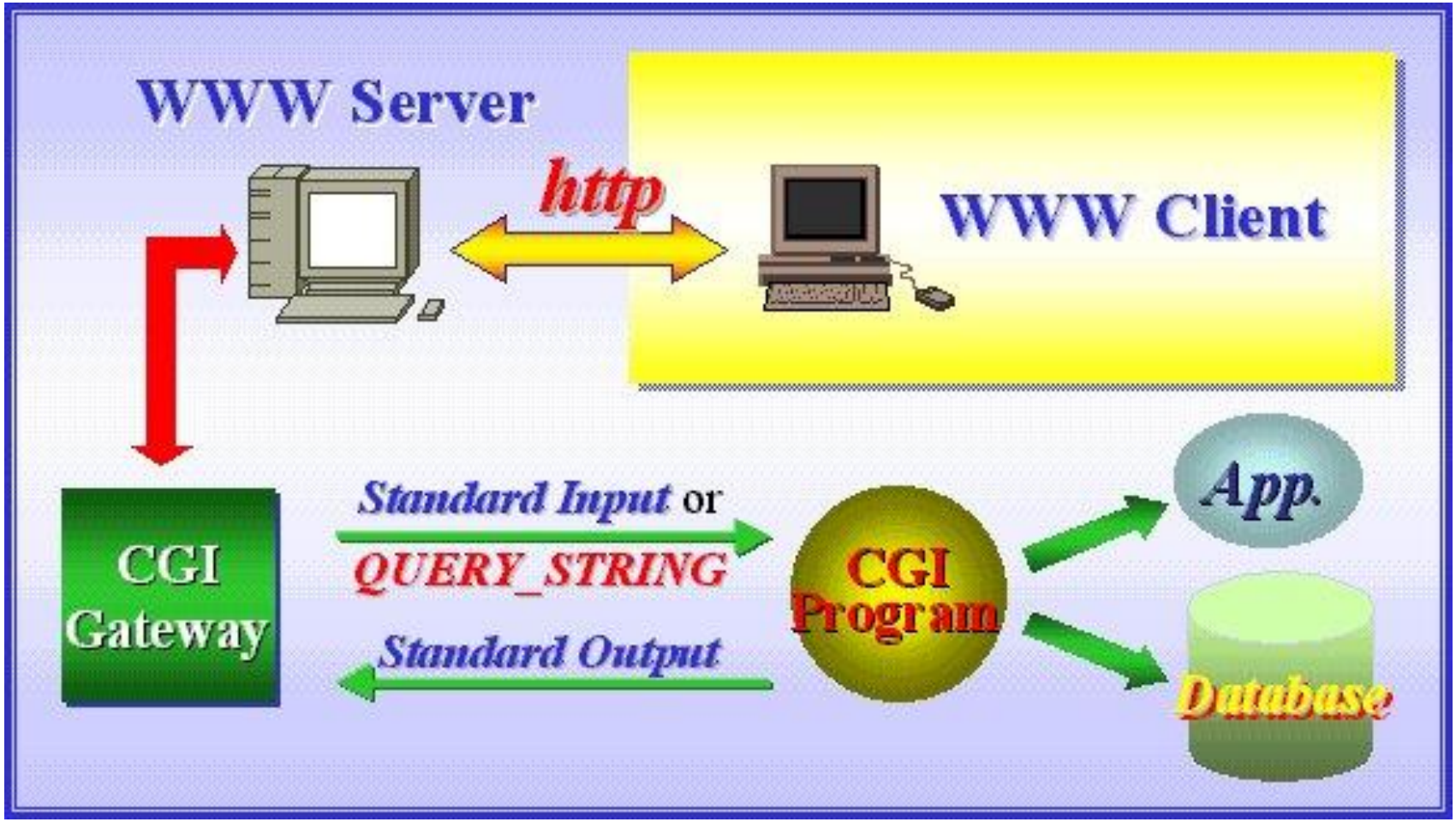
- CGI（Common Gateway Interface，通用网关接口） 是一个连接外部应用程序到 HTTP 服务器的标准
- CGI 定义了 Web 服务器与外部内容生成程序（通常称为 CGI 脚本或 CGI 程序）之间交互的方法，即：一种基于浏览器的输入、在 Web 服务器上运行的程序方法，从而实现动态 Web 的功能

CGI 程序

- CGI 程序可以用任何一种语言编写
  - 只要这种语言具有标准输入、输出和环境变量。
  - 例如：perl、python、ruby、php、bash、C 等
- CGI 程序通常是挂平台的
  - 可以运行在类 UNIX 和 Windows 等众多平台的服务器上
  - 实现同一功能的程序在不同平台上可能会有细微差异

梁如军（linuxbooks@126.com）

CGI 的工作原理



梁如军（linuxbooks@126.com）

CGI 的处理步骤

- Web 客户端通过网络把用户请求送到服务器
- Web 服务器接收用户请求
  - GET 方法：利用环境变量 QUERY_STRING 接收
  - POST 方法：利用标准输入接收；环境变量 CONTENT_LENGTH记录输入字符长度
- Web 服务器交给 CGI 程序处理
- CGI 程序把动态处理结果通过标准输出传送给 Web 服务器
- 服务器把最终的HTML页面送回到 Web 客户端

梁如军（linuxbooks@126.com）

Apache 与 CGI

- Apache支持CGI的模块
  - mod_cgi（用于基于进程的 prefork MPM）
  - mod_cgid（用于基于线程的 worker MPM）
- RHEL/CentOS下Apache默认加载了mod_cgi
- 配置 Apache 允许执行 CGI 程序有两种方法
  - 将所有的 CGI 程序放在指定的目录中，并使用 ScriptAlias 指令声明
  - 在任意目录中执行 CGI 程序

梁如军（linuxbooks@126.com）

ScriptAlias

- **/etc/httpd/conf/httpd.conf** 里有如下的配置段

```
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
<Directory "/var/www/cgi-bin">
    AllowOverride None
    Options None
    Require all granted
</Directory>
```

