Linux网络基础知识

# 配置linux服务器的网络

## 网络相关概念

### Ip地址

Ip地址的作用是用来识别网络中不同的节点。Ip地址有两种，一种是IPv4和IPv6。我们一般使用的ip地址是IPv4，它是由32位（4个字节）二进制数字组成，我们通常使用点分十进制。例如：172.168.0.1

Ip地址的分类：A(1-126),B(128-191),C(192-223)，D(组ip地址)，E(保留)。

私有地址：10.0.0.0~10.255.255.255 172.16.0.0~172.31.255.255

192.168.0.0~192.168.255.255

### 路由器的转发功能

路由器用于不同网段地址的通讯。

### DNS

Ip地址来访问地址不容易记忆。所以我们使用DNS域名解析服务器来进行ip地址的正向解析和ip地址的反向解析。

### 网络接口

lo接口：本地回送地址（Loop），用于网络测试及本地主机各网络之间的通讯。

eth接口：Ethernet以太网接口白哦是网卡设备接口。附加数字表示物理网卡的序号，从0开始。

ppp接口：表示PPP设备接口，附加数字表示PPP设备的序号，从0开始。

### 网络端口

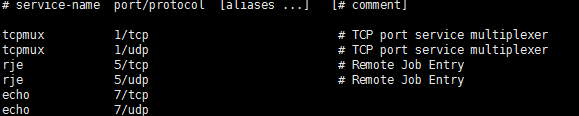
端口号主要来区分不同的应用进程。端口号的范围0~65535

常用服务端口0~255 256~1024专用网络端口 1024以上动态分配

## 配置linux服务器的网络

### /etc/services文件

列出了系统所有可用的网络服务和它们对应的端口号及协议。文件每一行对应一个服务，它由4个字段组成：服务名称 端口 协议名称 别名



### 修改和配置网卡文件

**通过配置文件修改**：vi /etc/sysconfig/network-scripts/ifcfg-eth0

|  |
| --- |
| DEVICE=eth0 #网卡代号，需要ifcfg-eth0相对应  BOOTPROTO=dhcp #ip获取协议，static静态，dhcp动态获取  HWADDR=00:24:8C:79:1A:2B #网卡地址即MAC地址  IPV6 INIT=yes #表示支持IPv6，no表示不支持  IPV6\_AUTOCONF=yes #表示自动配置ipv6  ONBOOT=yes #是否开机启动  NETMASK=255.255.0.0 #子网掩码  IPADDR=172.16.22.22 #ip地址  GATEWAY=192.128.1.1 #设置默认网关  TYPE=Ethernet #表示网卡的类型为以太网  PEERDNS=yes #表示允许从DHCP获得的DNS覆盖本地DNS  USERCTL=no #表示不允许普通用户修改配置 |

#重启网络服务配置生效

service network restart

注意：centos可以使用setup命令来修改网络参数

**通过命令修改：**

|  |
| --- |
| #手工设置ip和子网掩码  ifconfig eth0 ip netmask 子网掩码  #启动和断开网卡  ifconfig eth0 up/down |

### 修改机器hostname

**通过配置文件修改：**vi /etc/sysconfig/network

|  |
| --- |
| NETWORKING=yes # yes启动网络no关闭网络  HOSTNAME=hostname #主机名  GATEWAY=10.10.28.0 #设置路由 |

#重启网络服务配置生效

service network restart

**通过命令来修改**：

|  |
| --- |
| #查看和设置主机名的命令  hostname [主机名] |

### 修改主机名查询静态表/etc/hosts

/etc/hosts告诉主机域名和ip的对应关系。

|  |
| --- |
| #网络ip地址 主机名或者域名 主机别名  127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4  ::1 localhost localhost.localdomain localhost6 localhost6.localdomain6 |

注意：主机名和域名的区别，主机名通常在局域网内部使用，通过hosts文件，主机被解析到对应的ip地址上。域名通常使用在Internet上使用，如果本机不想使用Internet上的域名解析，可以更改hosts文件，加入自己的域名解析。可以在局域网中使用。

### 配置DNS域名解析服务器

配置DNS域名解析服务器/etc/resolv.conf

|  |
| --- |
| #声明主机域名，可以用domain替代，但是它只能接一个域名  search localdomain  #指定多个服务器名字，会按给出的先后顺序来决定主从服务器  nameserver 10.202.72.116  nameserver 10.202.72.118 |

