## 修改主机名

vi /etc/sysconfig/network

例如 修改为centos6 然后reboot 或者 通过hostname命令进行临时指定



## 配置静态IP

### 打开文件

vi /etc/sysconfig/network-scripts/ifcfg-eth0

修改以下内容并保存

DEVICE=eth0 #对应第一张网卡

TYPE=Ethernet

ONBOOT=yes #是否启动时运行

NM\_CONTROLLED=yes

BOOTPROTO=static　　 　 #使用静态IP，而不是由DHCP分配IP

DEFROUTE=yes

IPV4\_FAILURE\_FATAL=yes

IPV6INIT=no

NAME="System eth0" #名称

HWADDR=00:50:56:94:04:3C  #必须对应etho是的MAC地址（/etc/udev/rules.d/70-persistent-net.rules）  ，此项可以删掉

PEERDNS=yes

PEERROUTES=yes

IPADDR=192.168.1.128　　 #指定本机IP地址

NETMASK=255.255.255.0  #指定子网掩码

GATEWAY=192.168.1.2 #指定网关   子网掩码跟网关可以在VMware设置中找到

DNS1=192.168.1.2

### 修改DNS

修改DNS

打开文件

vi /etc/resolv.conf

修改以下内容并保存

nameserver 8.8.8.8 #GOOGLE的DNS服务器

nameserver 192.168.1.2　 #指定经路由器上指定的DNS服务器

配置完成后，执行以下命令重启网络：

service network restart

## 关闭图形界面

执行 init 3 命令可立即关闭图形界面，执行init 5 或者start x 可以立即开启图形界面

设置开机时不进入图形界面，需要修改/etc/inittab

vi /etc/inittab

将 id:5:initdefault:

改成

id:3:initdefault:

## 关闭防火墙

service iptables stop （临时关闭）

chkconfig iptables off （重启后生效）

## 关闭SElinux

setenforce 0 （临时生效）

修改 /etc/selinux/config 中的 SELINUX=disabled （重启后永久生效）

## 配置NTP服务

集群中所有主机必须保持时间同步，如果时间相差较大会引起各种问题。 具体思路如下：

master节点作为ntp服务器与外界对时中心同步时间。

这里选择cdr-yl-01节点为时钟同步服务器，其他节点为客户端同步时间到该节点。安装ntp:

yum -y install ntp

修改cdr-yl-01上的配置文件 /etc/ntp.conf :

restrict default ignore   //默认不允许修改或者查询ntp,并且不接收特殊封包

restrict 127.0.0.1        //给于本机所有权限

restrict 192.168.1.0 mask 255.255.255.0 notrap nomodify  //给于局域网机的机器有同步时间的权限

server  192.168.1.106     # local clock

driftfile /var/lib/ntp/drift

fudge   127.127.1.0 stratum 10

启动ntp

chkconfig ntpd on

service ntpd start

在cdr-yl-02、cdr-yl-03和cdr-yl-04节点上执行下面操作

ntpdate cdr-yl-01

Ntpd启动的时候通常需要一段时间大概5分钟进行时间同步，所以在ntpd刚刚启动的时候还不能正常提供时钟服务，报错”no server suitable for synchronization found”。启动时候需要等待5分钟。

如果想定时进行时间校准，可以使用crond服务来定时执行。

# 每天 1:00 Linux 系统就会自动的进行网络时间校准

00 1 \* \* \* root /usr/sbin/ntpdate 192.168.1.106 >> /root/ntpdate.log 2>&1

### [Linux NTP配置详解 (Network Time Protocol)](http://blog.csdn.net/iloli/article/details/6431757)

Network Time Protocol (NTP) 也是RHCE新增的考试要求. 学习的时候也顺便复习了一下如何设置Linux的时间,现在拿出来和大家分享

设置NTP服务器不难但是NTP本身是一个很复杂的协议. 这里只是简要地介绍一下实践方法  
和上次一样,下面的实验都在RHEL5上运行

**1. 时间和时区**

如果有人问你说现在几点? 你看了看表回答他说晚上8点了. 这样回答看上去没有什么问题,但是如果问你的这个人在欧洲的话那么你的回答就会让他很疑惑,因为他那里还太阳当空呢.

