# day32-linux&nginx【linux高级&nginx】

# 学习目标

01\_能够使用Linux中的crontab命令

02\_能够使用Linux中的网络管理命令

03\_能够使用Linux中SSH免密登录命令

04\_能够使用Linux中基本用户管理命令

05\_能够使用Linux中防火墙配置命令

06\_能够理解web的通讯原理

07\_能够在windows中安装，配置和部署Nginx服务器

# 项目清单

|  |
| --- |
|  |

# 回顾与提问

1\_如何理解Linux是一个多用户，多任务的操作系统？

|  |
| --- |
| 多用户：同时可以多个用户操作Linux。  多任务：一个用户可以同时启动多个应用程序。 |

2\_rm -rf ./a\*？是什么意思？

|  |
| --- |
| 强制删除当前目录下的以a开头的文件/目录，且递归。 |

# 问题引入

我们程序员开发好了应用程序后，都希望放到一个安全稳定的电脑中长期运行，

且对互联网用户公开提供服务。但现实是安装了windows操作系统的电脑长期

运行后容易变慢，且容易招到病毒的攻击等等，这一切都限制了我们的应用程序

最终不能部署到windows操作系统的电脑中。那么我们会选用什么操作系统的电脑

来部署我们的应用程序呢？

答案是：Linux。

# 第01章 定时任务【重点】

## 1.1什么是定时任务

预先设定好的时间一到，或指定的间隔时间一到，由Linux系统自动调用并执行预先定义好的系统指令，来完成具体的业务功能。

## 1.2命令功能

通过Linux中的crontab 命令，我们预先设定好的时间一到，或指定的间隔时间一到，就执行定时任务。时间间隔的单位可以是分钟、小时、日、月、年及以上的任意组合。

这个命令非常适合做定时性，周期性的工作，如：数据备份，程序自动运行等。

## 1.3安装crontab服务

在默认情况下，Linux系统没有安装和启动crontab这个定时任务服务，我们得联网下载并安装启动这个crontab服务。

语法：yum install package

解释：下载并安装指定的安装包或服务

1. yum（全称为 Yellow dog Updater, Modified）：是一个软件包管理器。能够从指定的服务器自动下载安装包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软件包，无须繁琐地一次次下载和安装。yum提供了查找、安装、删除某一个、一组甚至全部软件包的命令，而且命令简洁而又好记。
2. package：表示需要指定的安装包或服务，如crontabs定时服务

命令：

|  |
| --- |
| yum install crontabs 安装crontab服务，注意crontabs后面有个s |

图示：

|  |
| --- |
|  |

## 1.4启动/停止/重启/加载定时任务

在默认情况下，当你下载并安装crontab定时任务这个服务后，crontab已经启动，

不用再启动了。当然你也可以根据需求停止定时任务这个服务。

语法：service 服务 start/stop/restart/reload

解决：将指定的服务启动/停止/重启/加载

命令：

|  |
| --- |
| 1. service crond start 启动定时任务服务  2. service crond stop 停止定时任务服务，但定时任务还在，下次可以再启动  3. service crond restart 重启定时任务服务  4. service crond reload 重新加载定时任务配置文件 |

图示：

|  |
| --- |
|  |

## 1.5业务需求

每隔一分钟，让Linux输出当前的系统时间到/mydate.txt文件中。

## 1.6定时任务配置

语法：crontab [参数]

解释：定时任务的配置

1. -u user 给指定用户设置定时任务

2. -l 显示当前用户所有的定时任务

3. -e 编辑当前用户的定时任务，一行一个定时任务

4. -r 删除当前用户的定时任务

命令：

|  |
| --- |
| crontab -l 显示当前用户所有的定时任务  crontab -e 编辑当前用户的定时任务，一行一个定时任务 |

图示：

|  |
| --- |
|  |

当输入crontab -e后，会启动vi编辑器，来编写新的定时任务，一行写一个定时任务。

|  |
| --- |
|  |

定时任务配置格式如下：

分 时 日 月 年 需要执行的命令，中间至少有一个空格

命令：

|  |
| --- |
| \* \* \* \* \* date >> /mydate.txt 每年每月每日每时每分执行date命令，  将结果输出到/mydate.txt文件中，  如果mydate.txt文件不存在，则系统创建。  其中>>表示将结果输出到什么位置。  \*/1 \* \* \* \* date >> /mydate.txt  每隔1分钟  \*/2 \* \* \* \* date >> /mydate.txt  每隔2分钟 |

