CCNA Exam Prep: Back to Networking Basics with Hank Preston and Patrick Gargano

Syncing Success: The Role of NTP in Network Operations

> Delve into the importance of Network Time Protocol (NTP) in maintaining synchronized time across network devices. In this session, review the steps to configure NTP for both client and server modes. Discover why synchronized time is crucial for accurate event logging, seamless coordination of network operations, and effective security measures. Master this important topic in “time” to take your exam.

Time and the passage of time is inescapble in life and network engineering. There are many "time" related topics that come up while you study for your CCNA. There is latency, how long it takes takes for a data packet to move from the source to destination. And jitter, how much variation in that time between different packets.

In this lab we are going to explore NTP, or Network Time Protocol. NTP is how IT systems can "synchronize their watches". These IT systems can be network devices like switches and routers, end user computers and mobile devices, and servers hosting the critical applications we rely on. The "systems" also include the unimaginable number of IOT devices that exist around us in cars, planes, trains, ships, manufacturing plants, and so on. So yeah, NTP is a foundational protocol that is often overlooked, until something goes wrong with it...

In this lab you will NTP with some hands on exercises focused on:

- Asking a router and switch "what time is it?" and manually setting the time.
- Configuring the NTP client in IOS to synchronize the clock with a known good source - an NTP server.
- Configuring the NTP server in IOS to act as a time source for other clients

> This lab touches on the following topic from the CCNA v1.1 Topics List 4.2 Configure and verify NTP operating in a client and server mode

Setup and Scenario

This setup of lab based demonstrations includes a small network made up of an IOS based switch and router, that provides internet access to a `netadmin` host. The network is preconfigured to provide connectivity to the Internet for the `netadmin` host with DHCP, DNS, and NAT services provided by `rtr01`. `sw01` is configured with a management IP address, and there are DNS entries for both network devices configured to allow `netadmin` to reach the devices by name.

*Be sure to **START** the lab before continuing to the demo labs.*

```
netadmin:~$ ping -c 2 sw01

PING sw01 (192.168.0.2): 56 data bytes
64 bytes from 192.168.0.2: seq=0 ttl=42 time=0.975 ms
64 bytes from 192.168.0.2: seq=1 ttl=42 time=1.418 ms

netadmin:~$ ping -c 2 rtr01

PING rtr01 (10.0.0.1): 56 data bytes
64 bytes from 10.0.0.1: seq=0 ttl=42 time=1.447 ms
64 bytes from 10.0.0.1: seq=1 ttl=42 time=1.125 ms
```

Go ahead and open the console for the `netadmin` host and try out the above pings yourself. "Pinging" is fun :-)!!

> Note: The credentials for `netadmin` are `cisco / cisco`.

Finding an available NTP server

Before we dive into this lab and explore configuring NTP to keep network time in sync, we need to find an available NTP server that is reachable from our lab. There are many publicly available NTP providers that can be freely leveraged by anyone around the world. Two sources for time provided by reliable clusters of NTP servers are NIST (National Institute of Standards and Technology) and pool.ntp.org.

The NTP server addresses for these sources are:

- NIST - `time.nist.gov`
- NTP Pool - `pool.ntp.org`

If you are running this lab from a CML server in your home lab, or a cloud provider, you should be able to leverage either of these NTP server addresses.

While these sources are trusted worldwide, some companies block access to these public NTP servers from within private networks and opt to provide their own NTP servers for internal systems. This means that if you are running this lab on a CML server that is hosted on a network that blocks access to public NTP servers, you'll need to check with the IT team to find the address for a reachable NTP server to query.

To test to see if the public servers are reachable, follow these steps.

1. Open the console to `netadmin` and login.
2. Use the `ntpd` service to query the public servers.

```
ntpd -dw -p pool.ntp.org -p time.nist.gov
```

Note: End the `ntpd` query by pressing `Cntrl-C` (`^C`).

3. If you see output that looks like the below, these public servers are NOT available for you to leverage for this lab.

```
ntpd: 'pool.ntp.org' is 104.234.67.234
ntpd: 'time.nist.gov' is 132.163.97.4
ntpd: sending query to 132.163.97.4
ntpd: sending query to 104.234.67.234
ntpd: timed out waiting for 132.163.97.4, reach 0x00, next query in 1s
ntpd: 'time.nist.gov' is 132.163.97.4
ntpd: timed out waiting for 104.234.67.234, reach 0x00, next query in 1s
```

4. See if you can find an available NTP server for your lab environment. For example, suppose you are told an NTP server is available at `10.1.1.1`. Let's test that address.

```
ntpd -dw -p 10.1.1.1

# Output
ntpd: sending query to 10.1.1.1
ntpd: reply from 10.1.1.1: offset:+36.082774 delay:0.002821 status:0x24 strat:3 refid:0x05f1c00a rootdelay:0.0
```

   - This output shows a reachable NTP server. Don't worry about the details of the output just now.

