



链滴

Linux 系统初始化专题

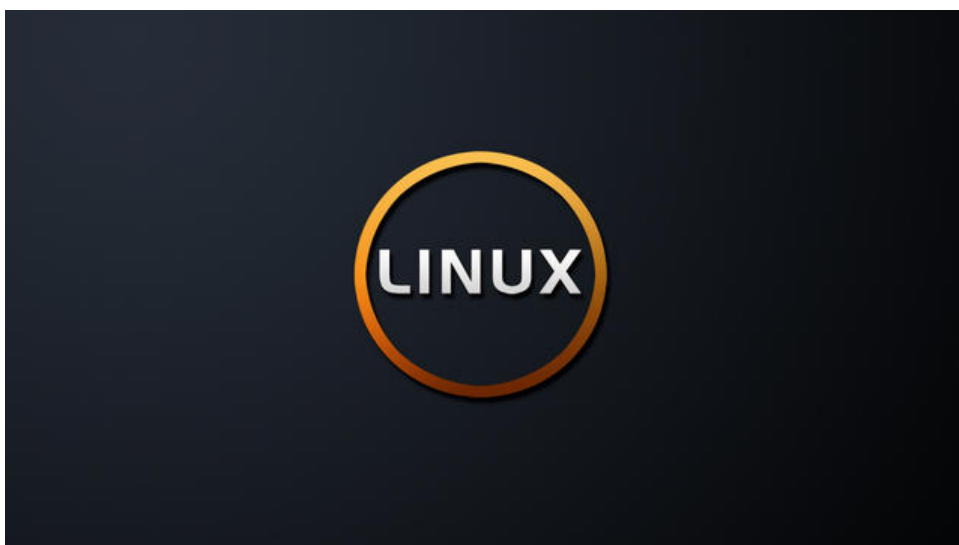
作者: [prifi](#)

原文链接: <https://ld246.com/article/1576423806563>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

前言：Linux 系统第一次安装完成后，总是需要进行一些初始化配置以及调优，这里进行一个总结以查。



一、Ubuntu 16.04系统初始化

- 修改主机名
- 关闭防火墙/selinux
- 网络配置
- 时间同步
- 初始化磁盘
- 配置ulimit
- 添加秘钥（允许或禁止账号密码登录）
- 配置 ssh 免密登录（可选）
- 添加阿里云镜像源（选择不同Linux发行版）
- 安装本地依赖及常用工具（可选）
- Linux 内核优化参数

修改主机名：

```
# Set hostname
HOSTNAME=""
hostnamectl set-hostname ${HOSTNAME}
```

```
# Binding Hosts
cat << EOF >> /etc/hosts
## SET CEPH HOSTS
${IP} ${HOSTNAME}
EOF
```

关闭防火墙/selinux：

```
# Disable iptable
```

```
ufw disable
ufw status
```

网络配置:

这里注意, 有些Linux发行版安装完系统后初始网卡为 `ens33` 或 `enp2s0`, 如何修改为 `eth0`:

```
vi /etc/default/grub
```

```
# 修改参数 GRUB_CMDLINE_LINUX 最后添加 net.ifnames=0 biosdevname=0;
update-grub
```

然后再修改网卡对应 `eth0` 并重启:

```
vi /etc/network/interfaces
## 静态IP
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.0.1
netmask 255.255.255.0
gateway 192.168.0.1
dns-nameservers 8.8.4.4 8.8.8.8
```

```
## DHCP
auto lo
iface lo inet loopback
```

```
auto eth0
iface eth0 inet dhcp
```

```
## 重启网卡
systemctl restart ifup@eth0
systemctl enable ifup@eth0
```

```
## DNS配置
vi /etc/resolv.conf
nameserver 8.8.4.4
nameserver 8.8.8.8
```

注意: ubuntu配置网络不生效需要重启`reboot`生效

时间同步:

```
## 设置时区
ln -snf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
bash -c "echo 'Asia/Shanghai' > /etc/timezone"
# timedatectl set-timezone UTC
```

```
## 同步网络时间
apt-get install -y ntpdate
# /etc/init.d/ntp stop
ntpdate ntp1.aliyun.com
```

```
## 写入硬件时间
date
hwclock --systohc
```

初始化磁盘:

这里使用脚本实现, 会自动创建 `/data` 目录挂载至指定磁盘 (格式为: ext4), 使用方法: `./part.sh v b`:

```
cat << 'EOF' > part.sh
#!/bin/bash
#
if [ ! -n "$1" ]; then
    echo '需要输入一个磁盘参数' ?

    exit 1
else
    disk=`lsblk|grep $1|grep disk`
    if [ ! -n "$disk" ]; then
        echo '磁盘输入有误'
        exit 1
    fi
fi

diskstat=`lsblk|grep $1|grep part`
if [ -n "$diskstat" ]; then
    echo '磁盘已分区'
    exit 1
else
    echo "n
p
1

w
" | fdisk /dev/$1
disknum=`lsblk -l|grep $1|grep part|awk '{print $1}'`
mkfs.ext4 /dev/$disknum
uuid=`blkid /dev/$disknum |awk '{print $2}'`
fstabstat=$(echo $uuid |awk -F=' ' '{print $NF}')
if [ -n "$fstabstat" ]; then
    echo "${uuid} /data ext4 defaults 0 0" >> /etc/fstab
    mkdir -p /data
    mount -a
fi
fi
EOF
```

如果需要配置LVM, 参考以下脚本:

```
apt-get install lvm2 -y
LV_SIZE='299'
pvcreate /dev/vdb && vgcreate vgdata /dev/vdb && \
lvcreate -L ${LV_SIZE}G -n data vgdata && df -Th &&\
```

```
mkfs.ext4 /dev/vgdata/data && mkdir /data && \
echo "UUID=\"44bde ...\" /data ext4 defaults 0 0" >> /etc/fstab && mount -a && df -Th
```

注意： 挂载磁盘尽量使用UUID, `blkid /dev/vdb1`

配置 ulimit:

```
ulimit -a
ulimit -n 65535
```

```
echo "* hard nofile 65535
* soft nofile 65535
root hard nofile 65535
root soft nofile 65535" >> /etc/security/limits.conf
```

生成秘钥 (允许或禁止账号密码 ssh 登录) :

```
## 允许 Root 密码 ssh 登录
sed -i 's/prohibit-password/yes/' /etc/ssh/sshd_config
sed -i 's/PasswordAuthentication no/PasswordAuthentication yes/g' /etc/ssh/sshd_config
systemctl restart sshd
```

```
## 禁止使用密码 ssh 登录
cat /etc/ssh/sshd_config |grep -E "PasswordAuthentication"
sed -i 's/PasswordAuthentication yes/PasswordAuthentication no/g' /etc/ssh/sshd_config
systemctl restart sshd
```

配置 SSH 免密登录:

```
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

阿里云镜像apt源:

```
mv /etc/apt/sources.list /etc/apt/sources.list.old
```

```
cat << 'EOF' > /etc/apt/sources.list
deb http://mirrors.aliyun.com/ubuntu/ xenial main
deb-src http://mirrors.aliyun.com/ubuntu/ xenial main
```

```
deb http://mirrors.aliyun.com/ubuntu/ xenial-updates main
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-updates main
```

```
deb http://mirrors.aliyun.com/ubuntu/ xenial universe
deb-src http://mirrors.aliyun.com/ubuntu/ xenial universe
deb http://mirrors.aliyun.com/ubuntu/ xenial-updates universe
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-updates universe
```

```
deb http://mirrors.aliyun.com/ubuntu/ xenial-security main
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-security main
deb http://mirrors.aliyun.com/ubuntu/ xenial-security universe
deb-src http://mirrors.aliyun.com/ubuntu/ xenial-security universe
EOF
```

```
## 更新源
```

```
apt-get clean all
apt-get update
```

注意： 不同Ubuntu版本使用 mirrors.aliyun.com 替换默认的 <http://archive.ubuntu.com/> 即可。

安装一些本地依赖及常用工具：

```
apt-get install gcc g++ make cmake libpcre3 libpcre3-dev libssl-dev wget \
libxslt1-dev libxslt1.1 build-essential autoconf libiconv-hook-dev libmcrypt-dev \
libxml2-dev libmysqlclient-dev build-essential zlib1g-dev \
libiconv-hook-dev libiconv-hook1 libjpeg8-dev libjpeg-turbo8-dev libjpeg-turbo8-dev libpng12-d
v libfreetype6-dev libgeopip-dev mysql-client mysql-common libncurses5 libncurses5-dev \
libcurl4-gnutls-dev curl libbz2-dev libicu-dev zlib1g-dev geopip-bin icu-devtools libcurl3-gnutl
libcurl3:amd64 libmagick++-dev unzip lrzsz
```

升级 openssl for ubuntu16.04 支持编译 Tengine2 以及 Nginx 时需要：

```
echo "deb http://archive.debian.org/debian jessie-backports main" > /etc/apt/sources.list.d/j
ssie-backports.list && \
apt-get -o Acquire::Check-Valid-Until=false update && \
apt-get -t jessie-backports install openssl libssl-dev && \
openssl version
```

Linux 内核优化参数：

//Linux实例出现间歇性丢包，无法连接实例

//内核日志：Feb 6 16:05:07 i-*** kernel: nf_conntrack: table full, dropping packet.