  - 以 /cgi-bin/ 开头的资源都映射到 /var/www/cgi-bin/ 目录
  - URL为 **http://www.ls-al.me/cgi-bin/test.pl** 的请求，Apache 会试图执行 /var/www/cgi-bin/test.pl 文件（CGI 程序文件必须存在而且可执行）

梁如军（linuxbooks@126.com）

创建CGI脚本并浏览测试

- 创建 /var/www/cgi-bin/test.pl

```
#!/usr/bin/perl
print "Content-type: text/html\n\n";
print "Hello, World. Perl“;
```

- 添加可执行权限并进行浏览测试

```
# chmod +x /var/www/cgi-bin/test.pl
# elinks http://www.ls-al.me/cgi-bin/test.pl
# elinks http://www.olabs.org/cgi-bin/test.pl
# elinks http://www.olabs.net/cgi-bin/test.pl
```


梁如军（linuxbooks@126.com）

为虚拟主机配置ScriptAlias

- 在 VirtualHost 容器中配置 ScriptAlias 指令

```
<VirtualHost *:80>
    ServerName www.olabs.org
    DocumentRoot "/var/www/vhosts/olabs.org/htdocs/"
    ..................
    ScriptAlias /cgi-bin/ "/var/www/vhosts/olabs.org/cgi-bin/"
    <Directory "/var/www/vhosts/olabs.org/cgi-bin">
        AllowOverride None
        Options None
        Require all granted
    </Directory>
</VirtualHost>
```


梁如军（linuxbooks@126.com）

ScriptAlias说明

- ScriptAlias指令除了可以映射目录之外，还可以直接映射CGI程序
- 例如

  ScriptAlias　/cgit　/var/www/cgi-bin/cgit

  - /var/www/cgi-bin/cgit 是一个CGI程序
  - 当访问 **http://xxx.xxx.xxx/cgit** 时将直接执行 /var/www/cgi-bin/cgit 程序

在任意目录中执行 CGI 程序

- 配置方法
  - 用 AddHandler 或 SetHandler 指令激活 cgi-script 处理器
  - 在目录容器的 Options 指令中启用 ExecCGI 选项
- AddHandler / SetHandler 指令

  AddHandler cgi-script .cgi .pl
  - 用于在文件扩展名与特定的处理器 之间建立映射
  - 告诉服务器哪些文件是 CGI 程序文件

梁如军（linuxbooks@126.com）

在任意目录中执行 CGI 程序配置举例

- /etc/httpd/conf.d/git.conf

```
Alias /git /var/www/git
<Directory /var/www/git>
    Options +ExecCGI
    AddHandler cgi-script .cgi
</Directory>
```

  - 使用别名将 http://xxx.xxx.xxx/**git** 的访问映射到磁盘的 /var/www/git 目录
  - 允许执行 /var/www/git 目录下的后缀名为 .cgi 的程序

梁如军（linuxbooks@126.com）

CGI的优缺点

- 优点
  - 安全性好
  - 用C语言写的CGI程序,编译后的运行速度比脚本运行速度要快
- 缺点
  - 需要开独立进程（fork-and-execute 模式）来处理用户请求，密集请求的情况下容易崩溃
  - 维护成本比脚本语言高
  - 通常CGI程序使用Perl编写，其语法相对复杂

梁如军（linuxbooks@126.com）

解决CGI的低效率

- 为了适应密集请求（高负载）型的Web服务器
- 解决CGI的低执行效率的方法
  - **使用Apache的第三方脚本语言模块**
    - 模块当 Apache 运行后就常驻内存
    - 不会像 CGI 那样每次都要花费时间去 fork 一次
  - **使用FastCGI技术**
    - 是一种常驻（Long-Live）型的 CGI
      - 类似于系统守护进程
    - 可以一直执行着为来自服务器的请求提供服务
    - 只要激活后，不会每次都要花费时间去 fork 一次

梁如军（linuxbooks@126.com）

Apache的脚本语言模块

- PHP：**mod_php**
  - http://www.php.net/
- Perl：**mod_perl**
  - http://perl.apache.org/
- Python：**mod_python**
  - http://www.modpython.org/
- Ruby：**mod_passenger**
  - http://www.modrails.com/

梁如军（linuxbooks@126.com）

FastCGI的优点

- 稳定性：FastCGI 是以独立的进程池运行来 CGI
  - 单独一个进程死掉,系统可以很轻易的丢弃,然后重新分配新的进程来运行之
- 安全性：FastCGI 和宿主服务器完全独立
  - 即使 FastCGI 僵死也不会导致服务器宕机
- 扩展性：FastCGI是一个中立的技术标准
  - 可以支持任何语言写的处理程序，如：PHP、Perl、Python、Ruby、Java、C/C＋＋等

梁如军（linuxbooks@126.com）

FastCGI的优点（续）

- 高性能
  - FastCGI 将动态逻辑的处理从 Web 服务器中分离出来
    - 大负荷的 I/O 处理还是留给宿主服务器
    - 宿主服务器可以一心一意作 I/O 处理
    - 大量的图片等静态 I/O 处理完全不需要逻辑程序的参与
  - 可以让 Web 服务器运行多个 FastCGI 应用程序的副本来提高性能
  - FastCGI 可以很有效地利用内存来作缓存来提高性能

梁如军（linuxbooks@126.com）

Apache 与 FastCGI

- Linux下常用的Web服务器均支持FastCGI
  - Apache、Nginx、Lighttpd、Cherokee
- Apache使用 **mod_fcgid** 模块实现
  - 由EPEL仓库的mod_fcgid包提供
- 配置 Apache 允许执行 FastCGI 程序
  - 与允许执行 CGI 程序类似

```
ScriptAlias /fcgi-bin  /var/www/fcgi-bin
```

```
AddHandler fastcgi-script .php .py .pl .fcgi
Options +ExecCGI
```