5. What should you do if the public servers aren't reachable AND you can't find information on an available server? Never fear! We have included an NTP server in the lab that can be used. It might not have exact accurate time, but it will work for the lab. This NTP server is available at `ntpserver.example.com`.

```
ntpd -dw -p ntpserver.example.com

# Output
ntpd: 'ntpserver.example.com' is 192.168.100.100
ntpd: sending query to 192.168.100.100
ntpd: reply from 192.168.100.100: offset:-0.191950 delay:0.002921 status:0x24 strat:1 refid:0x00000000 rootdel
```

Checking the current date/time on the router

So NTP can be used to set the time on network devices, but how do you check what the current clock is set to on an IOS router or switch?

1. Open up the console to `rtr01`, and enter enable mode.

2. Run the command `show clock` to check the current time on the router.

```
show clock

# Output
00:11:51.832 UTC Fri Jan 1 1993
```

3. Well.. unless you've gone back in time to the early nineties, the time displayed by the router probably doesn't look close to accurate.

4. Let's go ahead and "fix" the time manually. Check the current time on your computer and use the following command to correct the time.

   Note: The timezone setting on the router is UTC, be sure to convert your local time to UTC before setting the time in this step.

```
clock set 19:49:00 23 March 2025

! Command Format
clock set <MM:HH:SS> <1-31> <MONTH> <YEAR>

# Output
Mar 23 19:49:00.000: %SYS-6-CLOCKUPDATE: System clock has been updated from 00:00:46 UTC Fri Jan 1 1993 to 19:
```

5. Verify the time was updated correctly with `show clock`.

Excellent job! The clock on the router is right and we can all go out for some dinner and watch a movie right?

Configuring NTP client to maintain time sync

Unfortunately dinner and the movie is going to have to wait. It is just a fact of modern computing that computers eventually experience a time drift. Typically they drift slower over time. There are many contributing factors to time loss for computers, but it does happen. A little bit every day. And even if computers did NOT lose time, there is another reason why configuring NTP to maintain sync is important. How accurate were you when you configured the manual time in the previous step? Were you accurate to the second? To the millesecond? Accurate time for computers is important for a lot of applications.

So now that I've convinced you that we need a little help to keep our router's clock accurate, let us finally dive into NTP and use it to keep our router in sync.

> In the following steps, use the best NTP server for your particular lab environment. If a public NTP server is reachable, use that. If you have a private NTP server address that works, leverage that one. And if neither the public or private NTP servers are an option, then you can use `ntpserver.example.com`.

1. Back on the console for `rtr01` check the current status of NTP with the following commands.

   ```
   show ntp status

   # Output
   %NTP is not enabled.

   show ntp associations

   # No output
   ```

   > Okay, NTP is not running.

2. Now enter configuration mode and configure the `ntp server` for your router.

   ```
   ntp server ntpserver.example.com
   ```

3. Return to enable mode and check the status of NTP.

   ```
   show ntp status

   # Example Output
   Clock is unsynchronized, stratum 2, reference is 192.168.100.100
   nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
   ntp uptime is 74900 (1/100 of seconds), resolution is 4000
   reference time is EB8AE8C3.CC49BC90 (20:08:35.798 UTC Sun Mar 23 2025)
   clock offset is 18494.9973 msec, root delay is 1.95 msec
   root dispersion is 18499.91 msec, peer dispersion is 3.88 msec
   loopfilter state is 'FREQ' (Drift being measured), drift is 0.000000000 s/s
   system poll interval is 64, last update was 5 sec ago.
   ```

   - There is a lot of detail in the output above, much of it beyond the scope for the CCNA candidate.
   - The first line indicates that the router is currently `unsynchronized` with the server (ie `reference`)
   - `clock offset is 18494.9973 msec` indicates that there is an 18 second difference between the router's clock and the server. In this case our router is 18 seconds behind. If it was ahead, the value would be negative. This "offset" is why we are currently `unsynchronized`

4. NTP does NOT make large changes to a systems time, this could be disruptive to services. Rather it makes small adjustment with the goal being to bring the client into sync overtime. This process can take several minutes to hours depending on network conditions and the amount of offset involved.

   - You can speed up the process of synchronization by manually updating a devices clock to be closer to the NTP server.

5. Another handy show command is `show ntp associations`.

   ```
   show ntp associations

   # Output
     address         ref clock       st   when   poll reach  delay  offset   disp
   *~192.168.100.100 .                1     65     64     1  0.945 18495.1 7938.9
    * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
   ```

   - The output from this command will show the NTP status in a table format. Key details to note are:
   - The `*` indicating the currently selected reference clock
   - The `~` indicating that this is a "configured" server. Which most of your NTP servers will be.
   - The `offset` to indicate how far apart the client and server are from a time perspective

While NTP brings this router into sync with the server, we'll move onto enable the router to act as an NTP server for other network clients.

Enabling an IOS router to act as an NTP server

Now that we've got our edge router setup to keep its own clock in sync with a trusted NTP server, we want to set it up to provide time to other network devices. You might wonder, why not just have each and every device ask the same trusted server? Well, there are a few different reasons...

