**6\_Operation01Nginx安装升级\服务器\虚拟主机HTTP加密网站**

**一 搭建Nginx服务器**

**1.1 问题**

在IP地址为192.168.4.5的主机上安装部署Nginx服务，并可以将Nginx服务器，要求编译时启用如下功能：

支持SSL加密功能

设置Nginx账户及组名称均为nginx

Nginx服务器升级到更高版本。

然后客户端访问页面验证Nginx Web服务器：

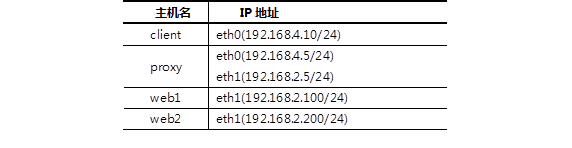
使用火狐浏览器访问

使用curl访问

**1.2 方案**

提前准备运维课程所需的所有虚拟机，为后续所有实验做准备，克隆4台RHEL7虚拟机，实验环境所需要的主机及对应的IP设置列表如表-1所示，正确配置IP地址、主机名称，并且为每台主机配置YUM源。不需要配置网关与DNS。

表－1 主机列表



第一天课程需要使用2台RHEL7虚拟机，其中一台作为Nginx服务器（192.168.4.5）、另外一台作为测试用的Linux客户机（192.168.4.10），如图-1所示。

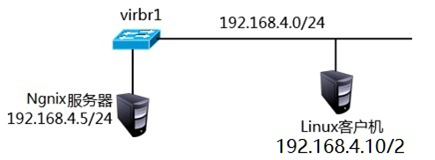


图-1

安装nginx-1.10.3版本时，需要使用如下参数：

--with-http\_ssl\_module：提供SSL加密功能

--user：指定账户

--group：指定组

--profix: 指定安装路径

**1.3 构建Nginx服务器**

1.3.1 使用源码包安装nginx软件包

[root@proxy ~]# yum -y install gcc pcre-devel openssl-devel

#安装依赖包

[root@proxy ~]# useradd -s /sbin/nologin nginx

#创建用户nginx,用户启动nginx服务

#用户root执行命令启动nginx时,自动将用户root降级为nginx

[root@proxy ~]# tar -xf nginx-1.10.3.tar.gz

[root@proxy ~]# cd nginx-1.10.3

[root@proxy nginx-1.10.3]# ./configure \

> --prefix=**/usr/local/nginx** \  #指定安装路径(此路径也是默认路径)

> --user=nginx \          #指定用户

> --group=nginx \            #指定组

> --with-http\_ssl\_module        #开启SSL加密功能

#使用2级提示符在1行命令中指定各项参数

#**/usr/local/nginx/sbin/nginx -V** #可查看上述配置的设置情况

[root@proxy nginx-1.10.3]# make && make install #编译并安装

1.3.2 nginx命令的用法

[root@proxy ~]# **/usr/local/nginx/sbin/nginx** #启动服务

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s stop #闭服务

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s reload

#不关闭服务并重新加载配置文件

[root@proxy ~]# /usr/local/nginx/sbin/nginx -V #查看软件信息

[root@proxy ~]# ln -s /usr/local/nginx/sbin/nginx /sbin/

#创建软连接,方便后期使用

netstat命令可以查看系统中启动的端口信息，该命令常用选项如下：

**-a** 显示所有端口的信息

**-n** 以数字格式显示端口号

**-t** 显示TCP连接的端口

**-u** 显示UDP连接的端口

**-l** 显示服务正在监听的端口信息，如httpd启动后，会一直监听80端口

**-p** 显示监听端口的服务名称是什么（也就是程序名称）

nginx服务默认通过TCP 80端口监听客户端请求：

[root@proxy ~]# netstat -anptu | grep nginx

tcp 0 00.0.0.0:80 0.0.0.0:\* LISTEN 10441/nginx

1.3.3 设置防火墙与SELinux（非必须的操作，如果有则关闭）

[root@proxy ~]# systemctl stop firewalld

[root@proxy ~]# setenforce 0

1.3.4 测试首页文件

Nginx Web服务默认首页文档存储目录为/usr/local/nginx/html/，在此目录下默认有一个名为index.html的文件，使用客户端访问测试页面：

