13 Configuring Key Features

The OmniCore routing switch provides several features that can be configured:

- **Proxy ARP** allows hosts with no routing abilities to locate the media addresses of devices on other subnets or networks.
- **BOOTP and DHCP** simplify the configuration of network stations.
- **Bridging** multiple subnets can be enabled either globally or on a port basis.
- **Spanning Tree Protocol (STP)** detects and removes loops or redundant paths in the network. **Fast STP (FSTP)** is an enhancement that reduces the recovery time resulting from link failure or change in link status.
- **Jumbo packets** are packets that exceed the standard 1,518 bytes in size.
- **Port mirroring** allows the copying of one port's traffic by another port for network management purposes.
- **RMON** allow each port in a LAN segment to be monitored and managed.
- **Syslog** is a standard method for logging events and errors.
- **Trunk groups** allow several links to be grouped as one link, effectively increasing the bandwidth for that grouping.
- **LACP** allow links to be aggregated automatically, effectively increasing the bandwidth for the aggregation.
- **Network Time Protocol (NTP) and Timezone.** NTP is used to synchronize the time of a computer client or server to a reference time source that is synchronized to Coordinated Universal Time (UTC). Timezone implementation translates UTC time to local time.

Configuring Proxy ARP

Address Resolution Protocol (ARP) is a TCP/IP-based protocol that maps MAC addresses to IP addresses. Proxy ARP allows hosts with no routing abilities to locate the media addresses of devices on other subnets or networks. When a OmniCore routing switch receives an ARP request for a host that is not in the same network as the request sender, the switch must determine the best route to that device. If the destination is a local host and is known to the OmniCore routing switch, then it will send an ARP reply message that identifies the address to the sender. If the destination is a remote network, then the request is routed appropriately.

On the OmniCore routing switch, you may configure proxy ARP settings, as well as create ARP cache entries for a specified VLAN. These entries are kept in ARP tables that are used to determine the addresses of the devices attached to it. In short, an ARP table associates a MAC address with an IP address. The major proxy ARP commands in the OmniCore CLI are listed in the following tables. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual.*

Proxy ARP Global Commands

Command	Default	Description
arp proxy	disable	Allows switch to send ARP response on behalf of a remote subnet.
arp timeout	300 seconds	Determines length of time an ARP-cache entry remains before it is deleted.

Proxy ARP Interface Command

Command	Default	Description
vlan arp	no default	Creates ARP cache entries.

To configure proxy ARP:

1. Enable ARP.

```
OmniCore> arp
OmniCore/arp> proxy enable
```

2. (Optional) Define the timeout value for cache entries.

```
OmniCore/arp> timeout 400
OmniCore/arp> show
Proxy                                    :enable
ARP Timeout                              :400 secs
```

3. (Optional) Create a cache entry for the specified VLAN. Make sure you specify an existing VLAN. In this example, a cache entry with an IP address 10.0.3.34 and a MAC address of 00:60:97:29:68:AE is created for VLAN 5.

```
OmniCore/arp> ..
OmniCore> vlan 5
OmniCore/vlan=5> arp 10.0.3.34 macaddr 00:60:97:29:68:AE
OmniCore/vlan=5> arp show
IpAddress    MediaAddress       Type
---------    ------------       ----
10.0.3.34    00:60:97:29:68:AE  static
```

Configuring BOOTP and DHCP

The Bootstrap Protocol (BOOTP) allows a network station to automatically discover and receive startup information, such as its IP address, without having to be manually configured. This information is assigned by a designated server. The Dynamic Host Configuration Protocol (DHCP) is similar to BOOTP but uses the concept of a “lease” to assign an IP address to a user for an amount of time. This duration will depend on how long the user will use the Internet connection at a specific location. When the user moves to a different location in the network, a different IP address can automatically be assigned.

The major BOOTP commands in the OmniCore CLI are listed in the following tables. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual.*

BOOTP Global Commands

Command	Default	Description
bootp maxhops	4 hops	Defines the maximum number of hops that can be reached before a BOOTP packet is discarded.
bootp minsecs	0 seconds	Defines the duration that the OmniCore switch waits before forwarding a BOOTP or DHCP relay request.
bootp mode	off	Designates the BOOTP and DHCP relay mode.

BOOTP Interface Command

Command	Default	Description
bootp relay	no default	Creates a server relay or an interface relay entry.

To configure BOOTP and DHCP:

1. Specify the relay mode as BOOTP, DHCP, or both.

```
OmniCore> bootp

OmniCore/bootp> mode both
```

2. (Optional) Define the maximum number of hops and the minimum number of seconds for forwarding a relay request.

```
OmniCore/bootp> maxhops 8

OmniCore/bootp> minsecs 7150

OmniCore/bootp> show
Relay Mode                              :off
Max Number of Hops                      :8
Min Number of Seconds                   :7150 secs
```

3. Create the desired server and interface relay entries for the specified index. A relay establishes a method for the delivery of BOOTP and DHCP packets to a server that is on a separate IP network from the user. A server refers to the designated BOOTP or DHCP server to which a packet will be relayed, while an interface refers to the interface on which a packet will be rebroadcast.