图示：

|  |
| --- |
|  |

当保存并退出vi编辑器后，定时任务立刻生效。

等一分钟，显示当前用户所有定时任务，和执行的输出内容。

命令：

|  |
| --- |
| crontab -l 显示当前用户所有的定时任务  cd / 定位于根目录  ls 显示当前目录下的内容  more mydate.txt 查看当前目录下mydate.txt文件的内容 |

图示：

|  |
| --- |
|  |

其它定时任务配置示例：

|  |
| --- |
| 30 21 \* \* \* date >> /mydate.txt  每晚的21:30，将时间输出到/mydate.txt文件中  10 10 1,10,20,30 \* 2017,2018 date >> /mydate.txt  每月1，10，20，30号，的10点10分，将时间输出到/mydate.txt文件中。 |

# 第02章 网络管理

## 2.1什么是网络管理

以命令的形式，查看和设置网络中每台虚拟机的相关信息，如主机名，IP地址，映射域名，网络服务，端口号监听等服务。

## 2.2主机名配置

每台虚拟机的默认主机名为localhost，为了区分在网络中的每台虚拟机的主机名，建议为每台虚拟机设置一个有业务含义的名字，如itheima，taobao，baidu。

主机名设置是暂时的，不是永久的，停止虚拟机再启动，又恢复默认主机名为localhost。

语法：

|  |
| --- |
| hostname 获取主机名  hostname 主机名 设置主机名 |

需求：

|  |
| --- |
| 先获取当前主机名，  再将当前主机名设置为itheima。 |

命令：

|  |
| --- |
| hostname 获取主机名  hostname itheima 设置主机名 |

图示：

|  |
| --- |
|  |

## 2.3 IP地址配置

网络中的每一台虚拟机都应有自己唯一的IP地址，这样便于其它用户可以远程连接到这台虚拟机上并进行操作。

那么如何查看一台虚拟机的IP地址呢？

语法：

|  |
| --- |
| ip addr |

需求：

|  |
| --- |
| 获取本机的IP地址。 |

命令：

|  |
| --- |
| ip addr 获取本机的IP地址 |

图示：

|  |
| --- |
|  |

如果按上述命令，并没有显示虚拟机的IP地址，你得配置一下虚拟机的网卡了。

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

## 2.4域名映射【将来项目中用到】

当我们访问百度网站时，用户是通过域名访问百度，而不是通过IP地址访问百度。

域名方便我们记忆和书写，但IP地址不好记忆，书写麻烦。

所以我们得将域名和IP地址建立一种映射关系，让用户输入域名，就等价于用户输入域名对应的IP地址，这种方式叫域名映射。

一个IP地址，可以映射成多个域名。

|  |
| --- |
|  |

在Linux中，/etc/hosts文件用于在通过域名进行访问时做IP地址映射之用。

所以，你想访问一个什么样的域名，就需要把这个域名和它对应的IP地址配置在

/etc/hosts文件中。

语法：

|  |
| --- |
| vi /etc/hosts 用vi编辑器打开hosts文件  hosts文件的格式如下：  IP地址 域名 IP地址在前，域名在后  IP地址 域名  IP地址 域名 |

需求：

|  |
| --- |
| 将当前虚拟机的IP地址映射为baidu这个域名，  那么将来用户可以这样访问一个web应用的资源：  www.baidu.com  <http://baidu:8080/web/index.html>  等价于  <http://192.168.56.101:8080/web/index.html> |

命令：

|  |
| --- |
| vi /etc/hosts 用vi编辑器打开hosts文件  192.168.56.101 baidu |

图示：

|  |
| --- |
|  |

区别：

IP地址：是虚拟机在网络中唯一地址，强调数字形式，如身份证号码。

主机名：是虚拟机在网络中的真的名字，强调对内，如：周兵。

域名：是虚拟机在网络中的假的名字，强调对外，如：枪手，神枪手。

## 2.5网络服务管理

当虚拟机没有网络，或没有IP地址，或网络运行不正常，或设置网络服务在后台自动启动，这时我们通常会检查一下虚拟机的网络这个服务是否正常。

语法：

|  |
| --- |
| service 服务 status 查看服务的状态  service 服务 stop 停止服务  service 服务 start 启动服务  service 服务 restart 重启服务  chkconfig 查看所有服务自启配置  chkconfig 服务 off 关掉服务的自动启动  chkconfig 服务 on 开启服务的自动启动 |