梁如军（linuxbooks@126.com）

APACHE日志统计分析工具

梁如军（linuxbooks@126.com）

AWStats简介

- **AWStats**（Advanced Web Statistics）是一个免费的功能强大的服务器日志分析工具
- AWStats 的功能
  - 提供访问量，访问次数，访问者IP，访问者国家或地区、页面浏览量，点击数，高峰时段、访客持续时间，数据流量等
  - 提供精确到每月、每日、每小时的统计数据
  - 提供访客操作系统、浏览器版本的统计信息
  - 提供Robots/Spiders 机械访问的统计、无效连接等
  - 搜索引擎、关键字、以及对不同文件类型的统计信息

梁如军（linuxbooks@126.com）

CentOS下的AWStats

- 安装AWStats

  # yum --enablerepo=epel install awstats

- 重要文件说明
  - 配置文件模板
    - **/etc/awstats/awstats.model.conf**
  - 每个虚拟主机的配置文件（*为虚拟主机名）
    - **/etc/awstats/awstats.*.conf**
  - 每小时生成一次AWStats 数据库的cron脚本
    - **/etc/cron.hourly/awstats**
  - 用于执行awstats.pl的Apache的CGI配置文件
    - **/etc/httpd/conf.d/awstats.conf**

梁如军（linuxbooks@126.com）

AWStats的配置文件

- AWStats在生成其统计数据库时需要其配置文件
  - AWStats 为不同的站点使用不同的配置文件
  - 配置文件的命名规则awstats.SITENAME.conf
  - 例如：站点名为 mysite.net，则配置文件名为 awstats.mysite.net.conf
- 常用配置语句
  - **SiteDomain=**
  - **HostAliases=**
  - **LogFile=**

梁如军（linuxbooks@126.com）

更新AWStats的统计数据库

- 生成指定站点的日志统计数据库

  # /usr/share/awstats/wwwroot/cgi-bin/awstats.pl -config=SITENAME

- 生成所有虚拟主机的统计数据库
  - 对指定配置文件目录下的每个配置文件生成统计数据库
  - **/usr/share/awstats/tools/awstats_updateall.pl**
  - AWStats 的cron脚本使用了 awstats_updateall.pl

AWStats 的 Apache 配置文件

- **/etc/httpd/conf.d/awstats.conf**
  - 此配置文件是被主配置文件包含的，是全局配置，所有的虚拟主机都将继承这个配置

```
.....
ScriptAlias /awstats/ "/usr/share/awstats/wwwroot/cgi-bin/"
<Directory "/usr/share/awstats/wwwroot">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from 127.0.0.1
</Directory>
.....
```


梁如军（linuxbooks@126.com）

AWStats配置举例

- 为虚拟主机配置AWStats
  - 在Apache中为AWStats的访问配置主机访问控制、认证授权
  - 生成虚拟主机的AWStats配置文件
  - 更新指定配置文件的AWStats的统计数据库
  - 访问CGI脚本获得AWStats的统计输出

参见教材的操作步骤

梁如军（linuxbooks@126.com）

LAMP的环境配置及应用

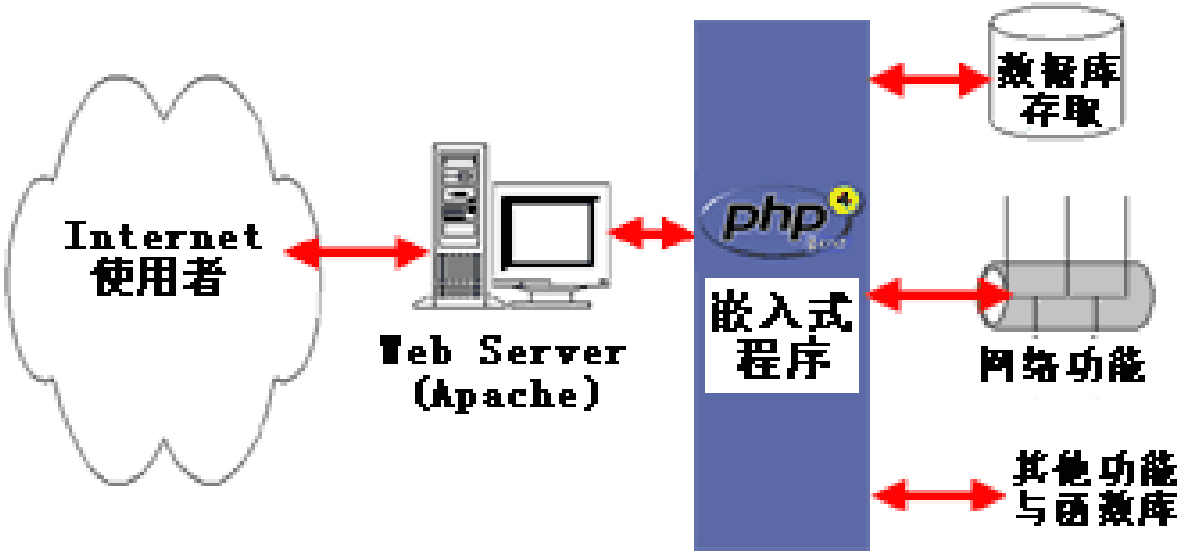
梁如军（linuxbooks@126.com）

LAMP简介

- LAMP是首字母缩略语（Acronym）
  - L：Linux 操作系统
  - A：Apache Web 服务器
  - M：MySQL 数据库
  - P：PHP 、Perl 、Python 或 Ruby 脚本语言
- LAMP的特点
  - 开放灵活、开发迅速、部署方便
  - 高可配置、安全可靠、成本低廉等
- 与Java平台和.NET平台鼎足三分
  - 尤其受中小企业的欢迎

狭义LAMP

- Linux+Apache+MySQL+PHP
- 是最常用的开源平台组合



梁如军（linuxbooks@126.com）

基于脚本语言的知名Web框架

CentOS
The Community ENTerprise Operating System

语言	Web框架	主页
PHP	symfony	http://www.symfony-project.org/
	Zend	http://framework.zend.com/
	CodeIgniter	http://codeigniter.com/
	CakePHP	http://cakephp.org/
Python	Django	http://www.djangoproject.com/
	Pylons	http://pylonshq.com/
	TurboGears	http://www.turbogears.org/
	Grok	http://grok.zope.org/
Ruby	Ruby on Rails	http://www.rubyonrails.org/
Perl	Catalyst	http://www.catalystframework.org/

