**Linux Basic Environment Deployment**

Version： 1.0.0

Author : Jxva

**Change Records**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| File No | Version | Author/Modifier | Created/Modifed | Reason for change | Change history |
|  | 1.0.0 | Jxva | 2015-01-13 |  | Draft |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| Note 1:  You change the archive must fill in the table every time.  Note 2:  Both "reason for change" and "change content" need to fill in each archive. | | | | | |

**Table of Contents**

[**Table of Contents** 3](#_Toc409680084)

[1 Operate System Installation 4](#_Toc409680085)

[1.1 Disk Partition 4](#_Toc409680086)

[1.2 Network Configuration 4](#_Toc409680087)

[1.3 SSH Configuration 4](#_Toc409680088)

[1.4 System Initialization 5](#_Toc409680089)

[1.4.1 安装sudo及配置权限 5](#_Toc409680090)

[1.4.2 删除无用的帐号与群组 5](#_Toc409680091)

[1.4.3 修改登录欢迎语 6](#_Toc409680092)

[1.4.4 创建基本的用户组 6](#_Toc409680093)

[1.4.5 修改ulimit限制 7](#_Toc409680094)

[1.4.6 修改主机名 7](#_Toc409680095)

[1.4.7 禁用SELinux 7](#_Toc409680096)

[1.4.8 禁用不常用的服务 8](#_Toc409680097)

[1.5 Kernel Optimization 8](#_Toc409680098)

[1.6 Dependence Environment 10](#_Toc409680099)

[1.7 开发域名分配 10](#_Toc409680100)

[2 域名端口规划 11](#_Toc409680101)

[2.1 Colinux简介 11](#_Toc409680102)

[2.2 Fdsafdasfsa 11](#_Toc409680103)

[2.3 fdsafdas 11](#_Toc409680104)

# Operate System Installation

服务器操作系统为CentOS 64位服务器版，版本一般选用6.5或7.0，建议采用7.0版本。

CentOS官方网址为：<http://www.centos.org/>

通常下载的CentOS版本如下：

<http://mirrors.aliyun.com/centos/7/isos/x86_64/CentOS-7.0-1406-x86_64-Minimal.iso> (推荐)

<http://mirrors.aliyun.com/centos/7/isos/x86_64/CentOS-7.0-1406-x86_64-Everything.iso>

## Disk Partition

一般CentOS安装时，我们采用CentOS-7.0-1406-x86\_64-Minimal.iso 即可，我们可以将ISO利用UltraISO烧入U盘进行安装，也可以采用VMWare或VirtualBox进行虚拟化安装，在安装过程中我们采用默认方式安装即可，如英文、美式键盘等，需要特别注意的地方是磁盘的分区时，选择Standard Partition分区，分区大小及分配见下表：

|  |  |  |  |
| --- | --- | --- | --- |
| **分区名称** | **分区大小(M)** | **分区类型** | **备注** |
| / | 204800 | ext4 | 根分区 |
| /boot | 512 | ext4 | 启动分区，必须有，否则无法启动 |
| /swap | 与物理内存相当 | swap | 交换分区，当物理内存不够时，会启用交换分区 |
| /data | 所有剩余空间 | ext4 | 独立一个分区做为数据盘，当系统损坏时，确保数据不丢失 |

## Network Configuration

//TODO

|  |
| --- |
| #vi /etc/sysconfig/network-scripts/ifcfg-eth0  IPV6INIT="no"  IPV6\_AUTOCONF="no"  IPADDR=192.168.128.222  NETMASK=255.255.255.0  GATEWAY=192.168.128.1  #NETWORK=192.168.128.0  #BROADCAST=192.168.128.255 |

//TODO

## SSH Configuration

//TODO

|  |
| --- |
| #vi /etc/ssh/sshd\_config  IgnoreRhosts yes  UseDNS no  //然后重启sshd服务  //for centos6  #/etc/init.d/sshd restart  //for centos7  #systemctl restart sshd |