需求：

|  |
| --- |
| 将network这个服务，设置为自动启动。 |

命令：

|  |
| --- |
| chkconfig 查看所有服务自启配置  chkconfig network off 关掉网络服务的自动启动  chkconfig network on 开启网络服务的自动启动 |

图示：

|  |
| --- |
|  |

|  |
| --- |
|  |

## 2.6进程端口的监听

有时我们想查看后台进程运行的相关信息。

语法：

|  |
| --- |
| netstat [参数] 显示进程的相关信息  1. -a 显示所有连线中的Socket |

需求：

|  |
| --- |
| 查看当前虚拟机上运行进程的相关信息。 |

命令：

|  |
| --- |
| netstat -a 显示当前虚拟机上运行进程的协议，IP，状态等信息 |

图示：

windows中的情况：



Linux中的情况：

|  |
| --- |
|  |

# 第03章 免密登录【重点】

## 3.1什么是免密登录

在企业中，一个网络中的多台Linux机器是可以相互之间进行登录和退出的。

用Linux自带的SSH命令，可以在输入正确密码的方式下，成功登录到另一台

Linux机器中。

但每次登录另一台Linux机器都要输入正确密码，难免让用户觉得麻烦。

那么能不能一个用户在不输入正确密码的情况下，也能登录另一台Linux机器呢？

答案是肯定的，这就是SSH免密登录。

## 3.2准备二台虚拟机

|  |
| --- |
|  |

## 3.3了解一下SSH工作机制

SSH是Secure Shell（安全外壳协议）的缩写。

SSH是专为远程登录会话和其他网络服务提供安全性的协议。

SSH的具体实现是由客户端和服务端的软件组成的：

1. 服务端是一个守护进程，他在后台运行并响应来自客户端的连接请求。

2. 客户端可以使用Linux自带的SSH工具，进行远程连接服务端。

SSH采用二种验证机制：

1. 基于口令的安全验证

只要你知道另一台机器上用户的帐号和口令，就可以登录到远程主机。

root除外。

2. 基于密钥的安全验证

你无需知道另一台机器上用户的帐号和口令，也可以登录到远程主机。

我们说的SSH免密登录，就说的是这种方式

## 3.4用SSH命令从一台虚拟机有密登录到另一台虚拟机步骤

1. 在JavaEE82虚拟机中输入：ssh 192.168.56.102

192.168.56.102是JavaEEOther虚拟机的IP地址

|  |
| --- |
|  |

1. 查看JavaEEOther虚拟机的IP地址，在JavaEE82虚拟机中输入：ip addr

|  |
| --- |
|  |

1. 退出JavaEEOther虚拟机，回到JavaEE82虚拟机，在JavaEE82虚拟机中输入：exit

|  |
| --- |
|  |

## 3.5了解一下免密登录工作原理

1. 需要依靠密匙，也就是你必须为自己创建一对密钥（私钥和公钥），并把公钥放在需要访问的虚拟机/电脑/服务器上。如果你要连接到SSH服务器上，客户端软件就会向服务器发出请求，请求用你的公钥进行安全验证。
2. 服务器收到请求之后，先在该服务器上你的主目录下寻找你的公钥，然后把它和你发送过来的公钥进行比较。如果两个公钥一致，服务器就用公钥加密并把它发送给客户端软件。
3. 客户端软件收到公钥之后就可以用你的私钥进行解密再把它发送给服务器。