### 配置路由参数

**通过配置文件修改**：vim /etc/sysconfig/network-scripts/route-eth1

|  |
| --- |
| #每一行对应一条静态路由记录  0.0.0.0/0 via 218.244.135.247 dev eth1 |

#重启网络服务

service network restart

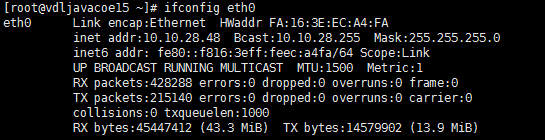
**通过命令修改**：

|  |
| --- |
| #使用route 命令添加的路由，机器重启或者网卡重启后路由失效  #-net：到一个网络的路由表； -host：到一个主机的路由表。  #dev：路由记录所表示的网络接口。  #gw：设置默认网关；  #添加到主机的路由  route add –host 192.168.1.11 dev eth0  route add –host 192.168.1.12 gw 192.168.1.1  #添加到网络的路由  route add –net 192.168.1.11 netmask 255.255.255.0 dev eth0  route add –net 192.168.1.11 netmask 255.255.255.0 gw 192.168.1.1  route add –net 192.168.1.0/24 dev eth1  //添加默认网关  route add default gw 192.168.2.1  //删除路由  route del –host 192.168.1.11 dev eth0 |

# 查看Linux服务器的网络连接

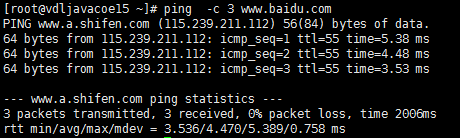
## 查看网络接口信息

ifconfig –a [网络接口名]



## 测试网络连接状态

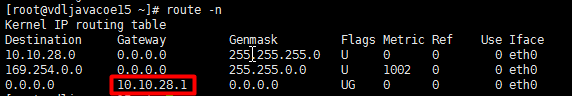
ping -c 3 [主机或者域名]



## 查看主机路由器

#查看路由，目标地址和子网掩码都为0.0.0.0的表示默认路由，标志显示UG。

route -n

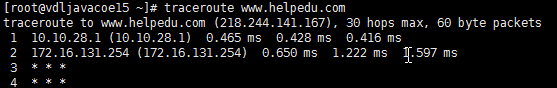


Flags为路由标志，标记当前网络节点的状态，Flags标志说明：

* U Up表示此路由当前为启动状态。
* H Host，表示此网关为一主机。
* G Gateway，表示此网关为一路由器。
* R Reinstate Route，使用动态路由重新初始化的路由。
* D Dynamically,此路由是动态性地写入。
* M Modified，此路由是由路由守护程序或导向器动态修改。
* ! 表示此路由当前为关闭状态。

## 跟踪数据包经过的路由

traceroute [ip/域名]



## 测试DNS服务器是否能正常解析

### nslookup命令

命令格式：nslookup [目标ip/域名] [给定DNS服务器ip]

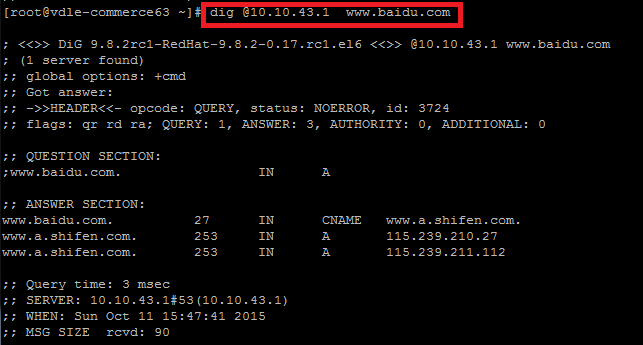


### dig命令

dig [@server] [FQDN] [type]

参数说明： @server 如果不想以 /etc/resolv.conf 作为DNS主机，可以填写其他的DNS服务器ip。 FQDN：完全合格的域名。

type 预设查询A标志，也可以填写其他的标志，如 mx, ns 等。



## 查看网络连接状态

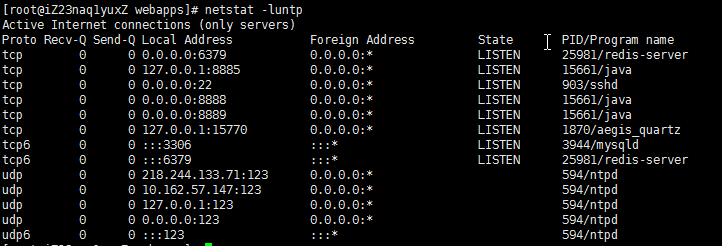
netstat [-anrltup]

* a：显示所有
* n：用数字显示
* r：显示路由器
* l：显示处于监听状态的进程
* t：显示TCP进程
* u：显示UDP进程
* p：显示进程号进程名信息

常用：netstat –luntp #查看系统监听端口

**找出程序运行的端口：netstat -ap | grep ssh**

找出运行在指定端口的进程：netstat -an | grep ':80'



## write指令

指令名称：write

指令所在路径：/usr/bin/write

执行权限：All User

语法：write <用户名> //发送消息给这个系统的其他用户。

功能描述：向另外一个用户发信息，以Ctrl+D作

为结束

范例： $ write webmaster

## wall指令

指令名称：wall

指令所在路径：/usr/bin/wall

执行权限：All User

语法：wall [message] [文件名]

功能描述：向所有用户广播信息

范例： $ wall Happy New Year!

