Network Clock Synchronization

We use chrony to synchronize the clocks of computers over a network.

$ apt-get install chrony

If you're running on an disconnected network you will have to manually download then install the chrony package.

$ aptitude download chrony timelimit

will download the two required packages to the current directory. Make sure to perform this command on a machine with the same architecture that you plan to install chrony on (I.E. i386, amd64 or armel).

To set up a topside, first install then edit the configuration file. A sample configuration file for a *master clock* — I.E. the machine that will always be on before the clients

```
1:2:3:4:5:6:7:8:                        keyfile /etc/chrony/chrony.keyscommandkey
                                        1driftfile /var/lib/chrony/chrony.driftloc
                                        al stratum 8allow 10/8       # allow anyon
                                        e with a 10.x.x.x address to sync to usall
                                        ow 192.168/16 # allow anyone with a 192.16
                                        8.x.x address to sync to usallow 172.16/12
                                          # allow any loopback address to sync to
                                        usmanual
```

To test the server:

$ ntpdate timeserver_ip

To setup a client first make sure you have IP access to the master and that the master is working.

Linux [1] clients should edit /etc/hosts to include a line like

timeserver IP.OF.MASTER.CLOCK

Set up the chrony.conf file in /etc/chrony/chrony.conf

A sample configuration file for a topside slave clock

```
1:2:3:4:5:6:7:8:9:                      server timeserverinitstepslew 20 timeserve
                                        rdriftfile /var/lib/chrony/chrony.driftkey
                                        file /etc/chrony/chrony.keyscommandkey 1lo
                                        cal stratum 12allow 10/8allow 192.168/16al
                                        low 172.16/12
```

A sample configuration file for a TS4710 slave clock:

```
1:2:3:4:5:6:7:8:9:10:                   server timeserverinitstepslew 20 timeserve
                                        rlinux [1]
                                        _hz 100 # this is absolutely REQUIRED!driftfile /pe
                                        rsistent/chrony.driftkeyfile /etc/chrony/chrony.key
                                        scommandkey 1local stratum 12allow 10/8allow 1
                                        92.168/16allow 172.16/12
```

To have chrony startup on boot on a debian [2] topside do

```
# update-rc.d chrony start 20 2 3 4 5 . stop 20 0 1 6 .
```

or add

/etc/init.d/chrony start

to /etc/rc.local

Testing

First, is it running?

$ ps aux | grep chronyd

On unsynchronized machines you should be able to

$ntpdate machine_ip

To see if a client is synchronized to the server

$chronyc tracking

and you should see the master's IP in 'Reference ID'. It may take a few minutes.

Initial Synchronization

When the clock is in an unknown state it is often preferred to manually synchronize the clock before starting chrony.

```
1:2:3:                                  sudo /etc/init.d/chrony stopsudo ntpdate t
                                        imeserversudo /etc/init.d/chrony start
```

Tags: Network Clock Synchronization [3]

Source URL: http://localhost:8888/kb2017/network-clock-synchronization

Links
[1] http://greenseainc.com/kb/lexicon/1#linux
[2] http://greenseainc.com/kb/lexicon/1#Debian
[3] http://localhost:8888/kb2017/tags/network-clock-synchronization