梁如军（linuxbooks@126.com）

安装配置LAMP环境

- 安装配置PHP
- 安装和测试phpMyAdmin

梁如军（linuxbooks@126.com）

安装PHP

- 安装PHP及其相关的软件包

```
# yum install php php-cli php-pear
# yum install php-pdo php-mysql
# yum install php-mcrypt php-mbstring
# yum install php-xml php-pecl-yaml
# yum install php-gd php-pecl-imagick
# yum install php-pecl-apc
# yum install php-pecl-memcached php-pecl-redis
```


梁如军（linuxbooks@126.com）

查看PHP的配置

- 查看PHP的配置
  - 查看PHP已加载的模块

 # php -m

  - 显示phpinfo()的信息输出

 # php -I

- 查看PHP的配置文件

```
# less /etc/php.ini
# ls /etc/php.d
apc.ini       json.ini      mysql.ini      pdo_sqlite.ini  xmlwriter.ini
curl.ini      mbstring.ini  odbc.ini       phar.ini        xsl.ini
……
```

配置PHP的主配置文件

- # vi /etc/php.ini

```
# 对于生产平台，应将display_errors设置为 Off
display_errors = Off
# 将log_errors设置为 On
log_errors = On
# 使用 zlib 库压缩输出并设置压缩级别
zlib.output_compression = On
zlib.output_compression_level = 1
# 不暴露PHP被安装在服务器上的事实
expose_php = Off

# 限制一个PHP脚本可能消耗的最大内存量
# 这有助于防止写得不好的脚本消耗服务器上的可用内存。
memory_limit = 256M
```


梁如军（linuxbooks@126.com）

配置PHP的主配置文件（续）

- # vi /etc/php.ini

```
# 为POST方法指定可接受的最大尺寸
post_max_size = 512M

# 设置可上传文件的最大尺寸
upload_max_filesize = 20M

# 不能使用URL（如： http:// 或 ftp://) 直接打开文件
allow_url_fopen = Off

[Date]
# 为日期函数定义默认时区
date.timezone = Asia/Shanghai
```


梁如军（linuxbooks@126.com）

配置PHP的APC模块

CentOS
The Community ENTerprise Operating System

vi /etc/php.d/apc.ini

```
# 是否启用 APC opcode cache，若系统中无其他 OPcache可以设置为1
apc.enable_opcode_cache=0
# 启用APC （这是默认配置，1是启用）
apc.enabled = 1
# 每个共享内存块的大小，（可以使用单位后缀M/G）
apc.shm_size = 64M
# 缓存条目在缓冲区中允许逗留的秒数。0 表示永不超时。建议值为
7200~36000
apc.ttl =  7200
# 缓存条目在垃圾回收表中能够存在的秒数
apc.gc_ttl = 3600
# 是否启用脚本更新检查。改变这个指令值要非常小心。
#   默认值 On 表示APC在每次请求脚本时都检查脚本是否被更新，
#   若检查到更新则自动重新编译和缓存编译后的内容。
apc.stat=1
```


梁如军（linuxbooks@126.com）

PHP的Zend Guard支持

- 有些PHP应用程序需要Zend组件（Zend Optimizer/Zend Guard Loader）的支持
  - Zend Guard为独立的软件供应商和IT管理者提供保护PHP应用程序源代码的能力。
  - Zend Guard Loader 是一个免费的应用程序，运行使用Zend Guard编码的文件并提高PHP应用程序的整体性能。
- 对于PHP5.4版，只能安装Zend Guard Loader。

安装配置Zend Guard（1）

- 查看当前安装的 PHP版本

  # php -v

  PHP 5.4.29 (cli) (built: May 28 2014 15:02:43)

  Copyright (c) 1997-2014 The PHP Group

  Zend Engine v2.4.0, Copyright (c) 1998-2014 Zend Technologies

- 下载 Zend Guard Loader
  - http://www.zend.com/en/products/guard/downloads
  - 下载符合$(arch)的用于Linux的PHP 5.4的最新版本
  - ZendGuardLoader-70429-PHP-5.4-linux-glibc23-x86_64.tar.gz

梁如军（linuxbooks@126.com）

安装配置Zend Guard（2）

- 解压并将ZendGuardLoader.so复制到PHP模块目录 /usr/lib64/php/modules/
- 配置ZendGuardLoader的模块配置文件

  # vi /etc/php.d/ZendGuardLoader.ini

```
zend_extension=/usr/lib64/php/modules/ZendGuardLoader.so
zend_loader.enable=1
```

- 检测安装

  # php -v

  .....

  with Zend Guard Loader v3.3, Copyright (c) 1998-2013, by Zend Technologies

梁如军（linuxbooks@126.com）

配置Apache的php模块

- /etc/httpd/conf.d/php.conf

```
<IfModule prefork.c>
  LoadModule php5_module modules/libphp5.so
</IfModule>
<IfModule !prefork.c>
  LoadModule php5_module modules/libphp5-zts.so
</IfModule>

<FilesMatch \.php$>
    SetHandler application/x-httpd-php
</FilesMatch>

AddType text/html .php

DirectoryIndex index.php

php_value session.save_handler "files"
php_value session.save_path    "/var/lib/php/session"
```


梁如军（linuxbooks@126.com）

测试PHP和MySQL

- 重新启动Apache

  # service httpd restart

- 在/var/www/html目录下编写一个测试脚本

  # echo '<?php phpinfo()?>' >
  /var/www/html/info.php

- 使用浏览器进行测试

  # elinks http://olabs.lan/info.php

梁如军（linuxbooks@126.com）

phpMyAdmin

- phpMyAdmin 是一个用PHP编写的基于Web的Mysql管理工具
- phpMyAdmin 界面友好，操作简单
- phpMyAdmin的主页
  - http://phpmyadmin.sourceforge.net/
- phpMyAdmin的安装和配置
  - # yum install phpMyAdmin