这里就有产生了一个如何定义时间的问题. 因为在地球环绕太阳旋转的24个小时中,世界各地日出日落的时间是不一样的.所以我们才有划分时区(timezone) 的必要,也就是把全球划分成24个不同的时区. 所以我们可以把时间的定义理解为一个时间的值加上所在地的时区(注意这个所在地可以精确到城市)

地理课上我们都学过格林威治时间(GMT), 它也就是0时区时间. 但是我们在计算机中经常看到的是UTC. 它是Coordinated Universal Time的简写. 虽然可以认为UTC和GMT的值相等(误差相当之小),但是UTC已经被认定为是国际标准,所以我们都应该遵守标准只使用UTC

那么假如现在中国当地的时间是晚上8点的话,我们可以有下面两种表示方式  
20:00 CST  
12:00 UTC

这里的CST是Chinese Standard Time,也就是我们通常所说的北京时间了. 因为中国处在UTC+8时区,依次类推那么也就是12:00 UTC了.

为什么要说这些呢(呵呵这里不是地理论坛吧...)

第一,不管通过任何渠道我们想要同步系统的时间,通常提供方只会给出UTC+0的时间值而不会提供时区(因为它不知道你在哪里).所以当我们设置系统时间的时候,设置好时区是首先要做的工作  
第二,很多国家都有夏令时(我记得小时候中国也实行过一次),那就是在一年当中的某一天时钟拨快一小时(比如从UTC+8一下变成UTC+9了),那么同理到时候还要再拨慢回来.如果我们设置了正确的时区,当需要改变时间的时候系统就会自动替我们调整

现在我们就来看一下如何在Linux下设置时区,也就是time zone

**2. 如何设置Linux Time Zone**

在Linux下glibc提供了我们事先编译好的许多timezone文件, 他们就放在/usr/share/zoneinfo这个目录下,这里基本涵盖了大部分的国家和城市  
ls -F /usr/share/zoneinfo/

Africa/      Chile/   Factory    Iceland      Mexico/   posix/      Universal  
America/     CST6CDT  GB         Indian/      Mideast/  posixrules  US/  
Antarctica/  Cuba     GB-Eire    Iran         MST       PRC         UTC  
Arctic/      EET      GMT        iso3166.tab  MST7MDT   PST8PDT     WET  
Asia/        Egypt    GMT0       Israel       Navajo    right/      W-SU  
Atlantic/    Eire     GMT-0      Jamaica      NZ        ROC         zone.tab  
Australia/   EST      GMT+0      Japan        NZ-CHAT   ROK         Zulu  
Brazil/      EST5EDT  Greenwich  Kwajalein    Pacific/  Singapore  
Canada/      Etc/     Hongkong   Libya        Poland    Turkey  
CET          Europe/  HST        MET          Portugal  UCT在这里面我们就可以找到自己所在城市的time zone文件. 那么如果我们想查看对于每个time zone当前的时间我们可以用zdump命令  
代码:  
 zdump Hongkong

Hongkong  Fri Jul  6 06:13:57 2007 HKT那么我们又怎么来告诉系统我们所在time zone是哪个呢? 方法有很多,这里举出两种

第一个就是修改/etc/localtime这个文件,这个文件定义了我么所在的local time zone.  
我们可以在/usr/share/zoneinfo下找到我们的time zone文件然后拷贝去到/etc/localtimezone(或者做个symbolic link)

假设我们现在的time zone是BST(也就是英国的夏令时间,UTC+1)  
代码:  
# date  
Thu Jul  5 23:33:40 BST 2007我们想把time zone换成上海所在的时区就可以这么做  
代码:  
ln -sf /usr/share/zoneinfo/posix/Asia/Shanghai /etc/localtime  
date

Fri Jul  6 06:35:52 CST 2007  
这样时区就改过来了(注意时间也做了相应的调整)