- Efficient use of bandwidth. NTP isn't a big drain on network resources, but even small packets add up if you have hundreds or thousands of devices asking "what time is it" every minute. So having a "local" NTP server for other devices to query for time is a great solution for effeciency.
- What happens if the public server can't be reached, or needs to be changed? It's much easier to have just a single device to update than every device on the network

- Distributing the load on the NTP servers themselves. NTP is designed to be setup in a distributed "tree" hierarchy. Where the initial reference time source is passed from tier to tier of NTP servers and eventually to clients. Distributing the load like this is better for the network overall

Speaking of the "tiers" of NTP servers, NTP has the concept of the "stratum" of the NTP server. Stratum is an integer from 1 - 15 that indicates how close to the reference clock this particular source is. A stratum of `1` is the base reference. A stratum of `2` is a server that directly learned the time from the `stratum 1` server. NTP clients can be configured with multiple NTP servers for redundancy. The client will prefer a lower stratum source over a higher stratum source.

Now that we've got some background information laid out, let's get the configuring!

1. Open the console for `rtr01` and enter configuration mode.
2. Enable the router to act as an NTP server with a stratum of 5 with this command.

   ```
   ntp master 5
   ```

3. Let's test that the NTP server is working, open the console for `netadmin` and send an NTP query to the router.

   ```
   ntpd -dw -p rtr01
   ```

   - The goal is that you'll get output that looks like this where we get a good reply from the server.

     ```
     ntpd: reply from 10.0.0.1: offset:-18.652104 delay:0.001246 status:0x24 strat:5 refid:0x01017f7f rootdela
     ```

   - If you get output that looks like this that indicates the "peer is unsynced", this is an indication that `rtr01` hasn't fully synchronized it's own time with its configured NTP server.

     ```
     ntpd: reply from 10.0.0.1: peer is unsynced
     ```

     - If you are seeing the above in your test, remove the configured NTP server on `rtr01` and repeat the query.

       ```
       no ntp server ntpserver.example.com
       ```

       In a "real network", the best solution would be to wait for the router to complete its synchronization. But with virtualized and simulated network devices like in this lab, the added overhead of simulation might prevent NTP from getting "fully healthy" and synchronized.

Once you've got a healthy result from the `ntpd` query test, your new NTP server is ready to go!

Using NTP to synchronize `sw01` from `rtr01`

We've setup the site router to act as an NTP server, let's use it to keep the switch's clock accurate!

1. Opening the console for `sw01` and check the current time. If it is "close" to the accurate time you can leave it alone. If it is wildly different from accurate, go ahead and manually fix it.
2. Now configure the switch to use `rtr01` as its NTP server. You can use the IP address for the router, or the DNS name.

   ```
   ntp server rtr01.example.com
   ```

3. Give it a few seconds for the initial NTP query to be sent and processed and check the status.

   ```
   show ntp status

   # Output
   Clock is unsynchronized, stratum 6, reference is 10.0.0.1
   nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
   ntp uptime is 2900 (1/100 of seconds), resolution is 4000
   reference time is EB8C338C.B3B64790 (19:39:56.702 UTC Mon Mar 24 2025)
   clock offset is -18902.4981 msec, root delay is 1.00 msec
   root dispersion is 19097.90 msec, peer dispersion is 189.44 msec
   loopfilter state is 'FREQ' (Drift being measured), drift is 0.000000000 s/s
   system poll interval is 64, last update was 15 sec ago.
   ```

   - Although the clock is indicated as unsynchronized, the synchronization process has started.
   - The switch reports a `stratum 6` with a reference of `10.0.0.1`. `10.0.0.1` is `rtr01`, and remember we configured it for `stratum 5`. The switch adds `1` to the stratrum value for its own clock.
   - Check the current clock offset for your network, how off is your time?

If you'd like to see the network get "in-sync", give it some time and check the status. You should see the offset shrink until it disappears and the status goes to `Clock is synchronized`. For example:

```
sw01#show ntp status
Clock is synchronized, stratum 6, reference is 10.0.0.1
nominal freq is 250.0000 Hz, actual freq is 250.1250 Hz, precision is 2**10
ntp uptime is 249000 (1/100 of seconds), resolution is 4000
```

```
reference time is EB8C3EB1.3541469D (20:27:29.208 UTC Mon Mar 24 2025)
clock offset is -91.5236 msec, root delay is 1.00 msec
root dispersion is 200.51 msec, peer dispersion is 64.49 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is -0.000499999 s/s
system poll interval is 64, last update was 49 sec ago.
```

Great Job!

That's all there is to the basic setup for NTP on network devices. It isn't a complicated protocol, but it is an important one.

As with many topics in networking, there are more topic areas you could explore. Accurate time is very important in network operations, and one way to disrupt a network is to inject incorrect time from unauthorized NTP servers. NTP supports authentication between servers and clients to make sure only "trusted" servers are used.

So for now, have a great ***time*** in your CCNA preparation studies!