```
net.netfilter.nf_conntrack_max = 655350
```

```
net.netfilter.nf_conntrack_tcp_timeout_established = 1200
```

//报“TCP: time wait bucket table overflow”错误，大量处于TIME_WAIT状态的连接

```
//net.ipv4.tcp_tw_recycle = 1
```

```
net.ipv4.tcp_syncookies = 1 //是否开启SYN Cookies。当出现SYN等待队列溢出时，启用co
okies来处理，可防范少量SYN攻击
```

```
net.ipv4.tcp_fin_timeout = 30 //对于本端断开的Socket连接，TCP保持在FIN-WAIT-2状态的
间（秒）。
```

```
net.ipv4.tcp_max_syn_backlog = 8192 //表示SYN队列的长度，默认为1024，加大队列长度为819
，可以容纳更多等待连接的网络连接数
```

```
net.ipv4.tcp_max_tw_buckets = 5000 //系统同时保持TIME_WAIT套接字的最大数量，如果超过
个数字，TIME_WAIT套接字将立刻被清除并打印警告信息。默认为180000
```

```
net.ipv4.tcp_tw_reuse = 1 //是否允许将TIME-WAIT sockets重新用于新的TCP连接。
```

```
net.ipv4.tcp_synack_retries = 2 //指明了处于SYN_RECV状态时重传SYN+ACK包的次数
```

```
net.ipv6.conf.all.disable_ipv6 = 1 //IPV6 相关配置
```

```
net.ipv6.conf.default.disable_ipv6 = 1
```

```
net.ipv6.conf.lo.disable_ipv6 = 1
```

//本地网络NAT环境通过SSH连接Linux实例，或者访问该Linux实例上的HTTP业务出现异常。Telne
测试会被reset

```
//cat /proc/sys/net/ipv4/tcp_tw_recycle
```

```
//cat /proc/sys/net/ipv4/tcp_timestamps //默认为1，tcp_tw_recycle 启用则激活
```

```
net.ipv4.tcp_tw_recycle=0
```

```
net.ipv4.tcp_timestamps=0
```

//临时修改

```
/sbin/sysctl -w kernel.parameter="[${Example}]"
```

注: [\$Example]为参数值, 如sysctl -w net.ipv4.tcp_tw_recycle="0"命令, 将参数值改为0。

```
// 支持 ip_forward  
net.ipv4.ip_forward = 1
```

```
//永久生效  
vi /etc/sysctl.conf  
sysctl -p
```


二、CentOS 7.5系统初始化

- 修改主机名
- 关闭防火墙/selinux
- 网络配置
- 时间同步
- 初始化磁盘
- 配置ulimit
- 生成免密登录秘钥
- 配置 SSH
- 添加阿里云镜像源 (选择不同Linux发行版)
- 安装本地依赖库及常用工具 (可选)
- Linux 内核优化参数

修改主机名:

```
# Set hostname  
HOSTNAME=""  
hostnamectl set-hostname ${HOSTNAME}  
# hostname $HOSTNAME && echo "$HOSTNAME" > /etc/hostname
```

```
# Binding Hosts  
cat << EOF >> /etc/hosts  
## SET CEPH HOSTS  
{IP} {HOSTNAME}  
EOF
```

关闭 防火墙 和 selinux:

```
systemctl stop firewalld  
systemctl disable firewalld  
systemctl disable --now dnsmasq
```

```
setenforce 0 ## 临时关闭 selinux  
sed -ri '/^[^#]*SELINUX=/s#=.+$#=#disabled#' /etc/selinux/config ## 永久 selinux 关闭需要 re  
oot  
getenforce
```

网络配置:

修改系统默认网卡为eth0

```
cd /etc/sysconfig/network-scripts/ && mv ifcfg-eno16777736 ifcfg-eth0
```

```
vi /etc/sysconfig/grub
```

```
# GRUB_CMDLINE_LINUX 变量中最后添加 net.ifnames=0 biosdevname=0 (注意引号)  
grub2-mkconfig -o /boot/grub2/grub.cfg
```

```
# 添加udev的规则(很有必要!)
```

```
cat << 'EOF' > /etc/udev/rules.d/70-persistent-net.rules
```

```
SUBSYSTEM=="net",ACTION=="add",DRIVERS=="?*",ATTR{address}=="您的网卡MAC地址",A  
TR {type} == "1",KERNEL=="eth*",NAME="eth0"
```