参考教材操作步骤

梁如军（linuxbooks@126.com）

常用的LAMP应用

- Portal CMS
- LMS/LCMS
- Wiki
- BLOG
- Forum
- Groupware
- WebMail
- BugTrackers
- phpDBadmin
- Web Hosting Control Panel

梁如军（linuxbooks@126.com）

常用的LAMP应用软件

- Moodle
- Wordpress
- Drupal
- phpBB
- ……

梁如军（linuxbooks@126.com）

Moodle 简介

- Moodle为远程教育提供了一种优秀的开源解决方案
  - 是使用PHP编写的面向对象的模块化动态教学环境
  - 是由澳大利亚教师Martin Dougiamas基于建构主义教育理论而开发的免费、开源的课程管理系统(Course Management System，CMS)
  - 具有内容管理、学习管理和课程管理三大功能
    - 包含论坛、测验、资源、投票、问卷、作业、聊天和博客等模块
    - 具有大量功能丰富的第三方插件
  - 是目前全球范围内应用最广泛的在线教学平台之一

Moodle配置举例

- 下载最新版Moodle
- 配置MySQL服务的InnoDB存储支持
- 配置Apache
- 运行Moodle的安装配置脚本
  - 或使用浏览器实现交互式安装
- 安排Moodle的cron任务

参考教材操作步骤

梁如军（linuxbooks@126.com）

JDK和TOMCAT

梁如军（linuxbooks@126.com）

CentOS 7下的Java运行环境

- CentOS支持的JDK
  - OpenJDK（http://openjdk.java.net/）
  - Oracle的JavaSE（JDK）
- CentOS 7 提供了三种版本的OpenJDK

```
# yum search openjdk|grep 'OpenJDK Runtime Environment‘

java-1.6.0-openjdk.x86_64 : OpenJDK Runtime Environment
java-1.7.0-openjdk.x86_64 : OpenJDK Runtime Environment
java-1.8.0-openjdk.x86_64 : OpenJDK Runtime Environment
```


梁如军（linuxbooks@126.com）

安装Oracle的JavaSE（JDK）

http://www.oracle.com/technetwork/java/javase/downloads/index.html

- 下载

```
# aria2c --header="Cookie: oraclelicense=accept-securebackup-cookie" \
  http://download.oracle.com/otn-pub/java/jdk/8u65-b17/jdk-8u65-linux-x64.rpm
```

- 安装并配置

```
# rpm -ivh jdk-8u65-linux-x64.rpm

# alternatives --install /usr/bin/java java          /usr/java/jdk1.8.0_65/bin/java          2000000
# alternatives --install /usr/bin/javac javac       /usr/java/jdk1.8.0_65/bin/javac         2000000
# alternatives --install /usr/bin/javaws javaws     /usr/java/jdk1.8.0_65/jre/bin/javaws    2000000
# alternatives --install /usr/bin/jar jar           /usr/java/jdk1.8.0_65/bin/jar           2000000
```


梁如军（linuxbooks@126.com）

配置Oracle的JavaSE（JDK）

- 配置方法2

```
# echo '
export JAVA_HOME=/usr/java/default
export PATH=$JAVA_HOME/bin:$JAVA_HOME/jre/bin:$PATH
export CLASSPATH=.:$JAVA_HOME/lib:$JAVA_HOME/jre/lib:$CLASSPATH
' > /etc/profile.d/java.sh
# chmod +x /etc/profile.d/java.sh
```

- 测试

  # . /etc/profile.d/java.sh

  # java -version

梁如军（linuxbooks@126.com）

Tomcat简介

- 是一个免费的开源的Jsp和Servlet的运行平台
  - 是Apache基金会的Jakarta项目中的一个核心项目
  - 由Apache、Sun/Oracle和其它一些公司及个人共同开发而成
  - Tomcat 技术先进、性能稳定，而且免费
- Tomcat同时也具有传统的Web服务器的功能
  - 与Apache/Nginx相比，它的处理静态Html的能力不如Apache/Nginx
  - 通常可以将Tomcat和Apache/Nginx集成到一起，让Apache/Nginx处理静态Html，而让Tomcat处理Jsp和Servlet。

梁如军（linuxbooks@126.com）

安装和启动Tomcat

- 安装Tomcat7（EPEL仓库）

```
# yum install tomcat log4j tomcat-native \
    tomcat-jsp-2.2-api tomcat-servlet-3.0-api
```

管理Tomcat服务

- 对于默认的tomcat实例
  - # systemctl {start|stop|status|restart} tomcat
  - # systemctl {enable|disable} tomcat
- 对于非默认的tomcat实例
  - # systemctl {start|stop|status|restart} tomcat@NAME
  - # systemctl {enable|disable} tomcat@NAME

梁如军（linuxbooks@126.com）

配置Tomcat
——配置 Tomcat的工作环境

- # vi /etc/tomcat/tomcat.conf

```
// 修改环境变量JAVA_HOME
JAVA_HOME="/usr/java/default"

// 修改环境变量 JAVA_OPTS
JAVA_OPTS="-Xms2048m -Xmx2048m
    -XX:PermSize=512m -XX:MaxPermSize=512m”
```


梁如军（linuxbooks@126.com）

配置Tomcat
——配置 Tomcat的服务

- # vi /etc/tomcat/server.xml

```
<Connector port="8080" protocol="HTTP/1.1"
        connectionTimeout="30000"  // 指定网络连接超时，单位：毫秒
        redirectPort="8443"    // 指定Tomcat服务器的重定向端口号
        enableLookups="false"  // 为了提高处理能力，禁止域名反向查询
        acceptCount="200"  // 指定当所有可以使用的处理请求的线程数都被使用时，
                           // 可以放到处理队列中的请求数，超过这个数的请求将不予处理
        maxThreads="400"           // 指定Tomcat可创建的最大的线程数。
        keepAliveTimeout="600000" // 指定keepAlive的时效时间，单位：毫秒
        URIEncoding="UTF-8"/>      // 指定对URI使用UTF-8编码
```