[root@client ~]# firefox 192.168.4.5

[root@client ~]# curl http://192.168.4.5

#curl,基于命令行的浏览器

**1.4 升级Nginx服务器**

1.4.1 编译新版本nginx软件

[root@proxy ~]# tar -zxvf nginx-1.12.2.tar.gz

[root@proxy ~]# cd nginx-1.12.2

[root@proxy nginx-1.12.2]# ./configure \

> --prefix=/usr/local/nginx \

> --user=nginx \

> --group=nginx \

> --with-http\_ssl\_module #完成后多一个objs目录

[root@proxy nginx-1.12.2]# make

#make编译,将C语言代码转换为二进制代码

1.4.2 备份老的nginx主程序，并使用编译好的新版本nginx替换老版本

[root@proxy nginx-1.12.2]# mv /usr/local/nginx/sbin/nginx \

>/usr/local/nginx/sbin/nginxold

[root@proxy nginx-1.12.2]# cp objs/nginx /usr/local/nginx/s

bin/ #拷贝新版本

[root@proxy nginx-1.12.2]# make upgrade #升级

#或者使用**killall nginx**杀死进程后再启动nginx。

#以下7行为make upgrade 提示信息

/usr/local/nginx/sbin/nginx -t

nginx: the configuration file /usr/local/nginx/conf/ngin

x.conf syntax is ok

nginx: configuration file /usr/local/nginx/conf/nginx.co

nf test is successful

kill -USR2 `cat /usr/local/nginx/logs/nginx.pid`

sleep 1

test -f /usr/local/nginx/logs/nginx.pid.oldbin

kill -QUIT `cat /usr/local/nginx/logs/nginx.pid.oldbin`

[root@proxy ~]# /usr/local/nginx/sbin/nginx –v #查看版本

**1.5 客户端访问测试**

1.5.1分别使用浏览器和命令行工具curl测试服务器页面

如果使用firefox火狐浏览器，注意在ssh远程的时候一定要加-X选项。

[root@client ~]# firefox http://192.168.4.5

[root@client ~]# curl http://192.168.4.5

**二 用户认证**

**2.1 问题**

沿用练习一，通过调整Nginx服务端配置，实现以下目标：

访问Web页面需要进行用户认证

用户名为：tom，密码为：123456

**2.2 方案**

通过Nginx实现Web页面的认证，需要修改Nginx配置文件，在配置文件中添加auth语句实现用户认证。最后使用htpasswd命令创建用户及密码即可。

实现此案例需要按照如下步骤进行。

**2.3 修改Nginx配置文件**

**主配置文件路径/usr/local/nginx/conf/nginx.conf**

**最基本格式:**

**server {**

**listen 端口号;**

**server\_name 域名;**

**root 网页根目录; #网页根目录可用相对路径,也可用绝对路径**

**} #以;号换行**

2.3.1 修改**/usr/local/nginx/conf/nginx.conf** #配置文件路径

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

.. ..

server {

listen 80;

server\_name localhost;

**auth\_basic "Input Password:"**; #认证提示符信息

**auth\_basic\_user\_file "/usr/local/nginx/pass"**;

#认证的密码文件,修改配置文件时不存在,需要另行创建

location / {

root html; #绝对路径为/usr/local/nginx/html

index index.html index.htm;

}

}

2.3.2生成密码文件，创建用户及密码

使用htpasswd命令创建账户文件，需要确保系统中已经安装了httpd-tools。

格式: htpasswd [-c] 密码文件路径 用户名

htpasswd -b [-c] 密码文件路径 用户名 密码

# -c表示新建,添加用户和密码时不使用-c

[root@proxy ~]# yum -y install httpd-tools

[root@proxy ~]# htpasswd -c /usr/local/nginx/pass tom

#创建密码文件

#/usr/local/nginx/pass 此路径及文件名由主配置文件指定

New password:

Re-type new password:

Adding password for user tom

[root@proxy ~]# htpasswd /usr/local/nginx/pass jerry

#追加用户，不使用-c选项,使用-c的话会覆盖之前的所有记录

New password:

Re-type new password:

Adding password for user jerry

[root@proxy ~]# cat /usr/local/nginx/pass

2.3.3重新加载配置

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s reload

#重新加载配置文件

#请先确保nginx是启动状态，否则运行该命令会报错,报错信息如下：

#[error] open() "/usr/local/nginx/logs/nginx.pid" failed (2: No such file or directory)

**2.4 客户端测试**

2.4.1登录192.168.4.10客户端主机进行测试

如果使用firefox火狐浏览器，注意在ssh远程的时候一定要加-X选项。

或者直接使用真实主机的火狐也可以。

[root@client ~]# firefox <http://192.168.4.5> #输入密码后可以访问

**三 基于域名的虚拟主机**

**1台服务器安装1个web服务,实现多个网站,达到省钱的目的**

**3.1 问题**

沿用练习二，配置基于域名的虚拟主机，实现以下目标：

实现两个基于域名的虚拟主机，域名分别为www.a.com和www.b.com

对域名为www.a.com的站点进行用户认证，用户名称为tom，密码为123456

**3.2 方案**

修改Nginx配置文件，添加server容器实现虚拟主机功能；对于需要进行用户认证的虚拟主机添加auth认证语句。

虚拟主机一般可用分为：基于域名、基于IP和基于端口的虚拟主机。

**3.3 步骤**

3.3.1修改配置文件

修改Nginx服务配置，添加相关虚拟主机配置如下

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

.. ..