```
# reboot
```

配置网卡:

```
## Stop NetworkManager
```

```
systemctl stop NetworkManager
```

```
systemctl disable NetworkManager
```

```
## Config Network
```

```
vim /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
-----  
[root@localhost]# vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
NAME=eth0
```

```
DEVICE=eth0
```

```
TYPE=Ethernet
```

```
ONBOOT=yes
```

```
BOOTPROTO=none
```

```
DEFROUTE=yes
```

```
IPADDR=192.168.1.200
```

```
#PREFIX=24
```

```
NETMASK=255.255.255.0
```

```
GATEWAY=192.168.1.1
```

```
IPV4_FAILURE_FATAL=no
```

```
DNS1=8.8.4.4
```

```
DNS2=8.8.8.8
```

```
IPV6INIT=no
```

```
IPV6_AUTOCONF=no
```

```
IPV6_DEFROUTE=no
```

```
IPV6_PEERDNS=no
```

```
IPV6_PEERROUTES=no
```

```
IPV6_PRIVACY=no
```

```
IPV6_FAILURE_FATAL=no
```

```
ARPCHECK=no
```

```
## Set DNS
```

```
vi /etc/resolv.conf
```

```
nameserver 8.8.4.4
```

```
nameserver 8.8.8.8
```

注意: 重启网络systemctl restart network 生效

时间同步:

```
## 设置时区
timedatectl status
timedatectl list-timezones | grep Shanghai
timedatectl set-timezone Asia/Shanghai ## ln -sf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
# timedatectl set-timezone UTC ## ln -sf /usr/share/zoneinfo/UTC /etc/localtime
# timedatectl set-time "YYYY-MM-DD HH:MM:SS"
# timedatectl set-time "HH:MM:SS"
timedatectl set-ntp yes
```

网络时间同步

```
yum install ntp ntpdate -y
# ntpdate cn.pool.ntp.org
ntpdate ntp1.aliyun.com &> /dev/null
echo "*/*5 * * * * /usr/sbin/ntpdate ntp1.aliyun.com &>/dev/null" >> /etc/crontab
hwclock --systohc
```

date

配置 ulimit:

```
## ulimit
ulimit -a
ulimit -n 65535
```

```
cat << EOF >> /etc/security/limits.conf
* soft nofile 65535
* hard nofile 65535
* soft nproc 65535
* hard nproc 65535
EOF
```

ulimit -n

生成免密登录秘钥:

```
ssh-keygen -t rsa -P ""
root@ubuntu: ~$ cat id_rsa.pub >>/root/.ssh/authorized_keys
```

配置 SSH (只监听IPv4端口, 关闭GSSAPI秘钥认证, 关闭DNS解析) :

```
sed -i s/'#ListenAddress 0.0.0.0'/'ListenAddress 0.0.0.0'/g /etc/ssh/sshd_config
grep ListenAddress /etc/ssh/sshd_config
```

```
sed -i s/'GSSAPIAuthentication yes'/'GSSAPIAuthentication no'/g /etc/ssh/sshd_config
grep GSSAPIAuthentication /etc/ssh/sshd_config
```

```
sed -i s/"^UseDNS yes"/"UseDNS no"/g /etc/ssh/sshd_config
service sshd restart
```

配置阿里云Yum源:

```
cd /etc/yum.repos.d/
```

```
mv CentOS-Base.repo CentOS-Base.repo.bak
curl -o /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo
curl -o /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-7.repo
mv /var/cache/yum /tmp
yum clean all && yum makecache
```

安装本地依赖库及常用工具:

```
yum install -y epel-release
```

```
yum install -y yum-utils device-mapper-persistent-data lvm2 net-tools \
contrack-tools wget vim ntpdate libseccomp libtool-ltdl vim wget \
openssl-devel ntpdate make gcc gcc-c++ cmake pcre pcre-devel \
zlib zlib-devel openssl ncurses-devel net-snmp sysstat lrzsz zip unzip \
tree net-tools lftp telnet iftop iotop
```

```
yum groupinstall -y "Development tools"
```

关闭 Swap (可选, 部署K8S需执行, 否则会报错) :

```
swapoff -a && sysctl -w vm.swappiness=0
sed -ri '/^[^#]*swap/s@^@#@' /etc/fstab
```

Linux 内核优化参数 (例如支持k8s安装) :

```
# 设置系统参数
cat << EOF > /etc/sysctl.d/k8s.conf
net.ipv4.ip_forward = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF
modprobe br_netfilter
sysctl -p /etc/sysctl.d/k8s.conf
```

--EOF--