**Introduction**

NTP is a protocol for synchronizing the time clocks on a collection of network devices using a distributed client/server or peer-to-peer mechanism. NTP uses UDP (User Datagram Protocol) as the transport mechanism. NTP evolved from the Time Protocol (RFC 868) and the ICMP Timestamp message (RFC 792).

NTP provides protocol mechanisms to specify the precision and estimated error of the local clock and the characteristics of the reference clock to which it may be synchronized.

**Products and software version that apply to this guide**

This guide applies to all AlliedWare Plus™ products, running version **5.4.4** or later.

Feature support may change in later software versions. For the latest information, see the following documents:

- The product's Datasheet
- The AlliedWare Plus Datasheet
- The product's Command Reference

These documents are available from the above links on our website at alliedtelesis.com.
NTP Overview

A public infrastructure is available that can be used as the time source for NTP in a network. The public infrastructure consists of a hierarchy of timing equipment—clock strata. The levels of the hierarchy are referred to as strata.

The heart of the infrastructure is the set of stratum-0 time sources which are Atomic Clocks—primary reference clocks, synchronized to national standards.

These are connected to widely accessible resources (such as backbone gateways or switches) operating as primary time servers, which constitute stratum 1 of the hierarchy. The primary time servers use NTP between them to crosscheck clocks, to mitigate errors due to equipment or propagation failures, and to distribute time information to local secondary time servers (stratum 2). The secondary time servers redistribute the time information to the remaining local hosts.

The hierarchical organization and distribution of time information reduces the protocol overhead, and allows selected hosts to be equipped with cheaper but less accurate clocks. NTP provides information which organizes this hierarchy on the basis of precision or estimated error.

Typically, the NTP process in an individual organization’s network will sync to a stratum 3 or stratum 4 time source, and use that source to provide a reference time for the devices in the network to synchronize to. However, that is not a fundamental requirement of the NTP protocol. As is discussed further below, it is quite feasible for the devices in a network to just peer with each other to achieve a consistent agreement on the time amongst themselves.

Operation of the NTP Protocol

In brief, the process that occurs in NTP is:

1. Devices sound out timestamp requests.
2. Other devices receive the requests and respond with packets containing timestamps, indicating their current system time.
3. The requesting devices receive these timestamps. They do not simply set their system time to the exact value contained in the received timestamp. They need to allow for the time it took the packet to travel to them. So, they need to calculate the travel time from the responder to themselves.
4. The travel time is calculated by knowledge of the time the request was sent, and when the response arrived, in addition to the time that the responder believed it received the request and sent its response. These values are all mashed together into formulae, out of which pop the travel time of the packets, and the time difference between the two devices’ system clocks.
5. The request/response is repeated a number of times, and the results are averaged, and outlier values removed, so that a progressively more accurate synchronisation between the two devices’ system times is achieved.
The same message format is used for both requests and replies. When a request is received, the server interchanges addresses and ports, fills in or overwrites certain fields in the message, recalculates the checksum, and returns it immediately. The information included in the NTP message allows each client/server peer to determine the timekeeping characteristics of its peers, including the expected accuracies of their clocks.

Each peer uses this information and selects the best time from possibly several other clocks, updates the local clock, and estimates its accuracy.

There is no provision in NTP for peer discovery or acquisition. Data integrity is provided by the IP and UDP checksums. No reachability, circuit-management, duplicate-detection, or retransmission facilities are provided or necessary.

By its very nature, clock synchronization requires long periods of time (hours or days) and multiple comparisons in order to maintain accurate timekeeping. The more comparisons performed, the greater the accuracy of the timekeeping.

Roles and Modes of Operation

In the protocol description above, it is evident that there are different roles that devices play in the process. There’s the ‘requester’ role and the ‘responder’ role.

There are two fundamental modes in which the NTP exchange can operate:

- **Client/Server mode**: The client requests and the server responds. The server will be a higher-stratum (i.e., lower stratum number) device than the client. So, the client’s time converges onto the server’s time.

- **Peer-to-peer mode**: Both devices in the conversation act as requesters and responders to each other. The peer devices are both at the same stratum. The devices’ times converge to each other.

Of course, a device that is operating as a client to a higher-stratum server can, at the same time, also be operating as a server to other, lower-stratum clients.

Moreover, a device that is operating as client or server to some partners, can, at the same time, also be operating as a peer to yet others.