//TODO

## System Initialization

### 安装sudo及配置权限

|  |
| --- |
| #yum install sudo  #vi /etc/pam.d/su  auth required pam\_wheel.so use\_uid group=wheel  #visudo  %wheel ALL=(ALL) NOPASSWD: ALL  %www ALL=/bin/vi, /usr/bin/vim, /usr/bin/tail, /bin/more, /bin/ls, /bin/rm, /bin/mv, /bin/cp, /bin/cat  %mis ALL=/bin/vi, /usr/bin/vim, /usr/bin/fab, /usr/bin/tail, /bin/cat, /bin/more, /bin/rm, /bin/mv, /bin/ls, /bin/ping, /sbin/service, /bin/kill, /usr/bin/kill, /usr/bin/killall, /bin/chown, /bin/chmod, /bin/chgrp, /usr/bin/yum, /bin/mkdir, /bin/touch, /bin/cp, /bin/netstat, /usr/bin/python, /sbin/iptables |

上面设置了两个组：

* 一个wheel组(管理员)，具备sudo权限，可以提权到root权限
* 一个www组(网站维护组)，只能拥有固定的命令

通常还会设置一个dev组(开发组)，让开发人员只能在自己的home目录进行开发操作

### 删除无用的帐号与群组

为了安全起见，这里把不需要使用的帐号与群组全部删除

|  |
| --- |
| userdel adm  userdel lp  userdel sync  userdel shutdown  userdel halt  userdel news  userdel uucp  userdel operator  userdel games  userdel gopher  userdel ftp  groupdel adm  groupdel lp  groupdel news  groupdel uucp  groupdel games  groupdel dip  groupdel pppusers  groupdel audio  groupdel video |

### 修改登录欢迎语

修改欢迎语，提示登录用户不要乱操作服务器

|  |
| --- |
| #vi /etc/motd    Welcome to Jxva Develop Compute Service!  -----------------------------------------------------------  Please read the following words carefully.  1. Don't reboot this machine. (It can't wake up automicly)  2. Don't delete any others file.  3. Please be sure to use your own account.  4. Only members of the wheel group can run sudo.  ----------------------------------------------------------- |

### 创建基本的用户组

创建www组及用户，及常用的用户名

|  |
| --- |
| groupadd www  useradd -g www www -s /sbin/nologin -d /var/www  mkdir -p /data/home  useradd -g wheel jxva -s /bin/bash -d /data/home/jxva  usermod -G wheel,root jxva  useradd -g wheel jiangz -s /bin/bash -d /data/home/jiangz |

上面的命令大致解说如下：

1. 创建一个www用户及www组，同时确保www用户无登录权限，并将www用户锁定在/var/www目录；
2. 创建一个jxva帐号，并加入wheel与root组，默认用户目录为/data/home/jxva；
3. 创建一个jiangz帐号，并加入wheel组，默认用户目录为/data/home/jiangz；

### 修改ulimit限制

|  |
| --- |
| //查看系统最大的文件打开数  #cat /proc/sys/fs/file-max  #vi /etc/security/limits.conf  \* soft nofile 204800  \* hard nofile 204800  \* soft nproc 65535  \* hard nproc 65535 |

或者使用如下命令亦可

|  |
| --- |
| sudo sh -c 'echo "\* soft nofile 204800" >> /etc/security/limits.conf'  sudo sh -c 'echo "\* hard nofile 204800" >> /etc/security/limits.conf'  //for centos6  sudo sed -i 's/4096/65535/' /etc/security/limits.d/20-nproc.conf  //for centos7  sudo sed -i 's/1024/65535/' /etc/security/limits.d/90-nproc.conf |

上面的命令设置系统单进程可以打开的文件数为204800个，系统可以开启65535个进程。

### 修改主机名

CentOS6下修改主机名

|  |
| --- |
| # vi /etc/hosts  //在127.0.0.1与::1后面增加主机名  # vi /etc/sysconfig/network  //修改主机名(HOSTNAME=)  //输入以下命令：  hostname 新主机名 |

CentOS7下修改主机名

CentOS 7 里面修改hostname的方式有所改变，修改/etc/hosts和/etc/sysconfig/network两个文件已经不能生效。使用的新命令：

|  |
| --- |
| $ hostnamectl set-hostname <hostname> |

主机名修改之后，然后用ssh客户端重新登录，就会显示新的主机名。

### 禁用SELinux

//TODO

|  |
| --- |
| #vi /etc/selinux/config  SELINUX=disabled  #setenforce 0 //临时生效  #getenforce //查看selinux状态 |