# Linux网络服务

## Linux下网络服务的分类

网络服务启动脚本在/etc/rc.d/init.d中(rpm包服务), rpm安装软件的默认标准：

* /etc:配置文件位置
* /etc/init.d/或者/etc/rc.d/init.d独立的服务脚本
* /etc/xinetd.d：基于xinetd服务的启动脚本
* /etc/xinetd.conf:xinetd配置文件
* /etc/sysconfig/:初始化环境配置文件的位置
* /var/lib:服务产生的数据的位置
* /var/log/:日志文件

源代码服务，一般安装在/usr/local下。

### 独立性服务

每个独立性服务都有自己的守护进程。如www对应httpd

### 依赖性服务

服务被其他的进程所管理，这个进程叫超级进程xinetd。通过

/etc/xinetd.conf来管理。xinetd服务现在基本上不怎么实用

xinetd的安装：yum -y install xinetd

xinetd的配置文件：/etc/xinetd.d/...(所有xinetd服务)

/etc/xinetd配置文件解析：

|  |
| --- |
| defaults  {  log\_type = SYSLOG daemon info #日志类型  log\_on\_failure = HOST #失败日志选项  log\_on\_success = PID HOST DURATION EXIT #成功日志选项  cps = 50 10 #每秒的连接数及服务启动间隔  instances = 50 #可以启动的实例最大值  }  includedir /etc/xinetd.d #该目录显示xinetd进程管理的所有服务。  xinetd启动和自启动是同步的。(disable=no) |

### 源代码服务

源代码服务的启动可以参考源代码安装说明~/INSTALL

源代码的服务可以使用链接二进制命令到/etc/init.d/目录，这样就可以使用service命令。ls /etc/rc.d/rcn.d/中：

* s后面的数字表示系统启动顺序，
* k代表结束顺序，可以用来确定系统的启动和关闭顺序

源代码服务要让chkconfig有效可以使用：连接到/etc/init.d/的文件中加入：

|  |
| --- |
| #chkconfig: level(35)  启动顺序   关闭顺序  #description:描述 |

例如：

|  |
| --- |
| ln -s /usr/local/apache2/bin/apachetcl /etc/init/apache  vi /etc/init/apache中添加注释（开始）  #chkconfig:35 86 76  #description:source package apache  chkconfig --add apache  chkconfig apache on |

## Linux下网络服务的配置方法

### 查看及配置服务的启动状态

ntsysv(\*代表系统启动自动打开，空格进行选取，tab进行跳变)

### service命令

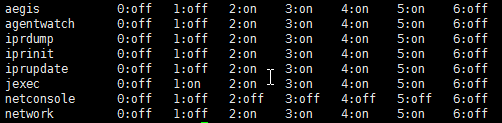
命令：service 服务守护进程名 start/stop/restart/status

service --status-all #查询所有安装的服务运行状态

守护进程名必须和/etc/rc.d/init.d/目录的文件名相对应

### chkconfig命令

|  |
| --- |
| chkconfig(选项)  --add：增加所指定的系统服务，让chkconfig指令得以管理它，并同时在系统启动的叙述文件内增加相关数据；  --del：删除所指定的系统服务，不再由chkconfig指令管理，并同时在系统启动的叙述文件内删除相关数据；  --level<等级代号>：指定读系统服务要在哪一个执行等级中开启或关毕。 |
| --实例  #查看，设置服务的启动状态  chkconfig --list [服务名]  #设置服务的默认启动方式  chkconfg --level 运行级别列表 服务名称 on/off/reset  #增加httpd服务  chkconfig --add httpd  #删除httpd服务  chkconfig --del httpd |



### 服务开机自启方式

|  |
| --- |
| 1. chkconfig --level 2345 on/off 2. 修改/etc/rc.d/rc.local文件(源代码安装服务适合使用) 3. 使用ntssysv(redhat专有)  推荐使用修改配置文件 |

## CentOS7中启动/停止/重启服务命令

CentOS 7.0中一个最主要的改变，就是切换到了systemd。它用于替代红帽企业版Linux前任版本中的SysV和Upstart，对系统和服务进行管理。systemd兼容SysV和Linux标准组的启动脚本。**Systemd**是一个Linux操作系统下的系统和服务管理器。它被设计成向后兼容SysV启动脚本，并提供了大量的特性，如开机时平行启动系统服务，按需启动守护进程，支持系统状态快照，或者基于依赖的服务控制逻辑。

先前的使用SysV初始化或Upstart的红帽企业版Linux版本中，使用位于/etc/rc.d/init.d/目录中的bash初始化脚本进行管理。而在RHEL 7/CentOS 7中，这些启动脚本被服务单元取代了。服务单元以.service文件扩展结束，提供了与初始化脚本同样的用途。要查看、启动、停止、重启、启用或者禁用系统服务，你要使用systemctl来代替旧的service命令。

systemctl可以看作是service和chkconfig的组合：

|  |
| --- |
| #服务是否开机自启动  systemctl enable/ disable httpd |
| #启动停止和重启服务  systemctl start/stop/restart httpd |
| #查看服务状态  systemctl list-units --type=service(所有服务的启动状态)  systemctl status httpd(服务详细信息)  systemctl is-active httpd(只显示是否active) |