The roles of client and server peer do not define the operation of a whole NTP device, they simply describe its role in any of the multiple NTP conversations in which it might be partaking at any given time.
In summary, an NTP device can take on one or more of the following three roles:

- **Client**—it can be a client to one or more servers, from whom it requests reference timestamps.
- **Server**—it sends reference timestamps in response to requests from one or more clients.
- **Peer**—it exchanges timestamps with one or more other peers, and they all collectively converge onto an agreed time.

### Different peer modes

When an NTP device is operating as a peer in an NTP conversation, it can perform this peer role in three different modes:

1. **Symmetric Active**: An NTP entity operating in symmetric active mode sends messages announcing its willingness to synchronize and be synchronized by its peers.
2. **Symmetric Passive**: An NTP entity enters symmetric passive mode in response to a message from a peer operating in Symmetric Active mode. An NTP entity operating in this mode announces its willingness to synchronize and be synchronized by its peers.
3. **Broadcast**: An NTP entity operating in broadcast mode periodically sends messages announcing its willingness to synchronize all of its peers but not to be synchronized by any of them.

### NTP on the Switch

The implementation of NTP on the switch is based on the following RFCs:

- RFC 958, Network Time Protocol (NTP)
- RFC 1305, Network Time Protocol (Version 3) Specification, Implementation and Analysis
- RFC 1510, The Kerberos Network Authentication Service (V5)

The switch supports client, server and peer modes.

When the switch is operating in client mode, then the server(s) to which it refers to must be more accurate clock source(s) than itself or another switch directly connected to a more accurate clock source.

If the switch receives a synchronization request from an NTP client, it temporarily changes to server mode. It replies to the request with the current time from the switch’s internal clock along with other information useful for synchronization. The switch’s internal clock is accurate to 0.005 seconds.
This example illustrates how to configure two switches, one at a Head Office and one at a Regional Office, to provide a network time service. The Head Office switch is connected to a primary time server and provides the most accurate time information. The switch at the Regional Office uses the Head Office switch as its peer to avoid the cost of an additional WAN connection but provides slightly less accurate time information.

- To configure NTP on the switch, the NTP module must be enabled and an NTP peer must be defined. NTP transfers time information in UTC format.
- To set the switch to automatically change the time when summer time starts and ends, enable a summer time offset setting.

Example configuration parameters for a network time service:

<table>
<thead>
<tr>
<th>Site</th>
<th>Regional Office</th>
<th>Head Office</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch name</td>
<td>RG1</td>
<td>HO1</td>
</tr>
<tr>
<td>IP Address of Switch</td>
<td>10.5.35.114</td>
<td>10.12.25.4</td>
</tr>
<tr>
<td>IP Address of Peer</td>
<td>10.5.35.113</td>
<td>172.16.7.3</td>
</tr>
</tbody>
</table>

**Step 1: Enable NTP and define the NTP peer.**

The NTP feature must be enabled on all switches that are to provide a network time service. Each switch must have a peer defined where the switch synchronizes its own internal clock. Enable NTP on the Head Office switch and specify a primary time server as the peer by using the commands:

```
awplus# configure terminal
awplus(config)# ntp peer 172.16.7.3
```

Note that you can also specify an IPv6 address for an NTP peer:

```
awplus# configure terminal
awplus(config)# ntp peer 2001:0db8:010d::1
```

**Step 2: Configure the NTP parameters.**

On each switch, the offset of local time from UTC time must be specified. In this example, both switches are in the same time zone, which is 12 hours ahead of UTC time. Use the following commands on both switches:

```
awplus(config)# clock timezone utc plus 12
```

**Step 3: Check the NTP configuration.**

Check the NTP configuration on each switch by using the command:

```
awplus# show ntp status
```

This command displays the following information on the Head Office switch:

- Clock is synchronized, stratum 0, actual frequency is 0.0000
- Hz, precision is 20 reference time is 00000000.000000000 (6:28:16.000 UTC Fri Feb 7 2036) clock offset is 0.000 msec
- root delay is 0.000 msec root dispersion is 0.000 msec
Secure Configuration of NTP

NTP, like any service on the switch, is vulnerable to attack. The simplest attack is a Denial of Service attack—flooding the NTP module with requests or updates. A more sophisticated attack involves spoofing a time source, thereby providing an incorrect time to the switch when the switch is acting as an NTP client. AlliedWare Plus provides two mechanisms for guarding against NTP attack: NTP filtering and NTP authentication.