第二种方法也就设置TZ环境变量的值. 许多程序和命令都会用到这个变量的值. TZ的值可以有多种格式,最简单的设置方法就是使用tzselect命令  
代码:  
# tzselect  
...  
TZ='America/Los\_Angeles';export TZtzselect  
会让你选择所在的国家和城市(我省略了这些步骤),最后输出相应的TZ变量的值.那么如果你设置了TZ的值之后时区就又会发生变化

代码:  
# date  
Thu Jul  5 15:48:11 PDT 2007  
通过这两个例子我们也可以发现TZ变量的值会override /etc/localtime. 也就是说当TZ变量没有定义的时候系统才使用/etc/localtime来确定time zone. 所以你想永久修改time zone的话那么可以把TZ变量的设置写入/etc/profile里

好了现在我们知道怎么设置时区了,下面我们就来看看如何设置Linux的时间吧

**3. Real Time Clock(RTC) and System Clock**

说道设置时间这里还要明确另外一个概念就是在一台计算机上我们有两个时钟:一个称之为硬件时间时钟(RTC),还有一个称之为系统时钟(System Clock)

硬件时钟是指嵌在主板上的特殊的电路, 它的存在就是平时我们关机之后还可以计算时间的原因  
系统时钟就是操作系统的kernel所用来计算时间的时钟. 它从1970年1月1日00:00:00 UTC时间到目前为止秒数总和的值 在Linux下系统时间在开机的时候会和硬件时间同步(synchronization),之后也就各自独立运行了

那么既然两个时钟独自运行,那么时间久了必然就会产生误差了,下面我们来看一个例子  
代码:  
# date  
Fri Jul  6 00:27:13 BST 2007  
# hwclock --show  
Fri 06 Jul 2007 12:27:17 AM BST  -0.968931 seconds   
通过hwclock --show 命令我们可以查看机器上的硬件时间(always in local time zone), 我们可以看到它和系统时间还是有一定的误差的, 那么我们就需要把他们同步

如果我们想要把硬件时间设置成系统时间我们可以运行以下命令  
代码:  
# hwclock --hctosys    
反之,我们也可以把系统时间设置成硬件时间  
代码:  
# hwclock --systohc    
那么如果想设置硬件时间我们可以开机的时候在BIOS里设定.也可以用hwclock命令  
代码:  
# hwclock --set --date="mm/dd/yy hh:mm:ss"    
如果想要修改系统时间那么用date命令就最简单了  
代码:  
# date -s "dd/mm/yyyy hh:mm:ss"

现在我们知道了如何设置系统和硬件的时间. 但问题是如果这两个时间都不准确了怎么办? 那么我们就需要在互联网上找到一个可以提供我们准确时间的服务器然后通过一种协议来同步我们的系统时间,那么这个协议就是NTP了. 注意接下去我们所要说的同步就都是指系统时间和网络服务器之间的同步了

**4. 设置NTP Server前的准备**

其实这个标题应该改为设置"NTP Relay Server"前的准备更加合适. 因为不论我们的计算机配置多好运行时间久了都会产生误差,所以不足以给互联网上的其他服务器做NTP Server. 真正能够精确地测算时间的还是原子钟. 但由于原子钟十分的昂贵,只有少部分组织拥有, 他们连接到计算机之后就成了一台真正的NTP Server. 而我们所要做的就是连接到这些服务器上同步我们系统的时间,然后把我们自己的服务器做成NTP Relay Server再给互联网或者是局域网内的用户提供同步服务

好了,前面讲了一大堆理论,现在我们来动手实践一下吧. 架设一个NTP Relay Server其实非常简单,我们先把需要的RPM包装上  
是否已经安装了NTP包可以用这条命令来确定：

[root@NTPser ~]# rpm -qa | grep ntp  
ntp-4.2.2p1-9.el5\_4.1  
chkfontpath-1.10.1-1.1  
出现以上代码则表示已安装NTP包，否则用下面方法安装：  
代码:  
  