server {

listen 80; #端口

server\_name [www.a.com;](http://www.a.com;) #域名

auth\_basic "Input Password:"; #认证提示符

auth\_basic\_user\_file "/usr/local/nginx/pass";

#认证密码文件

location / {

root html; #指定网站根路径

index index.html index.htm;

}

}

… …

server {

listen 80; #端口

server\_name [www.b.com;](http://www.b.com;) #域名

location / {

root www; #指定网站根路径

index index.html index.htm;

}

}

3.3.2创建网站根目录及对应首页文件

[root@proxy ~]# mkdir /usr/local/nginx/www

[root@proxy ~]# echo "www" > /usr/local/nginx/www/index.html

3.3.3重新加载配置

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s reload

#请先确保nginx是启动状态，否则运行该命令会报错,报错信息如下：

#[error] open() "/usr/local/nginx/logs/nginx.pid" failed (2: No such file or directory)

3.4 客户端测试

3.4.1 修改客户端主机192.168.4.10的**/etc/hosts**文件，进行**本地域名解析**

[root@client ~]# vim /etc/hosts

192.168.4.5    www.a.com www.b.com

3.4.2 登录192.168.4.10客户端主机进行测试

注意：请先关闭真实机的firefox，再SSH -X远程连接调用虚拟机的firefox。

[root@client ~]# firefox <http://www.a.com> #输入密码后可以访问

[root@client ~]# firefox <http://www.b.com> #直接访问

提示：或者直接使用真实主机做客户端主机验证，修改真实主机的/etc/hosts文件，直接使用真实主机的火狐浏览器访问也可以。

**3.5 扩展课外实验**：

yum -y install google\*simpli\*chinese\* yum方式安装中文简体字体

其他类型的虚拟主机

3.5.1 基于端口的虚拟主机（参考模板）

server {

listen 8080; #端口

server\_name web1.example.com; #域名

......

}

server {

listen 8000; #端口

server\_name web1.example.com; #域名

.......

}

3.5.2 基于IP的虚拟主机（参考模板）

server {

listen 192.168.0.1:80; #IP地址与端口

server\_name web1.example.com; #域名

... ...

}

server {

listen 192.168.0.2:80; #IP地址与端口

server\_name web1.example.com;

... ...

}

**四 SSL虚拟主机**

**4.1 问题**

沿用练习三，配置基于加密网站的虚拟主机，实现以下目标：

域名为www.c.com

该站点通过https访问

通过私钥、证书对该站点所有数据加密

**4.2 方案**

源码安装Nginx时必须使用--with-http\_ssl\_module参数，启用加密模块，对于需要进行SSL加密处理的站点添加ssl相关指令（设置网站需要的私钥和证书）。

加密算法一般分为对称算法、非对称算法、信息摘要。

对称加密算法：AES、DES，主要应用在单机数据加密。

非对加密称算法：RSA、DSA，主要应用在网络数据加密。

信息摘要：MD5、sha256，主要应用在数据完整性校验。

md5sum 文件名:查看文件校验码

**4.3 步骤**

4.3.1 配置SSL虚拟主机

生成私钥与证书

~]# cd /usr/local/nginx/conf #进入目录下

~]# openssl genrsa > cert.key #在目录下生成私钥

~]# openssl req -new -x509 -key cert.key > cert.pem

#在目录下生成证书(公钥)

4.3.2 修改Nginx配置文件，设置加密网站的虚拟主机(配置文件最下面一段)

~]# vim /usr/local/nginx/conf/nginx.conf

server {

listen 443 ssl;

server\_name www.c.com;

ssl\_certificate cert.pem; #这里是证书(公钥)文件

ssl\_certificate\_key cert.key; #这里是私钥文件

ssl\_session\_cache shared:SSL:1m;

ssl\_session\_timeout 5m; #配置超时时间5分钟

ssl\_ciphers HIGH:!aNULL:!MD5; #不能用空密码,不能用MD5加密

ssl\_prefer\_server\_ciphers on;

location / {

root html;

index index.html index.htm;

}

}

4.3.3 重新加载配置

~]# /usr/local/nginx/sbin/nginx -s reload

#请先确保nginx是启动状态，否则运行该命令会报错,报错信息如下：

#[error] open() "/usr/local/nginx/logs/nginx.pid" failed (2: No such file or directory)

4.3.4客户端验证

修改客户端主机192.168.4.10的/etc/hosts文件，进行域名解析

[root@client ~]# vim /etc/hosts

192.168.4.5 [www.c.com](http://www.c.com) [www.a.com](http://www.a.com) www.b.com

登录192.168.4.10客户端主机进行测试

[root@client ~]# firefox <https://www.c.com> #信任证书后可以访问