如图所示：

|  |
| --- |
|  |

## 3.6用SSH命令从一台虚拟机免密登录到另一台虚拟机步骤

1. 在JavaEE82虚拟机中输入：ssh-keygen，

如有提示，按回车，生成JavaEE82这台虚拟机的公钥和私钥。

|  |
| --- |
|  |

1. 在JavaEE82虚拟机中输入：ssh-copy-id 192.168.56.102，如有提示，按回车，

将刚刚生成的JavaEE82虚拟机的公钥复制到JavaEEOther这台虚拟机中，

这时免密配置成功。

192.168.56.102是JavaEEOther虚拟机的IP地址。

|  |
| --- |
|  |

1. 在JavaEE82虚拟机中输入：ssh 192.168.56.102，不再需要密码了。

192.168.56.102是JavaEEOther虚拟机的IP地址。

|  |
| --- |
|  |

# 第04章 基本用户和用户组管理

## 4.1什么是基本用户和用户组管理

在Linux系统中，除超级管理员外，其它普通用户都是以组为单位的，以组为单位设置该组

可以操作的内容等。

那么做为Linux管理员，得学会创建用户，修改用户，创建用户组，修改用户组等操作。

可以将超级管理员root，看作是root组。

## 4.2用户操作

### 4.2.1创建新用户【useradd -m】

语法：useradd -m 新用户名

解释：创建新的用户名，-m表示在/home目录下会创建一个子目录，目录名与用户名相同

命令：

|  |
| --- |
| useradd -m zhaojun 创建新用户zhaojun |

图示：

|  |
| --- |
|  |

### 4.2.2设置新用户密码【passwd】

语法： passwd 用户名

解释：为新创建的用户设置一个至少8位数的密码

命令：

|  |
| --- |
| passwd zhaojun 设置新用户zhaojun的密码  \*\*\*\*\*\*\*\* 密码设置不会回显 |

图示：

|  |
| --- |
|  |

普通用户：zhaojun

密码：12345678

### 4.2.3用新用户登录JavaEE82虚拟机步骤

1. 打开SSH Secure客户端工具，用zhaojun普通用户，12345678密码，登录Linux。

|  |
| --- |
|  |

1. 打开JavaEE82虚拟机，用zhaojun普通用户，12345678密码，登录Linux。

|  |
| --- |
|  |

注意：root是超级管理员，提示符：#

zhaojun是普通用户，提示符：$

### 4.2.4修改用户名【usermod】

语法：usermod -l 新用户名 旧用户名

解释：修改旧用户名为新用户名，但该用户所在组名不变

你可以使用新用户名登录Linux，但旧用户名就不能再登录Linux了。

1. -l 修改用户名

命令：

|  |
| --- |
| usermod -l itheima zhaojun 修改zhaojun用户名为itheima用户名 |

图示：

|  |
| --- |
|  |

|  |
| --- |
|  |

## 4.3 用户组操作

### 4.3.1创建新用户组【groupadd】

语法：groupadd 新用户组名

解释：创建新的用户组

命令：

|  |
| --- |
| groupadd j2se 创建新用户组j2se |

图示：

|  |
| --- |
|  |

### 4.3.2修改用户组名【groupmod -n】

语法：groupmod -n 新用户组名 旧用户组名

解释：修改旧用户组名为新用户组名

1. -n 设置新用户组名

命令：

|  |
| --- |
| groupmod -n javaee j2se 将新用户组名j2se修改为javaee |

图示：

|  |
| --- |
|  |

### 4.3.3将用户添加到新用户组中【usermod -g】

语法：usermod -g 新用户组名 用户名

解释：将用户添加到新用户组中去

解释：

1. -g 修改用户所属的用户组

命令：

|  |
| --- |
| usermod -g javaee itheima 将itheima用户添加到javaee新用户组中 |

图示：

|  |
| --- |
|  |

### 4.3.4从新用户组中删除用户【gpasswd -d】

语法：gpasswd -d 用户名 新用户组名

解释：将用户从新用户组中删除

1. -d 从新用户组中删除用户

命令：

|  |
| --- |
| gpasswd -d itheima javaee 从javaee新用户组中删除itheima用户 |

图示：

|  |
| --- |
|  |

## 4.3为普通用户配置sudo权限

sudo，可以理解为超级管理员执行的操作，通常是系统级别的指令，只有超级管理员才有这个权限。

如果一个普通用户想执行超级管理员的系统级别的指令，如创建新用户等，那是不可能的。

除非超级管理员授予普通用户sudo权限。

下面我们就来让超级管理员root授予普通用户itheima的sudo权限。

1. 在JavaEE82虚拟机中，以root超级管理员登录，输入：vi /etc/sudoers

这是一个只读文件。

|  |
| --- |
|  |

1. 在/etc/sudoers文件中找到root这一行，在下面添加一行，内容如下，保存并退出

不一定修改成功。

|  |
| --- |
|  |

1. 在JavaEE82虚拟机中，以itheima普通用户登录，输入：sudo useradd -m haha

命令：

|  |
| --- |
| itheima@localhost ~]$ sudo useradd -m haha 创建新用户haha |

查询命令的帮助：man 命令

如：man ls

