Nginx的特点:高性能的http和反向代理服务器,IMAP/POP3/SMTP代理服务器与Apache的不同之处

查看服务相关进程及端口信息

进行nginx升级,不需要停止服务,只要在编译之后将旧文件进行备份替换之后再make并make upgrade

Nginx的报错日志在 /var/log/nginx/error.log

在全局选项中:有进程所有者 user 启动进程数量 worker\_processes 日志文件 error\_log PID文件 pid 单个进程最大并发量 worker\_connections

配置容器 在server{内 定义虚拟主机和发布目录

**用户认证--即登录的时候需要输入用户和密码**location /{在server{内

在root html; 与index index.html;后面添加 auth\_basic “auth-domain”;

和auth\_basic\_user\_file /usr/local/nginx/pass;**完成配置后还需要安装包httpd-tools**

**虚拟机的三种类型:基于域名,基于ip,基于端口.都在server{处的listen和server\_name进行修改;ssl加密网站,启用下面注释掉的#443内容**

**auth\_basic "Input Password:"; (内容随意) //认证提示符**

**auth\_basic\_user\_file "/usr/local/nginx/pass"; //认证密码文件**

**装包httpd-tools**

**htpasswd -c /usr/local/nginx/pass tom //创建密码文件**

**htpasswd /usr/local/nginx/pass jerry //追加用户，不使用-c选项**

**root html; //指定网站根路径(在/usr/local/nginx下的文件夹)**

**访问基于域名的虚拟主机 firefox <http://www.a.com> [需要在访问的客户机里设置/etc/hosts进行域名解析]192.168.4.5 www.a.com**

**当设置基于端口的虚拟主机时 server\_name 处的域名不变,只要更改listen处的端口号**

**当设置基于ip的虚拟主机时 只改listen处的ip加上端口**

**地址重写的好处:缩短URL,隐藏实际路径提高安全性;易于用户记忆和键入;易于被搜索引擎收录**

**SVN版本控制系统(自由/开源)**

**/var/svn/project 下面各种配置文件**

**/usr/lib/systemd/system 主机重要的配置文件**

**/tmp/code 自定义存放下载版本库的地方**

**在装完subversion后,创建版本库,需要自建一个/var/svn该svn文件夹之前不存在,之后用svnadmin create /var/svn/project 进行 版本仓库的建立.[svn所有的配置文件都在该目录下].之后本地导入初始化数据操作,要在目标目录下svn import . file:///var/svn/project/ -m "备注"**

**--更改配置文件,创建账户与密码 /conf下的svnserve.conf**

**anon-access = none //19行，匿名无任何权限**

**auth-access = write //20行，有效账户可写**

**password-db = passwd //27行，密码文件**

**authz-db = authz //34行，ACL访问控制列表文件**

**设置用户及密码 /conf下的/passwd**

**harry = 123456 //用户名和密码**

**设置用户的权限 /conf下的/authz**

**[/] //定义ACL访问控制**

**harry = rw //用户对项目根路径可读可写**

**#注意!所有更改的命令行都要定格写**

**启动svn服务 svnserve -d -r /var/svn/project/**

**备注：启动服务也可以使用svnserve -d启动，但客户端访问时需要指定绝对路径（svn://服务器IP/var/svn/project）。**

**将服务器上的代码下载到本地**

**svn --username harry --password 123456 co svn://192.168.2.100/ code**

**会在当前目录生成名为code的文件夹,而下载的东西都在这下面**

**#之后可以对下载的东西进行修改,之后用svn ci -m "此次修改的注释"将本地修改的数据同步到服务器.**

**#查看版本仓库基本信息:svn info svn://201.1.2.100 (此步骤也需要用户账户登录之后才能查看)**

**#查看版本仓库的日志,可以查看不同用户对版本仓库的更改:svn log svn://201.1.2.100**

**#可以在本地下载的仓库文件夹(code)新建一个文件,之后用 svn add test.sh之后svn ci -m "备注"可以将本地新建的文件更新到服务器的版本仓库--可以看见更改记录和更改备注,但是看不见**

**#svn mkdir 新文件夹 #svn rm 删除的东西 之后用svn ci -m “备注”**

**可以进行新建和删除**

**#svn diff 可以对比查看现在版本和上一个版本的不同 之后加特定文件可以进行精确查询**

**#svn cat svn://201.1.2.100/haha.txt 可以查看服务器版本仓库的特定内容**

**#对某个文件的内容进行操作(包括添加内容,删除)之后(在提交同步之前)可以svn revert 文件名 进行还原**

**#删除本地下载版本库内文件 用 svn update 进行更新,可以还原现在版本的文件(再未提交之前) update是根据服务器版本库当前版本文件状态进行还原,故在你操作过程中版本进行更新了,update无法还原到当初下载时候的状态,故可以考虑版本还原**

**##!根据版本号进行还原文件 svn merge -r7:2 (版本号) 文件号 只要未进行提交更新都不会影响服务器版本库而只会影响本地下载的版本库**

**#不同机子更新不同文件时不会发生冲突,但改同一文件的时候需要进行人工协商修改结果,不然无法提交更新**

**https://github.com/**