   This example creates an interface relay entry with an index of 1 and an IP address of 10.0.0.45, and a server relay entry with an index of 3 and an IP address of 10.0.0.101.

```
OmniCore/bootp> relay 1 interface 10.0.0.45 create

OmniCore/bootp> relay 1 show
Relay Index                             :1
Server Address                          :0.0.0.0
Interface Address                       :10.0.0.45
Relay Current State                     :enable

OmniCore/bootp> relay 3 server 10.0.0.101 create

OmniCore/bootp> relay 3 show
Relay Index                             :3
Server Address                          :10.0.0.101
Interface Address                       :0.0.0.0
Relay Current State                     :enable
```

Configuring Bridging and Spanning Tree

Layer 2 bridging is the process of connecting together, and switching packets between, Local Area Networks (LANs). The bridging function is performed by a device that examines layer 2 information within packets.

The OmniCore routing switch supports bridging as defined in the IEEE 802.1d standard and Fast STP (FSTP) as defined in the IEEE 802.1w standard. FSTP protocol is active in the routing switch whenever *bridge stp status* is enabled. The *bridge mode* command can be used to specify a single spanning tree or a per-VLAN spanning tree.

STP detects and removes loops or redundant paths in a network. An enhancement to STP, Fast Spanning Tree Protocol (FSTP) reduces network recovery time resulting from a link failure or change in link status thereby reducing the data losses and session timeouts that could appear as a result from device failures or network topology changes. STP/FSTP can be enabled on Ethernet, Gigabit Ethernet, FDDI, and POS interface modules. STP/FSTP is disabled by default.

The major bridging and STP commands are listed in the following tables. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual.*

Bridging and STP Commands

Command	Default	Description
Single and Per-VLAN Commands		
bridge mode	single	Defines a single spanning tree, or multiple spanning trees (one per VLAN).
bridge stp status	enable	Activates bridge STP and FSTP (802.1w).
Single Commands		
bridge forward-time	15 seconds	Defines length of time the system waits to transmit forwards after a bridge port has been enabled.
bridge hello-time	2 seconds	Defines length of time the system waits between Bridge Protocol Data Units (BPDUs).
bridge max-age	20 seconds	Defines length of time a bridge will wait to receive BPDUs.
bridge priority	32768 (IEEE)	Defines a bridge's priority for designating a spanning tree's primary bridge.
port bridge stp path-cost	cost of 4 (1000 Mbps)	Defines path cost for a specified bridge port.
port bridge stp status	enable	Activates STP bridging on a specified port.
Per-VLAN Commands		
port bridge stp vlan path-cost	cost of 4 (1000 Mbps)	Defines path cost for a specified bridge port within the specified VLAN.
port bridge stp vlan status	enable	Activates STP bridging on a specified port within the specified VLAN.
vlan bridge forward-time	15 seconds	Defines length of time the system waits to transmit forwards after a bridge port has been enabled for the specified VLAN.

Bridging and STP Commands (Continued)

vlan bridge hello-time	2 seconds	Defines length of time the system waits between Bridge Protocol Data Units (BPDUs) for the specified VLAN.
vlan bridge max-age	20 seconds	Defines length of time a bridge will wait to receive BPDUs for the specified VLAN.
vlan bridge priority	32768 (IEEE)	Defines a bridge's priority for designating a spanning tree's primary bridge for the specified VLAN.
vlan bridge stp status	enable	Activates bridge STP and FSTP (802.1w) for the specified VLAN.

Note that when using single STP with tagged ports, all other switches must support untagged Bridge Protocol Data Units (BPDUs) across tagged links. Also, the OmniCore implementation of multiple STP (per-VLAN mode) supports tagged BPDUs on tagged ports and may be incompatible with other vendor's implementations. Check with Alcatel Technical Support if you have any questions concerning inter-operability issues.

A standard feature included with STP/FSTP on OmniCore switches is the *edgeport* command. This command enables a specified bridge port for immediate forwarding of packets thereby eliminating the listening/learning delay of STP. The commands are listed in the following table. For more information regarding these commands, refer to the *OmniCore CLI Reference Manual.*

Edgeport Commands

Command	Default	Description
port bridge stp edgeport	disable	Activates the specified bridge port for immediate forwarding of packets (*bridge mode* is set to "single").
port bridge stp vlan edgeport	disable	Activates the specified bridge port for immediate forwarding of packets within the specified VLAN (*bridge mode* is set to "pervlan").

Configuring Bridging and STP for a Single Spanning Tree

Single spanning tree mode should only be used when a single VLAN is active.