# 第05章 Web的通讯原理【重点】

学习Java开发Web应用之前，首先我们得对日常生活中，用户通过用浏览器访问服务器之间的通讯过程要了解一下，基本过程如图所示：

|  |
| --- |
|  |

解释如下：

1. Web浏览器建立与Web服务器的TCP连接。

2. Web浏览器向Web服务器发送请求命令。

3. Web浏览器发送请求信息。

4. Web服务器处理中。

5. Web服务器发送响应信息。

6. Web服务器向浏览器发送数据。

7. Web服务器关闭与Web浏览器的TCP连接。

# 第06章 Nginx【重点】

## 6.1什么是Nginx

Nginx是一种服务器软件，其最主要，最基本的功能是可以与服务器硬件结合，让程序员可以将程序发布在Nginx服务器上，让成千上万的用户可以浏览。

除此之外，Nginx还是一种高性能的HTTP和反向代理服务器，同时也是一个代理邮件服务器。也就是说，我们在Nginx上可以：

1. 可以发布网站 ，只能处理用户对html页面的访问。

2. 可以实现负载均衡

3. 可以作为邮件服务器实现收发邮件等功能

本课程我们只讨论Nginx发布网站的功能。

所谓的负载均衡是指，当同时有多个用户访问我们的服务器的时候，为了减少服务器的压力，我们需要将用户引入服务器集群中的较轻压力的服务器，分担较重压力服务器的负担。

|  |
| --- |
|  |

## 6.2在Windows中安装Nginx步骤

注意：步骤无需强行记忆，初学者按文档步骤一步一步的操作。

1. 进入http://nginx.org/网站，下载nginx-1.13.8.zip文件。

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 进入CMD控制台模式，并切换到Nginx安装文件所在的目录。

|  |
| --- |
|  |

1. 用start nginx命令，启动Nginx服务器。

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 打开Windows中的浏览器，访问本机中Nginx服务器首页，

Nginx服务器默认监听80端口。

|  |
| --- |
| <http://127.0.0.1:80>  或  http://localhost |

|  |
| --- |
|  |

1. 用nginx -s stop命令，停止Nginx服务器。

|  |
| --- |
|  |

|  |
| --- |
|  |

## 6.3在Windows版的Nginx中配置和发布HTML页面步骤

1. 修改Nginx目录中conf目录下的nginx.conf配置文件，并保存。

|  |
| --- |
|  |

1. 用start nginx命令，启动Nginx服务器。

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 用nginx -s stop命令，停止Nginx服务器。

|  |
| --- |
|  |

|  |
| --- |
|  |

## 6.4在Linux中安装Nginx步骤

注意：步骤无需强行记忆，初学者按文档步骤一步一步的操作。

1. 进入http://nginx.org/网站，下载nginx-1.13.9.tar.gz文件。

|  |
| --- |
|  |

1. 用Windows客户端工具SSH Secure上传nginx-1.13.9.tar.gz文件到Linux的root目录下。

|  |
| --- |
|  |

1. 进入Linux的root目录下。

|  |
| --- |
|  |

1. 连接网络，安装Nginx依赖环境gcc

Nginx是C/C++语言开发，建议在Linux上运行，安装Nginx需要先将官网下载的源码进行编译，编译依赖gcc环境，所以需要安装gcc。

|  |
| --- |
| yum install gcc-c++ 安装gcc，期间有提示，一律选y，  gcc-c++中间无空格 |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 连接网络，安装Nginx依赖环境pcre/zlib/openssl。

|  |
| --- |
| yum -y install pcre pcre-devel  yum -y install zlib zlib-devel  yum -y install openssl openssl-devel  -y表示安装过程如有提示，默认选择y |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 解压nginx-1.13.9.tar.gz文件，并进入Nginx安装目录。

|  |
| --- |
| tar -xvf nginx-1.13.9.tar.gz -C /usr/local  解压nginx-1.13.9.tar.gz文件到/usr/local目录下  cd /usr/local/nginx-1.13.9  进入/usr/local/nginx-1.13.9这个目录 |

|  |
| --- |
|  |

1. 根据gcc等环境信息，编译Nginx并安装Nginx。

|  |
| --- |
| ./configure 配置nginx，在nginx-1.13.9目录中执行这个配置文件  make 编译nginx  make install 安装nginx，会在与nginx-1.13.9同级的目录中产生一个nginx执行目录 |

|  |
| --- |
|  |

1. 进入/usr/local/nginx/sbin这个目录，启动Nginx服务器。

|  |
| --- |
| cd /usr/local/nginx/sbin 进入/usr/local/nginx/sbin这个目录  ./nginx 启动Nginx |

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 开放Linux的对外访问的端口80，在默认情况下，Linux不会开放端口号80。