# rpm -ivh ntp-4.2.2p1-5.el5.rpm   
那么第一步我们就要找到在互联网上给我们提供同步服务的NTP Server

[http://www.pool.ntp.org](http://www.pool.ntp.org/)是NTP的官方网站,在这上面我们可以找到离我们城市最近的NTP Server. NTP建议我们为了保障时间的准确性,最少找两个个NTP Server  
那么比如在英国的话就可以选择下面两个服务器  
0.uk.pool.ntp.org  
1.uk.pool.ntp.org

它的一般格式都是number.country.pool.ntp.org

第二步要做的就是在打开NTP服务器之前先和这些服务器做一个同步,使得我们机器的时间尽量接近标准时间.   
这里我们可以用ntpdate命令手动更新时间  
代码:  
# ntpdate 0.uk.pool.ntp.org  
 6 Jul 01:21:49 ntpdate[4528]: step time server 213.222.193.35 offset -38908.575181 sec  
# ntpdate 0.pool.ntp.org  
 6 Jul 01:21:56 ntpdate[4530]: adjust time server 213.222.193.35 offset -0.000065 sec  
假如你的时间差的很离谱的话第一次会看到调整的幅度比较大,所以保险起见可以运行两次. 那么为什么在打开NTP服务之前先要手动运行同步呢?  
1. 因为根据NTP的设置,如果你的系统时间比正确时间要快的话那么NTP是不会帮你调整的,所以要么你把时间设置回去,要么先做一个手动同步  
2. 当你的时间设置和NTP服务器的时间相差很大的时候,NTP会花上较长一段时间进行调整.所以手动同步可以减少这段时间

5. 配置和运行NTP Server

现在我们就来创建NTP的配置文件了, 它就是**/etc/ntp.conf**. 我们只需要加入上面的NTP Server和一个driftfile就可以了  
代码:  
# vi /etc/ntp.conf

server 210.72.145.44     #这是中国国家授时中心的IP  
server 0.uk.pool.ntp.org  
server 1.uk.pool.ntp.org  
                                       
fudge 127.127.1.0 stratum 0  stratum  这行是时间服务器的层次。设为0则为顶级，如果要向别的NTP服务器更新时间，请不要把它设为0  
  
**driftfile /var/lib/ntp/ntp.drift**  非常的简单. 接下来我们就启动NTP Server,并且设置其在开机后自动运行  
代码:  
# /etc/init.d/ntpd start  
# chkconfig --level 35 ntpd on

**6. 查看NTP服务的运行状况**

现在我们已经启动了NTP的服务,但是我们的系统时间到底和服务器同步了没有呢? 为此NTP提供了一个很好的查看工具: ntpq (NTP query)

我建议大家在打开NTP服务器后就可以运行ntpq命令来监测服务器的运行.这里我们可以使用watch命令来查看一段时间内服务器各项数值的变化  
代码:  
# watch ntpq -p  
Every 2.0s: ntpq -p                                  Sat Jul  7 00:41:45 2007

     remote           refid      st t when poll reach   delay   offset  jitter  
===========================================================  
+193.60.199.75   193.62.22.98     2 u   52   64  377    8.578   10.203 289.032  
\*mozart.musicbox 192.5.41.41      2 u   54   64  377   19.301  -60.218 292.411

现在我就来解释一下其中的含义

  remote: 它指的就是本地机器所连接的远程NTP服务器  
      refid: 它指的是给远程服务器(e.g. 193.60.199.75)提供时间同步的服务器  
          st: 远程服务器的层级别（stratum）. 由于NTP是层型结构,有顶端的服务器,多层的Relay Server再到客户端. 所以服务器从高到低级别可以设定为1-16. 为了减缓负荷和网络堵塞,原则上应该避免直接连接到级别为1的服务器的.  
            t: 这个.....我也不知道啥意思^\_^  
    when: 我个人把它理解为一个计时器用来告诉我们还有多久本地机器就需要和远程服务器进行一次时间同步  
       poll: 本地机和远程服务器多少时间进行一次同步(单位为秒). 在一开始运行NTP的时候这个poll值会比较小,那样和服务器同步的频率也就增加了,可以尽快调整到正确的时间范围.之后poll值会逐渐增大,同步的频率也就会相应减小  
    reach: 这是一个八进制值,用来测试能否和服务器连接.每成功连接一次它的值就会增加  
    delay: 从本地机发送同步要求到服务器的round trip time  
    offset: 这是个最关键的值, 它告诉了我们本地机和服务器之间的时间差别. offset越接近于0,我们就和服务器的时间越接近  
     jitter: 这是一个用来做统计的值. 它统计了在特定个连续的连接数里offset的分布情况. 简单地说这个数值的绝对值越小我们和服务器的时间就越精确