//TODO

### 禁用不常用的服务

//TODO

|  |
| --- |
| 停止postfix服务  systemctl disable postfix  systemctl stop postfix  关闭防火墙  service firewalld stop  禁用防火墙  systemctl disable firewalld |

//TODO

## Kernel Optimization

//TODO

编辑/etc/sysctl.conf内容如下：

|  |
| --- |
| kernel.sysrq = 0  kernel.core\_uses\_pid = 1  kernel.msgmnb = 65536  kernel.msgmax = 65536  kernel.shmmax = 68719476736  kernel.shmall = 4294967296  net.core.somaxconn = 511  net.core.netdev\_max\_backlog = 81920  net.ipv4.ip\_forward = 0  net.ipv4.conf.all.rp\_filter = 0  net.ipv4.conf.default.rp\_filter = 0  net.ipv4.conf.default.accept\_source\_route = 0  net.ipv4.neigh.default.gc\_stale\_time = 120  net.ipv4.conf.default.arp\_announce = 2  net.ipv4.conf.all.arp\_announce = 2  net.ipv4.conf.lo.arp\_announce = 2  net.ipv4.ip\_local\_port\_range = 1024 65000  net.ipv4.tcp\_syncookies = 1  net.ipv4.tcp\_keepalive\_time = 300  net.ipv4.tcp\_fin\_timeout = 30  net.ipv4.tcp\_timestamps = 1  net.ipv4.tcp\_tw\_reuse = 1  net.ipv4.tcp\_tw\_recycle = 1  #net.ipv4.tcp\_tw\_timeout = 60  net.ipv4.tcp\_max\_tw\_buckets = 5000  net.ipv4.tcp\_max\_syn\_backlog = 1024  net.ipv4.tcp\_synack\_retries = 2  net.ipv4.tcp\_max\_orphans = 131072  net.ipv4.tcp\_no\_metrics\_save = 1  net.ipv4.tcp\_sack = 1  net.ipv4.tcp\_window\_scaling = 1  net.ipv6.conf.all.disable\_ipv6 = 1  net.ipv6.conf.default.disable\_ipv6 = 1  net.ipv6.conf.lo.disable\_ipv6 = 1  vm.overcommit\_memory = 1  vm.swappiness = 0  #net.ipv4.tcp\_orphan\_retries = 3  #cat /proc/sys/net/ipv4/tcp\_tw\_timeout  #cat /proc/sys/net/ipv4/tcp\_timestamps |

#cat /proc/sys/net/ipv4/tcp\_tw\_timeout

#cat /proc/sys/net/ipv4/tcp\_timestamps

//

|  |
| --- |
| #!/bin/sh  #history  #export HISTTIMEFORMAT="[%Y.%m.%d %H:%M:%S]"  export HISTTIMEFORMAT="`whoami` %F %T "  USER\_IP=`who -u am i 2>/dev/null| awk '{print $NF}'|sed -e 's/[()]//g'`  HISTDIR=/var/log/history  if [ -z $USER\_IP ]  then  USER\_IP=`hostname`  fi  if [ ! -d $HISTDIR ]  then  sudo mkdir -p $HISTDIR  sudo chmod 777 $HISTDIR  fi  if [ ! -d $HISTDIR/${LOGNAME} ]  then  mkdir -p $HISTDIR/${LOGNAME}  chmod 300 $HISTDIR/${LOGNAME}  fi  export HISTSIZE=4096  DT=`date +%Y%m%d\_%H%M%S`  #export HISTFILE="$HISTDIR/${LOGNAME}/${USER\_IP}\_history.$DT"  export HISTFILE="$HISTDIR/${LOGNAME}/${USER\_IP}\_history.log"  chmod 600 $HISTDIR/${LOGNAME}/\*\_history\* 2>/dev/null |

## Dependence Environment

//TODO

## Data Directories

/data/backup

/data/database

/data/software

/data/program

/data/wwwroot

/data/ftproot

/data/workspace

/data/document

/data/project/git

/data/project/docker

/data/project/gerrit

# 域名端口规划

//TODO

## Colinux简介

//TODO

## Fdsafdasfsa

//TODO

|  |
| --- |
| #include <stdio.h>  int main(void) {  return 0;  } |

## fdsafdas

//TODO

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

//TODO