梁如军（linuxbooks@126.com）

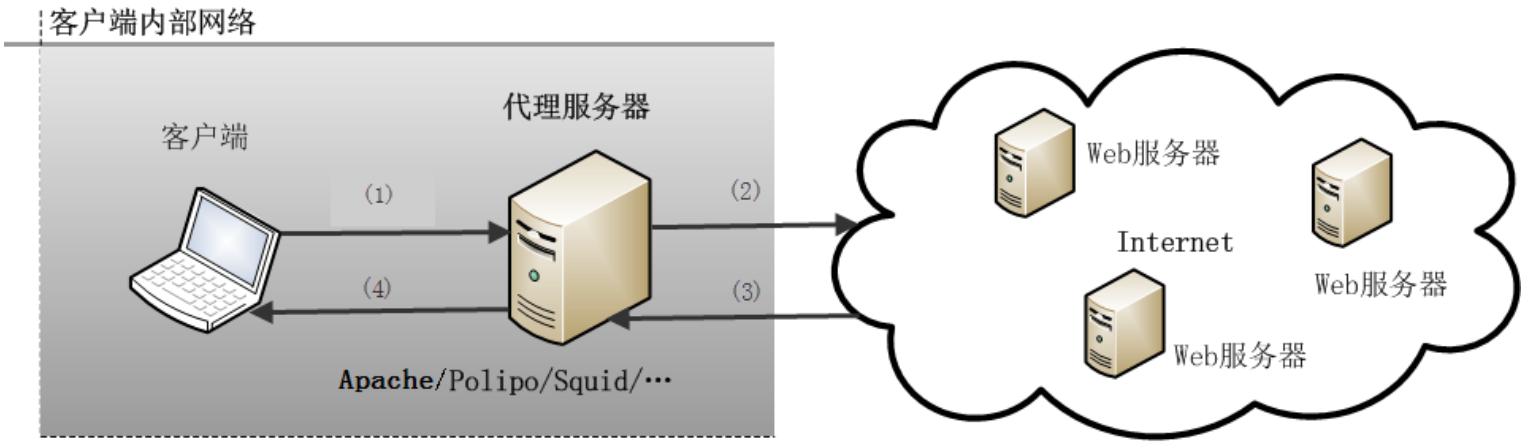
APACHE与TOMCAT

梁如军（linuxbooks@126.com）

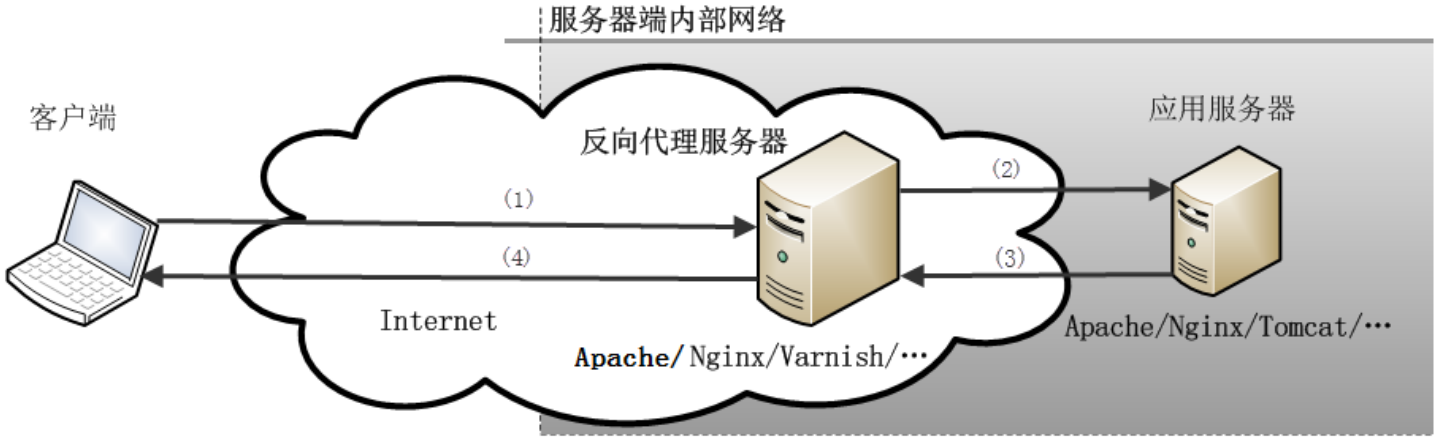
反向代理简介

- 反向代理（Reverse Proxy）服务器是指用代理服务器来接受来自Internet上的连接请求，然后将请求转发给内部网络上的应用服务器（Web/Mail服务器等）；并将从应用服务器上得到的响应结果返回给请求连接的客户端。
- 反向代理服务器对客户来说表现为一个应用服务器
- 使用反向代理服务，必须将被代理的后端应用服务器的DNS解析指向反向代理服务器

代理服务器与反向代理服务器



(a) 代理服务器



(b) 反向代理服务器

梁如军（linuxbooks@126.com）

反向代理的处理步骤

- 步骤1：Web客户端的Web请求发送给反向代理服务器。
- 步骤2：反向代理服务器接到客户端的请求后将其转发给后端的Web应用服务器。
  - 大多数反向代理服务器具有缓存功能。
  - 若反向代理服务器启用了缓存功能，则反向代理服务器接到客户端的请求后首先在自己的缓存中查找是否缓存过与之对应的响应，若找到相应的缓存对象且未失效则直接将其返回给客户端；否则将请求转发给后端的Web应用服务器。

梁如军（linuxbooks@126.com）

反向代理的处理步骤（续）

- 步骤3：后端 Web应用服务器接到反向代理服务器的Web请求，Web服务器经过处理将响应发送给反向代理服务器。
- 步骤4：反向代理服务器接到后端的Web应用服务器发来的响应后将其发送给客户端。若反向代理服务器启用了缓存功能，同时还会在其自己的缓存中保留一份响应结果，以加快相同请求的后续访问。

梁如军（linuxbooks@126.com）

反向代理的优缺点

- 优点
  - 反向代理作为后台应用服务器的替身，大大地提高了安全性，保护了敏感信息不外泄
- 缺点
  - 反向代理作为后台应用服务器的替身，一旦其宕机便无法对外提供服务。
  - 为了避免反向代理成为单点故障，通常的做法是架设两台或两台以上的反向代理服务器，使用高可用技术（如：Keepalive、Heartbeat等）实现故障切换。