那么大家细心的话就会发现两个问题: 第一我们连接的是0.uk.pool.ntp.org为什么和remote server不一样? 第二那个最前面的+和\*都是什么意思呢?

第一个问题不难理解,因为NTP提供给我们的是一个cluster server所以每次连接的得到的服务器都有可能是不一样.同样这也告诉我们了在指定NTP Server的时候应该使用hostname而不是IP

第二个问题和第一个相关,既然有这么多的服务器就是为了在发生问题的时候其他的服务器还可以正常地给我们提供服务.那么如何知道这些服务器的状态呢? 这就是第一个记号会告诉我们的信息

\* 它告诉我们远端的服务器已经被确认为我们的主NTP Server,我们系统的时间将由这台机器所提供  
+ 它将作为辅助的NTP Server和带有\*号的服务器一起为我们提供同步服务. 当\*号服务器不可用时它就可以接管  
－ 远程服务器被clustering algorithm认为是不合格的NTP Server  
x 远程服务器不可用

了解这些之后我们就可以实时监测我们系统的时间同步状况了

**7. NTP安全设置**

运行一个NTP Server不需要占用很多的系统资源,所以也不用专门配置独立的服务器,就可以给许多client提供时间同步服务, 但是一些基本的安全设置还是很有必要的  
那么这里一个很简单的思路就是第一我们只允许局域网内一部分的用户连接到我们的服务器. 第二个就是这些client不能修改我们服务器上的时间

关于权限设定部分   
权限的设定主要以 restrict 这个参数来设定，主要的语法为：   
restrict IP地址 mask 子网掩码 参数   
其中 IP 可以是IP地址，也可以是 default ，default 就是指所有的IP   
参数有以下几个：   
ignore　：关闭所有的 NTP 联机服务   
nomodify：客户端不能更改服务端的时间参数，但是客户端可以通过服务端进行网络校时。   
notrust ：客户端除非通过认证，否则该客户端来源将被视为不信任子网   
noquery ：不提供客户端的时间查询   
注意：如果参数没有设定，那就表示该 IP (或子网)没有任何限制！

在/etc/ntp.conf文件中我们可以用restrict关键字来配置上面的要求

首先我们对于默认的client拒绝所有的操作  
代码:  
restrict default kod nomodify notrap nopeer noquery

然后允许本机地址一切的操作  
代码:  
restrict 127.0.0.1

最后我们允许局域网内所有client连接到这台服务器同步时间.但是拒绝让他们修改服务器上的时间  
代码:  
restrict 192.168.1.0 mask 255.255.255.0 nomodify

把这三条加入到/etc/ntp.conf中就完成了我们的简单配置. NTP还可以用key来做authentication,这里就不详细介绍了

**8. NTP client的设置**

做到这里我们已经有了一台自己的Relay Server.如果我们想让局域网内的其他client都进行时间同步的话那么我们就都应该照样再搭建一台Relay Server,然后把所有的client都指向这两台服务器(注意不要把所有的client都指向Internet上的服务器). 只要在client的**/etc/ntp.conf**加上这你自己的服务器就可以了  
代码:  
server ntp1.leonard.com  
server ntp2.leonard.com

LINUX客户端使用  
ntpdate 172.30.218.114   
来向NTP服务器同步自己的时间  
其它LINUX如果仅作为只客户端的话,则不能启动ntpd服务!否则无法运行ntpdata 服务器地址 来同步时间  
之后可以使用cron或修改crontab文件定期向NTP服务器更新时间,并用  
# hwclock --systohc    
将系统时间设置为硬件时间