涉及到Linux防火墙命令。

|  |
| --- |
| /sbin/iptables -I INPUT -p tcp --dport 80 -j ACCEPT  Linux防火墙开放80端口  外界通过80端口请求可以进入虚拟机 |

|  |
| --- |
|  |

1. 打开Windows中的浏览器，访问虚拟机中Nginx服务器首页，

Nginx服务器默认监听80端口。

|  |
| --- |
| <http://192.168.56.101:80> |

|  |
| --- |
|  |

1. 进入/usr/local/nginx/sbin这个目录，停止Nginx服务器。

|  |
| --- |
| cd /usr/local/nginx/sbin 进入/usr/local/nginx/sbin这个目录  ./nginx -s stop 停止Nginx |

|  |
| --- |
|  |

|  |
| --- |
|  |

## 6.5在Linux版的Nginx中配置和发布HTML页面步骤

1. 修改/usr/local/nginx/conf/目录中的nginx.conf配置文件，并保存。

|  |
| --- |
| cd /usr/local/nginx/conf 切换到/usr/local/nginx/conf这个目录  ls 查看当前目录中是否有nginx.conf配置文件  vi nginx.conf 用vi修改这个nginx.conf文件后，保存退出 |

|  |
| --- |
|  |

修改nginx.conf内容如下：

|  |
| --- |
|  |

1. 进入/usr/local/nginx/sbin这个目录，启动Nginx服务器。

|  |
| --- |
| cd /usr/local/nginx/sbin 进入/usr/local/nginx/sbin这个目录  ./nginx 启动Nginx |

|  |
| --- |
|  |

1. 打开Windows中的浏览器，访问虚拟机中Nginx服务器首页，

Nginx服务器默认监听80端口。

|  |
| --- |
| <http://192.168.56.101:80> |

|  |
| --- |
|  |

# 随堂测试

# 附：防火墙管理

## 什么是防火墙管理

防火墙类似于一个安全卫士管家，它能对你访问别人电脑，别人访问你的电脑，进行拦截并处理，有的阻止，有的放行，有的转发。

Linux中防火墙配置根据/etc/sysconfig/iptables配置文件来控制本机的“出、入”网络访问行为，其对行为的配置策略有四个策略表，如下图：

|  |
| --- |
|  |

|  |
| --- |
|  |

上述表解释如下，同学们了解一下即可：

1. raw表

主要用于决定数据包是否被状态跟踪机制处理。在匹配数据包时，raw表的规则要优先于其他表。

2. mangle表

主要用于修改数据包的TOS（Type Of Service，服务类型）、TTL（Time To Live，生存周期）指以 及为数据

包设置Mark标记，以实现Qos(Quality Of Service，服务质量)调整以及策略路由等应用，由于需要相应的路

由设备支持，因此应用并不广泛。

3. nat表

主要用于修改数据包的IP地址、端口号等信息（网络地址转换，如SNAT、DNAT、MASQUERADE、

REDIRECT）。属于一个流的包 (因为包的大小限制导致数据可能会被分成多个数据包)只会经过这个表一次。

如果第一个包被允许做NAT或Masqueraded，那么余下的包都会自动地 被做相同的操作，也就是说，余下的

包不会再通过这个表。

4. filter表

主要用于对数据包进行过滤，根据具体的规则决定是否放行该数据包（如DROP、ACCEPT、REJECT、LOG）。filter 表对应的内核模块为iptable\_filter，包含三个规则链：

INPUT链：INPUT针对那些目的地是本地的包

FORWARD链：FORWARD过滤所有不是本地产生的并且目的地不是本地(即本机只是负责转发)的包

OUTPUT链：OUTPUT是用来过滤所有本地生成的包

在默认情况下，Linux系统的防火墙状态是打开的，已经启动。

## 查看防火墙状态

语法：

|  |
| --- |
| service 服务 status |

命令：

|  |
| --- |
| service iptables status |

## 关闭防火墙

语法：

|  |
| --- |
| service 服务 stop |

命令：

|  |
| --- |
| service iptables stop |

## 启动防火墙

语法：

|  |
| --- |
| service 服务 start |

命令：

|  |
| --- |
| service iptables start |

## 禁止防火墙自启

语法：

|  |
| --- |
| chkconfig 服务 off |

命令：

|  |
| --- |
| chkconfig iptables off |

## 允许防火墙自启

语法：

|  |
| --- |
| chkconfig 服务 on |

命令：

|  |
| --- |
| chkconfig iptables on |