### 一、配置

#### 1. 所有服务器安装NTP服务

|  |
| --- |
| yum install ntp |

#### 配置NTP服务

服务端修改默认网段

|  |
| --- |
| vim /etc/ntp.conf |
| # restrict 127.0.0.1  # restrict ::1  restrict 10.8.1.0 mask 255.255.255.0 no modify no trip  #server 0.centos.pool.ntp.org iburst  #server 1.centos.pool.ntp.org iburst  #server 2.centos.pool.ntp.org iburst  #server 3.centos.pool.ntp.org iburst  server 127.127.1.0  fudge 127.127.1.0 stratum 10 |

客户端修改网段

|  |
| --- |
| vim /etc/ntp.conf |
| server 10.8.1.6 iburst |

### 二、注意事项

#### 1. 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 mask netmask\_IP parameter

其中 IP 可以是软件地址，也可以是 default ，default 就类似 0.0.0.0

#　　至于paramter则有：

#　　　ignore　：关闭所有的 NTP 联机服务

#　　　nomodify：表示 Client 端不能更改 Server 端的时间参数，不过，

#　　　Client 端仍然可以透过 Server 端来进行网络校时。

#　　　notrust：该 Client 除非通过认证，否则该 Client 来源将被视为不信任网域

#　　　noquery：不提供 Client 端的时间查询

#　　　notrap：不提供trap这个远程事件登入

#　　如果paramter完全没有设定，那就表示该 IP (或网域)“没有任何限制”

restrict default nomodifynotrapnoquery　# 关闭所有的 NTP 要求封包

restrict 127.0.0.1　　　 #这是允许本级查询

restrict 192.168.0.1 mask 255.255.255.0 nomodify

#在192.168.0.1/24网段内的服务器就可以通过这台NTP Server进行时间同步了

#### 2. 上层主机的设定

#　　要设定上层主机主要以 server 这个参数来设定，语法为：

#　　server [IP|HOST Name] [prefer]

#　　Server 后面接的就是我们上层 Time Server 啰！而如果 Server 参数

#　　后面加上perfer的话，那表示我们的 NTP 主机主要以该部主机来作为

#　　时间校正的对应。另外，为了解决更新时间封包的传送延迟动作，

#　　所以可以使用driftfile来规定我们的主机

#　　在与 Time Server 沟通时所花费的时间，可以记录在driftfile

#　　后面接的文件内，例如下面的范例中，我们的 NTP server 与

#　　cn.pool.ntp.org联机时所花费的时间会记录在 /etc/ntp/drift文件内

server 0.pool.ntp.org

server 1.pool.ntp.org

server 2.pool.ntp.org

server cn.pool.ntp.org prefer

#其他设置值，以系统默认值即可

server 127.127.1.0 # local clock

fudge 127.127.1.0 stratum 10

driftfile /var/lib/ntp/drift

broadcastdelay 0.008

keys /etc/ntp/keys

总结一下，restrict用来设置访问权限，server用来设置上层时间服务器，driftfile用来设置保存漂移时间的文件。

二、使用ntpdate比较简单。格式如下

[root@linux ~]# ntpdate [-nv] [NTP IP/hostname]

[root@linux ~]# ntpdate 192.168.0.2

[root@linux ~]# ntpdate time.ntp.org

但这样的同步，只是强制性的将系统时间设置为ntp服务器时间。如果cpu tick有问题，只是治标不治本。所以，一般配合cron命令，来进行定期同步设置。比如，在crontab中添加： root路径下

在/etc/crontab 文件下 添加

0 12 \* \* \* \* /usr/sbin/ntpdate 192.168.0.1

这样，会在每天的12点整，同步一次时间。ntp服务器为192.168.0.1。

三、两点之间的不同

使用ntpd服务，要好于ntpdate加cron的组合。因为，ntpdate同步时间，会造成时间的跳跃，对一些依赖时间的程序和服务会造成影响。比如sleep，timer等。而且，ntpd服务可以在修正时间的同时，修正cpu tick。理想的做法为，在开机的时候，使用ntpdate强制同步时间，在其他时候使用ntpd服务来同步时间。

#### 其他

要注意的是，ntpd有一个自我保护设置: 如果本机与上源时间相差太大, ntpd不运行. 所以新设置的时间服务器一定要先ntpdate从上源取得时间初值, 然后启动ntpd服务。ntpd服务运行后, 先是每64秒与上源服务器同步一次, 根据每次同步时测得的误差值经复杂计算逐步调整自己的时间, 随着误差减小, 逐步增加同步的间隔. 每次跳动, 都会重复这个调整的过程.

使用之前得弄清楚一个问题，ntpd与ntpdate在更新时间时有什么区别。ntpd不仅仅是时间同步服务器，它还可以做客户端与标准时间服务器进行同步时间，而且是平滑同步，并非ntpdate立即同步，在生产环境中慎用ntpdate，也正如此两者不可同时运行。

时钟的跃变，对于某些程序会导致很严重的问题。许多应用程序依赖连续的时钟——毕竟，这是一项常见的假定，即，取得的时间是线性的，一些操作，例如数据库事务，通常会地依赖这样的事实：时间不会往回跳跃。不幸的是，ntpdate调整时间的方式就是我们所说的”跃变“：在获得一个时间之后，ntpdate使用settimeofday(2)设置系统时间，这有几个非常明显的问题：

第一，这样做不安全。ntpdate的设置依赖于ntp服务器的安全性，攻击者可以利用一些软件设计上的缺陷，拿下ntp服务器并令与其同步的服务器执行某些消耗性的任务。由于ntpdate采用的方式是跳变，跟随它的服务器无法知道是否发生了异常（时间不一样的时候，唯一的办法是以服务器为准）。

第二，这样做不精确。一旦ntp服务器宕机，跟随它的服务器也就会无法同步时间。与此不同，ntpd不仅能够校准计算机的时间，而且能够校准计算机的时钟。

第三，这样做不够优雅。由于是跳变，而不是使时间变快或变慢，依赖时序的程序会出错（例如，如果ntpdate发现你的时间快了，则可能会经历两个相同的时刻，对某些应用而言，这是致命的）。因而，唯一一个可以令时间发生跳变的点，是计算机刚刚启动，但还没有启动很多服务的那个时候。其余的时候，理想的做法是使用ntpd来校准时钟，而不是调整计算机时钟上的时间。

NTPD 在和时间服务器的同步过程中，会把 BIOS 计时器的振荡频率偏差——或者说 Local Clock 的自然漂移(drift)——记录下来。这样即使网络有问题，本机仍然能维持一个相当精确的走时。