**9. 一些补充和拾遗（挺重要）**

1. 配置文件中的driftfile是什么?  
我们每一个system clock的频率都有小小的误差,这个就是为什么机器运行一段时间后会不精确. NTP会自动来监测我们时钟的误差值并予以调整.但问题是这是一个冗长的过程,所以它会把记录下来的误差先写入driftfile.这样即使你重新开机以后之前的计算结果也就不会丢失了

2. 如何同步硬件时钟?  
NTP一般只会同步system clock. 但是如果我们也要同步RTC(hwclock)的话那么只需要把下面的选项打开就可以了  
代码:  
# vi /etc/sysconfig/ntpd  
SYNC\_HWCLOCK=yes

3、利用crontab让LINUX NTP定时更新时间  
注：让linux运行ntpdate更新时间时，linux不能开启NTP服务，否则会提示端口被占用：如下  
[root@ESXI ~]# ntpdate 1.rhel.pool.ntp.org                                   
20 May 09:34:14 ntpdate[6747]: the NTP socket is in use, exiting  
  
crontab文件配置简要说明  
命令格式的前一部分是对时间的设定，后面一部分是要执行的命令。时间的设定我们有一定的约定，前面五个\*号代表五个数字，数字的取值范围和含义如下：  
分钟　(0-59)  
小時　(0-23)  
日期　(1-31)  
月份　(1-12)  
星期　(0-6)//0代表星期天  
除了数字还有几个个特殊的符号就是“\*”、“/”和“-”、“,”，“\*”代表所有的取值范围内的数字，“/”代表每的意思,“\*/5”表示每5个单位，“-”代表从某个数字到某个数字,“,”分开几个离散的数字。以下举几个例子说明问题：  
每天早上6点：  
0 6 \* \* \*  command  
每两个小时：  
0 \*/2 \* \* \*  command  
晚上11点到早上8点之间每两个小时，早上八点：  
0 23-7/2,8 \* \* \* command  
每个月的4号和每个礼拜的礼拜一到礼拜三的早上11点：  
0 11 4 \* 1-3 command   
1月1日早上4点：  
0 4 1 1 \* command

3.3、设置开机自动启动服务  
运行setup或其它服务设置工具，将crond服务勾选上   
chkconfig --level 2345 crond on  定义在这几个系统运行级别上启用crond (系统安装完默认就是这个设置)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**10.NTP客户端的设置**

一、LINUX做为客户端自动同步时间  
如果想定时进行时间校准，可以使用crond服务来定时执行。  
编辑 /etc/crontab 文件  
加入下面一行：

30 8 \* \* \* root /usr/sbin/ntpdate 192.168.0.1; /sbin/hwclock -w  #192.168.0.1是NTP服务器的IP地址  
然后重启crond服务  
service crond restart   
这样，每天 8:30 Linux 系统就会自动的进行网络时间校准。

二、WINDOWS 需要打开windows time服务和RPC的二个服务  
如果在打开windows time 服务，时报 错误1058，进行下面操作   
1.运行 cmd 进入命令行，然后键入  
w32tm /register  进行注册  
正确的响应为：W32Time 成功注册。