Apache的代理支持

- Apache通过mod_proxy模块提供代理支持，它既支持正向代理也支持反向代理
- 通过模块实现各种协议/方案的代理功能
  - **mod_proxy_http**：支持HTTP协议
  - **mod_proxy_connect**：支持HTTP 协议的 CONNECT 方法，用于 SSL 请求
  - **mod_proxy_ftp**：支持FTP协议
  - **mod_proxy_ajp**：支持 AJP13（Apache JServe Protocol version 1.3）协议
  - **mod_proxy_fcgi**：支持 FastCGI
  - **mod_proxy_scgi**：支持SCGI
  - **mod_proxy_wstunnel**：支持WebSocket协议

梁如军（linuxbooks@126.com）

Apache与代理相关的配置指令

- ProxyRequests
- ProxyPreserveHost
- ProxyPass
- ProxyPassMatch
- ProxyPassReverse
- <Proxy> </Proxy>
- <ProxyMatch> </ProxyMatch>

梁如军（linuxbooks@126.com）

使用Apache反向代理HTTP

```
<VirtualHost _default_:80>
    DocumentRoot /var/www/html
    ServerName www.olabs.lan

    ProxyRequests Off                    // 关闭正向代理
    ProxyPreserveHost On

    <Proxy http://www.olabs.lan>         // 允许对http://www.olabs.lan
的访问使用代理
        AllowOverride None
        Require all granted
    </Proxy>
    ProxyPass /8848/ http://www.olabs.lan:8848/
    ProxyPassReverse /8848/ http://www.olabs.lan:8848/
</VirtualHost>
```


梁如军（linuxbooks@126.com）

使用Apache反向代理Tomcat

```
<VirtualHost _default_:80>
    DocumentRoot /var/www/html
    ServerName www.olabs.lan
    ……

    Alias /docs /usr/share/tomcat/webapps/docs
    <Directory /usr/share/tomcat/webapps/docs>
        Options Indexes FollowSymlinks
        Require all granted
    </Directory>

    ProxyRequests Off                       // 关闭正向代理
    ProxyPass /docs !                       // 不对 /docs 的URI访问进行反向代理

    ProxyPass / ajp://localhost:8009/       // 使用 AJP协议反向代理 Tomcat的默认实例
    ProxyPassReverse / ajp://localhost:8009/        // 避免Tomcat内的地址重定向绕过代理服务器
</VirtualHost>
```


梁如军（linuxbooks@126.com）

CentOS
The Community ENTerprise Operating System

APACHE与负载均衡

梁如军（linuxbooks@126.com）

负载均衡简介

- 使用一组服务器设备同时承担一项业务
  - 在这一组设备前端添加一个或多个设备负责分发客户请求
  - 每次请求根据特定的负载均衡算法分发到一组服务器设备之一
  - 这种设备称为负载均衡设备或负载均衡器
  - 负载均衡设备不是基础网络设备，而是一种性能优化设备

梁如军（linuxbooks@126.com）

负载均衡的分类（1）

- DNS负载均衡
- NAT负载均衡
- 反向代理负载均衡
  - 将来自Internet上的连接请求以反向代理的方式动态地转发给内部网络上的多台服务器进行处理，从而达到负载均衡的目的。

负载均衡的分类（2）

- 第二层负载均衡
- 第四层负载均衡
- 第七层负载均衡
  - 也称应用层负载均衡。
  - 提供了一种对访问流量的高层控制方式，如根据报头内的信息来执行负载均衡任务等。

梁如军（linuxbooks@126.com）

负载均衡的分类（3）

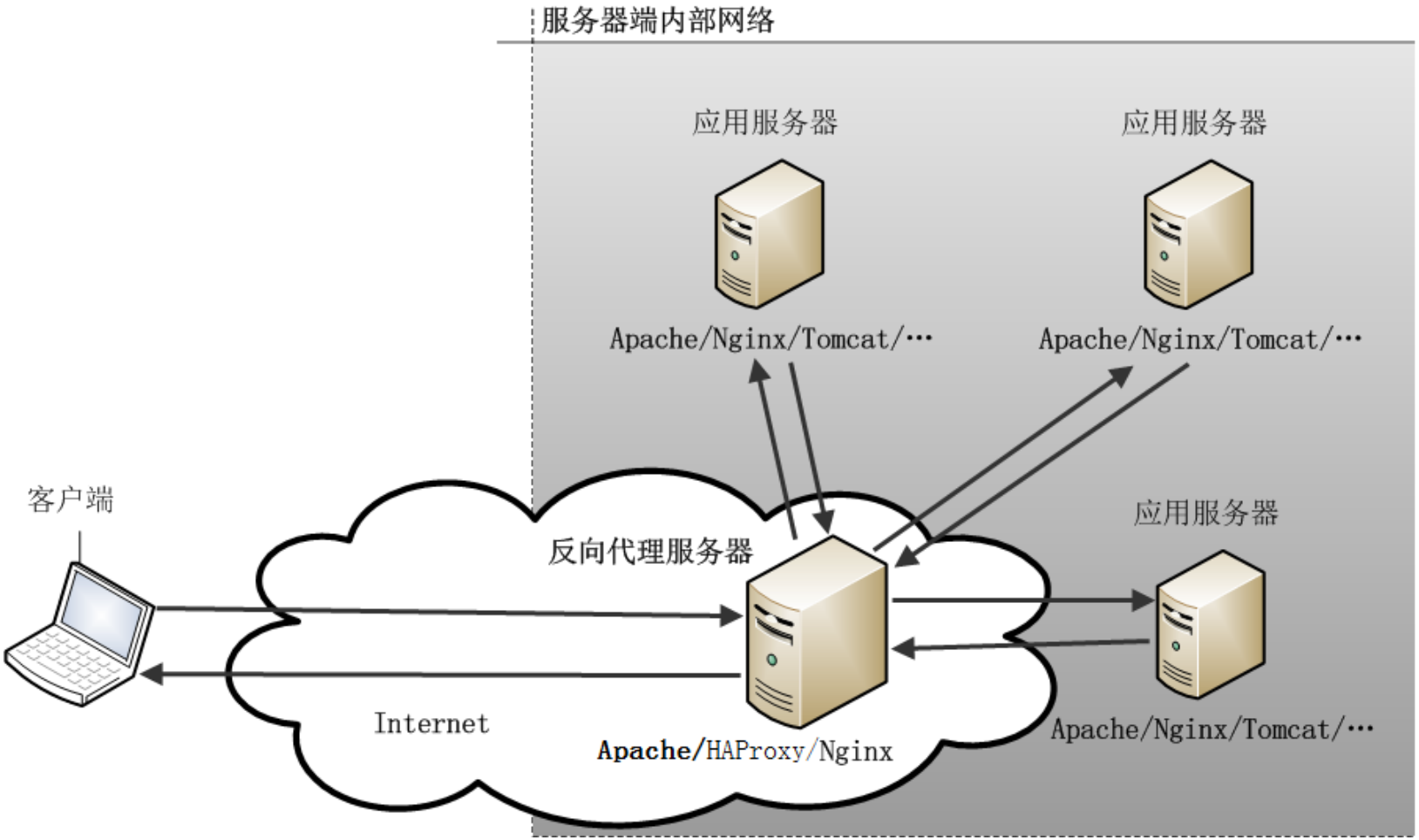
- 软件负载均衡
  - 在一台或多台服务器相应的操作系统上安装一个或多个附加软件来实现负载均衡
  - 优点是基于特定环境，配置简单，使用灵活，成本低廉，可以满足一般的负载均衡需求。
  - 如：LVS（4层实现）、Nginx（7层实现）
- 硬件负载均衡
  - 使用专门的负载均衡设备（称之为负载均衡器）
  - 硬件负载均衡在功能、性能上优于软件方式，不过成本昂贵。
  - 如：F5 BIG-IP、Radware 的 AppDirector