NTP filtering

Filtering makes use of access lists to specify the IP addresses with which the switch’s NTP process will interact. A number of different types of access list can be applied to the NTP module, to control the different types of relationship that can occur in NTP. Access lists can include the following types:

peer

Time requests and NTP control queries will be accepted from devices whose addresses pass this access list. The switch’s NTP process is able to synchronize itself to a device whose address passes this access list.

query-only

NTP control queries are accepted from devices whose addresses pass this access list.

serve

Time requests and NTP control queries will be accepted from devices whose addresses pass this access list. The switch’s NTP process is not able to synchronize itself to a device whose address passes this access list.

serve-only

Only time requests are accepted from devices whose addresses pass this access list. The access lists are applied using the command:

```
awplus(config)# ntp access-group [peer|query-only|serve|serve-only] [<1-99>|<1300-1999>]
```

where `<1-99>` or `<1300-1999>` is the ID of a standard IP access list that has been created using the command:

```
awplus(config)# access-list {<1-99>|<1300-1999>} {deny|permit} <source>
```

NTP authentication

The purpose of NTP authentication is to enable the client to authenticate the server, and not vice versa. NTP authentication specifically deals with malicious users attempting to spoof a valid NTP server. The authentication is performed by using an MD5 key. The server and the client must be both configured to perform authentication and to use the same MD5 key.
Configuring authentication on the NTP client

On the client, there are four commands required to configure authentication:

1. **Enable authentication for NTP**
   
   ```
   awplus(config)# ntp authenticate
   ```

2. **Create one or more MD5 keys that can be used for NTP authentication**
   
   ```
   awplus(config)# ntp authentication-key <keynumber> md5 <key>
   ```
   
   where the `<keynumber>` is an ID number that will be used in other commands to refer to this key. The `<key>` is just a string, for example: AB123434, and is limited to the maximum command line length, which is 464 characters without spaces (less the length of the preceding parameters).

3. **Create a list of which of these keys is currently trusted (i.e. can currently be used for authentication).**
   
   ```
   awplus(config)# ntp trusted-key <keynumber>
   ```
   
   where the `<keynumber>` is the ID number of an MD5 key that has been created for NTP authentication.

4. **When defining the NTP server that the switch wishes to receive time updates from, specify the key that will be used to authentication the session to this server.**
   
   ```
   awplus(config)# ntp server <serveraddress> key <keynumber>
   ```
   
   where the `<keynumber>` is the ID number of an MD5 key that has been created for NTP authentication, and is in the list of trusted keys.
   
   The server must also be configured to use this key.

Configuring authentication on the NTP server

When configuring the switch as an NTP server, the configuration required to enable authentication on the server is:

1. **Enable authentication for NTP.**
   
   ```
   awplus(config)# ntp authenticate
   ```

2. **Create an MD5 key that can be used for NTP authentication.**
   
   ```
   awplus(config)# ntp authentication-key <keynumber> md5 <key>
   ```
   
   where the `<keynumber>` is an ID number that will be used in other commands to refer to this key. The `<key>` is just a string.

3. **Designate this key as currently trusted (i.e. can currently be used for authentication).**
   
   ```
   awplus(config)# ntp trusted-key <keynumber>
   ```
   
   where the `<keynumber>` is the ID number of the MD5 key.
All clients that wish to carry out authenticated sessions with this server must specify this key as the key they will use for sessions to this server.

**Note:** Authentication is initiated by the client. If the server is configured for authentication but the client is not, then the server will still accept the client’s session and serve time updates to the client. But if the client is configured for authentication and the server is not, then the session will never become established.

### Configuring authentication for NTP peers

It is also possible for a pair of NTP peers to authenticate each other. In this case, the configuration is the same as in the NTP client case, except that the final command is replaced by one that defines a peer relationship

```
awplus(config)# ntp peer <peeraddress> key <keynumber>
```

Of course, both peers must use the same key in the session with each other.

Filtering and authentication can be used together to provide a secure configuration. Filtering limits the addresses from which NTP messages will be accepted, and authentication enables a client to determine that a server is who it says it is.

### Troubleshooting

**Problem** The switch is not assigning the time to devices on the LAN.

**Solutions:** Check that the NTP peer’s IP address is entered correctly. Check that the NTP peer can reach the switch, by pinging the switch from the NTP peer.

**Problem** The switch’s clock does not synchronize with the NTP peer.

**Solution:** The switch’s clock can synchronize with the NTP peer only when its initial time is similar to the NTP peer’s time (after setting the UTC offset). Manually set the switch’s time so that it is approximately correct, and enable NTP again.

Check that the UTC offset is correct.

**Problem** The switch’s time is incorrect, even though it assigns the correct time to devices on the LAN.

**Solution:** The UTC offset is probably incorrect, or needs to be adjusted for the beginning or end of summer time.