2.如果上一步正确，用 net start "windows time" 或 net start w32time 启动服务。

**11.其它造成无法成功更新的原因：**1、客户端的日期必须要设置正确，不能超出正常时间24小时，不然会因为安全原因被拒绝更新。其次客户端的时区必须要设置好，以确保不会更新成其它时区的时间。  
2、fudge 127.127.1.0 stratum 10 如果是LINUX做为NTP服务器，stratum(层级)的值不能太大，如果要向上级NTP更新可以设成2  
3、LINUX的NTP服务器必须记得将从上级NTP更新的时间从系统时间写到硬件里去 hwclock --systohc   
     NTP一般只会同步system clock. 但是如果我们也要同步RTC(hwclock)的话那么只需要把下面的选项打开就可以了  
      代码:  
      # vi /etc/sysconfig/ntpd  
      SYNC\_HWCLOCK=yes  
4、Linux如果开启了NTP服务，则不能手动运行ntpdate更新时间（会报端口被占用），它只能根据/etc/ntp.conf 里server 字段后的服务器地址按一定时间间隔自动向上级NTP服务器更新时间。可以运行命令 ntpstat 查看每次更新间隔如：  
[root@ESXI ~]# ntpstat  
synchronised to NTP server (210.72.145.44) at stratum 2    #本NTP服务器层次为2，已向210.72.145.44 NTP同步过  
   time correct to within 93 ms                                               #时间校正到相差93ms之内  
   polling server every 1024 s                                                 #每1024秒会向上级NTP轮询更新一次时间

## 配置国内较好的yum源

网易（163）yum源是国内最好的yum源之一 ，无论是速度还是软件版本，都非常的不错，将yum源设置为163yum，可以提升软件包安装和更新的速度，同时避免一些常见软件版本无法找到。具体设置方法如下：

1. 进入yum源配置目录

cd /etc/yum.repos.d

1. 备份系统自带的yum源

mv CentOS-Base.repo CentOS-Base.repo.bk  
下载163网易的yum源：  
wget http://mirrors.163.com/.help/CentOS6-Base-163.repo

1. 更新玩yum源后，执行下边命令更新yum配置，使操作立即生效

yum makecache

1. 除了网易之外，国内还有其他不错的yum源，比如中科大和搜狐的，大家可以根据自己需求下载

中科大的yum源：  
wget <http://centos.ustc.edu.cn/CentOS-Base.repo>  
sohu的yum源  
wget <http://mirrors.sohu.com/help/CentOS-Base-sohu.repo>

理论上讲，这些yum源redhat系统以及fedora也是可以用 的，但是没有经过测试，需要的站长可以自己测试一下。

## 安装mysql

1. 查看操作系统相关信息。

[root@linuxidc ~]# cat /etc/issue

[CentOS](http://www.linuxidc.com/topicnews.aspx?tid=14) release 6.5 (Final)

Kernel \r on an \m

[root@linuxidc ~]# uname -a

Linux linuxidc 2.6.32-431.el6.x86\_64 #1 SMP Fri Nov 22 03:15:09 UTC 2013 x86\_64 x86\_64 x86\_64 GNU/Linux

1. 创建需要下载rpm软件包的目录(这个随意指定)。

[root@linuxidc ~]# mkdir -p /taokey/tools

1. 查看下是否有系统自带mysql的rpm包，如果有，需要删除自带的旧rpm包。

[root@linuxidc ~]# rpm -qa | grep mysql

mysql-libs-5.1.71-1.el6.x86\_64

[root@linuxidc ~]# yum -y remove mysql-libs-5.1\*

[root@linuxidc ~]# rpm -qa | grep mysql

[root@linuxidc ~]#

1. 在MySQL官网下载安装MySQL-5.6.21所需的rpm软件包。

需要下载三个rpm软件包：

MySQL-client-5.6.21-1.rhel5.x86\_64.rpm

MySQL-devel-5.6.21-1.rhel5.x86\_64.rpm

MySQL-server-5.6.21-1.rhel5.x86\_64.rpm

[root@linuxidc ~]# cd /taokey/tools/

[root@linuxidc tools]# wget \

http://dev.mysql.com/Downloads/MySQL-5.6/MySQL-server-5.6.21-1.rhel5.x86\_64.rpm

[root@linuxidc tools]# wget \

http://dev.mysql.com/Downloads/MySQL-5.6/MySQL-devel-5.6.21-1.rhel5.x86\_64.rpm

[root@linuxidc tools]# wget \

http://dev.mysql.com/Downloads/MySQL-5.6/MySQL-client-5.6.21-1.rhel5.x86\_64.rpm

1. 下载完之后，安装三个rpm软件包。

[root@linuxidc tools]# rpm -ivh MySQL-server-5.6.21-1.rhel5.x86\_64.rpm

error: Failed dependencies:

        libaio.so.1()(64bit) is needed by MySQL-server-5.6.21-1.rhel5.x86\_64

        libaio.so.1(LIBAIO\_0.1)(64bit) is needed by MySQL-server-5.6.21-1.rhel5.x86\_64

        libaio.so.1(LIBAIO\_0.4)(64bit) is needed by MySQL-server-5.6.21-1.rhel5.x86\_64

安装MySQL-server报错，原因是没有安装libaio，系统缺少libaio.so此软件包，下边yum安装一下libaio.so软件包。

[root@linuxidc tools]# yum install -y libaio

[root@linuxidc tools]# rpm -ivh MySQL-server-5.6.21-1.rhel5.x86\_64.rpm

[root@linuxidc tools]# rpm -ivh MySQL-client-5.6.21-1.rhel5.x86\_64.rpm

Preparing...                ########################################### [100%]

  1:MySQL-client          ########################################### [100%]

[root@linuxidc tools]# rpm -ivh MySQL-devel-5.6.21-1.rhel5.x86\_64.rpm

Preparing...                ########################################### [100%]

  1:MySQL-devel            ########################################### [100%]

1. 修改配置文件位置。

[root@linuxidc tools]# cp /usr/share/mysql/my-default.cnf /etc/my.cnf

1. 初始化MySQL及修改MySQL默认的root密码。

[root@linuxidc tools]# /usr/bin/mysql\_install\_db

[root@linuxidc tools]# service mysql start

[root@linuxidc tools]# ps -ef | grep mysql

root      2188    1  0 14:48 pts/1    00:00:00 /bin/sh /usr/bin/mysqld\_safe --datadir=/var/lib/mysql --pid-file=/var/lib/mysql/linuxidc.pid

mysql    2303  2188 30 14:48 pts/1    00:00:02 /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib64/mysql/plugin --user=mysql --log-error=/var/lib/mysql/linuxidc.err --pid-file=/var/lib/mysql/linuxidc.pid

root      2331  1853  0 14:49 pts/1    00:00:00 grep mysql

[root@linuxidc tools]# netstat -anpt | grep 3306

tcp        0      0 :::3306                    :::\*                  LISTEN      2303/mysqld

[root@linuxidc tools]# more /root/.mysql\_secret

# The random password set for the root user at Thu Apr  9 14:43:59 2015 (local time): F6K3v\_xggFoLQeiN

[root@linuxidc tools]# mysql -uroot -pF6K3v\_xggFoLQeiN

Warning: Using a password on the command line interface can be insecure.

Welcome to the MySQL monitor.  Commands end with ; or \g.

Your MySQL connection id is 1

Server version: 5.6.21

Copyright (c) 2000, 2014, [Oracle](http://www.linuxidc.com/topicnews.aspx?tid=12) and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective

owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SET PASSWORD = PASSWORD('root');

mysql> exit

Bye

[root@linuxidc tools]# mysql -uroot -proot

Warning: Using a password on the command line interface can be insecure.

Welcome to the MySQL monitor.  Commands end with ; or \g.

Your MySQL connection id is 6

Server version: 5.6.21 MySQL Community Server (GPL)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective

owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

1. 设置MySQL服务开机自启动。

[root@linuxidc tools]# chkconfig mysql on

[root@linuxidc tools]# chkconfig mysql --list

mysql          0:off  1:off  2:on    3:on    4:on    5:on    6:off

到此为止，使用RPM安装MySQL5.6版本数据库安装完毕

## 设置ssh无密码登陆（server端到所有节点）

在主节点上执行ssh-keygen -t rsa一路回车，生成无密码的密钥对。

用ssh-copy-id把公钥复制到远程主机上

例如cdr-yl-01主机到其它3节点设置无密码登录

ssh-copy-id -i ~/.ssh/id\_rsa.pub root@ cdr-yl-02

ssh-copy-id -i ~/.ssh/id\_rsa.pub root@ cdr-yl-03

ssh-copy-id -i ~/.ssh/id\_rsa.pub root@ cdr-yl-04