梁如军（linuxbooks@126.com）

基于Apache的负载均衡

- Apache实现了基于应用层的反向代理负载均衡
- Apache提供了mod_proxy_balancer模块用于配置反向代理的上游服务器组
  - 实现了HTTP、FTP 和 AJP13三种协议的负载均衡
  - 支持三种负载均衡调度算法（可以通过 lbmethod 参数指定）：
    - byrequests：加权请求计数（Weighted Request Counting）
    - bytraffic：加权流量计数（Weighted Traffic Counting）
    - bybusyness：等待请求计数（Pending Request Counting）
  - 连接黏着（Stickyness）：当一个用户请求按照负载均衡算法发送到一台选定的后端服务器后，相同用户的后续连接请求也同样发送到相同的后端服务器

Apache做反向代理负载均衡器



梁如军（linuxbooks@126.com）

使用Apache 配置反向代理负载均衡

- 使用 Proxy容器声明一个负载均衡器
  - 定义一个后台服务器池（HTTP协议）

```
<Proxy "balancer://mycluster">
    # 在Proxy容器内分别定义每个可用的后台服务器
    BalancerMember  "http://192.168.1.50:80"
    BalancerMember  "http://192.168.1.51:80"
</Proxy>
```

- 在 ProxyPass和 ProxyPassReverse指令中引用之

```
ProxyPass "/test"  "balancer://mycluster"
ProxyPassReverse  "/test"  "balancer://mycluster"
```


梁如军（linuxbooks@126.com）

使用Apache 配置反向代理负载均衡（续）

- 使用 Proxy容器声明一个负载均衡器
  - 定义一个后台服务器池（ AJP协议）

```
<Proxy "balancer://BackendServers" >
 # 在Proxy容器内分别定义每个可用的后台服务器及其权值
   BalancerMember "ajp://10.0.0.1:8009"
   BalancerMember "ajp://10.0.0.2:8009" loadfactor=10
   BalancerMember "ajp://10.0.0.3:8009" loadfactor=5 </Proxy>
```

- 在 ProxyPass和 ProxyPassReverse指令中引用之

```
ProxyPass  /  "balancer://BackendServers/"
ProxyPassReverse  /  "balancer://BackendServers/"
```


梁如军（linuxbooks@126.com）

本章思考题

- Linux环境下常用的脚本语言有哪些？各自有何特点？
- 常见的动态网站技术有哪些？与CGI相比FastCGI有哪些特点和优势？
- 什么是AWStats？AWStats提供了哪些功能？
- 什么是LAMP？LAMP的常见应用有哪些？
- 什么是Portal CMS、LMS/LCMS、Wiki、BLOG、Forum、Groupware、WebMail、BugTrackers、phpDBadmin、Web Hosting Control Panel？
- 什么是反向代理？Apache的反向代理能代理哪些后端服务？
- 什么是负载均衡？如何分类？Apache使用哪种负载均衡技术和算法？

梁如军（linuxbooks@126.com）

本章实验

- 学会配置AWStats访问Apache的访问日志统计。
- 学会配置基于Apache动态语言模块的LAMP环境。
- 学会安装和配置常见的LAMP应用软件
  - 如：Drupal、Joomla、MediaWiki、Wordpress、phpBB和Moodle等
- 学会安装和配置常见的国产LAMP应用软件
  - 如：康盛公司（http://www.comsenz.com/）的SupeSite、Discuz!X 等产品
- 学会配置Tomcat的默认实例和第二实例。
- 学会使用Apache反向代理Tomcat。

进一步学习

- 安装extras仓库提供的RPM包 centos-release-scl，使用sclo仓库安装 PHP 5.5。
- 学习配置基于Python+WSGI+Apache的Django应用（如OSQA）。
- 学习基于Ruby+Apache的Redmine或Gitlab的安装、配置和使用。

- 学习基于PHP的 ISPConfig（http://www.ispconfig.org/）的安装、配置和使用。
- 学习使用 ownCloud（https://owncloud.org/）搭建自己的私有云存储平台。

- 下载Tomcat 8的二进制包并安装配置Tomcat 8服务。