1. If the bridge mode has been previously set to "pervlan" mode, it is necessary to disable STP status before the bridge mode (in step 2) can be changed:

```
OmniCore> bridge stp status disable
```

2. Set the bridge mode to single mode.

```
OmniCore> bridge mode single

OmniCore> bridge mode show
   Bridge Mode         :single
```

3. Enable STP. When STP is enabled, all ports of the routing switch automatically participate in STP.

```
OmniCore> bridge stp status enable
```

4. (Optional) To disable STP on a port that you do not want to participate in STP, use the following example. In this example Ethernet port 1 on slot 3 is disabled.

```
OmniCore> ethernet 3 1 bridge stp status disable

OmniCore> ethernet 3 1 bridge stp show
   STP Status          :disable
   STP Port State      :disabled
   STP Path Cost       :19
   STP Edge Port       :disable
```

5. (Optional) To re-enable STP on a port that you want to participate in STP, use the following example.

```
OmniCore> ethernet 3 1 bridge stp status enable
```

While the defaults for bridging and STP features should be sufficient for most networks, you may, if necessary, modify various global and interface settings. See the *OmniCore CLI Reference Manual* for more information on bridging and STP commands.

Configuring Bridging and STP Per VLAN (Multiple VLANs)

Per-VLAN spanning tree should be used when more than one VLAN is active.

❖ Note ❖

Using the single spanning tree mode when more than one VLAN is active can cause certain network segments to lose connectivity.

1. If the bridge mode has been previously set to “single” mode, it is necessary to disable STP status before the bridge mode (in step 2) can be changed.

```
OmniCore> bridge stp status disable
```

2. Set the bridge mode to per-VLAN mode.

```
OmniCore> bridge mode pervlan

OmniCore> bridge mode show
   Bridge Mode         :pervlan
```

3. Re-enable STP. When STP is enabled, all ports, including all VLAN ports, of the routing switch automatically participate in STP.

```
OmniCore> bridge stp status enable
```

4. (Optional) To disable STP on a particular port within a particular VLAN, use the following example. In this example, STP is disabled on Ethernet port 1 on slot 3 of VLAN 2.

```
OmniCore> ethernet 3 1 bridge stp vlan 2 status disable

OmniCore> ethernet 3 1 bridge stp vlan 2 show
   Vlan Id             :2
   STP Status          :disable
   STP Port State      :forwarding
   STP Path Cost       :19
   STP Edge Port       :disable
```

5. (Optional) To re-enable STP on a particular port within a particular VLAN, use the following example.

```
OmniCore> ethernet 3 1 bridge stp vlan 2 status enable
```

While the defaults for bridging and FSTP features should be sufficient for most networks, you may, if necessary, modify various global and interface settings. See the *OmniCore CLI Reference Manual* for more information on bridging and STP commands.

Configuring Edgeport for a Single Spanning Tree

To enable Edgeport for a specified bridge port and for a single spanning tree, see the following example. This command only functions if the *bridge mode* command is set to “single” (single spanning tree domain).

```
OmniCore> slot 3 port 7 bridge stp edgeport enable
```

Configuring Edgeport Per VLAN

To enable Edgeport for a specified bridge port per VLAN, see the following example. This command only functions if the *bridge mode* command is set to “pervlan”.

```
OmniCore> slot 3 port 7 bridge stp vlan 3 edgeport enable
```

Configuring Jumbo Packet Settings

Jumbo packets are untagged packets that exceed 1,518 bytes in size. When a jumbo packet enters the OmniCore port, it can be fragmented into individual segments for processing by the switch fabric. The packet can then be sent as individual fragments (IP jumbo packets only) or reassembled and transmitted in whole.

How a jumbo packet is received, processed, and transmitted is determined by the switch's jumbo packet settings. These settings can be defined globally and on a per-port basis. Note that only ports on the 6-port 1000BASE-SX Gigabit Ethernet interface module are capable of processing jumbo packets.

❖ Note ❖

Do not mix jumbo and non-jumbo capable ports in the same VLAN if the OmniCore switch will be processing non-IP (raw) jumbo frames. Non-jumbo capable ports cannot reassemble raw fragments and thus transmit them as invalid frames. This is not an issue with IP frames since IP fragments are still valid frames.

The major jumbo packet commands in the OmniCore CLI are listed in the following tables. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual.*

Jumbo Packet Global Commands

Command	Default	Description
system jumbo frag-type	none	Designates the fragmenting mode.
system jumbo ipfrag-size	1518	Defines the IP fragment size.
system jumbo max-mtu	1518	Defines the maximum MTU value.
system jumbo reassembly-mode	none	Determines the jumbo packet reassembly mode.

Jumbo Packet Interface Commands

Command	Default	Description
port frag-type	useGlobal	Designates the fragmenting mode for an interface.
port reassembly-mode	useGlobal	Determines the reassembly mode for an interface.

To configure Jumbo Packet Settings:

1. Define the maximum transmission unit (MTU) value for all jumbo packets. The following example sets the MTU value to 9,000 bytes; you can customize this value for your own needs.

```
OmniCore> system jumbo

OmniCore/system/jumbo> max-mtu 9000

OmniCore/system/jumbo> max-mtu show
Max MTU                                  :9000
```

2. Set the fragmenting mode. This global command can be overridden on a port-by-port basis using the *port frag-type* command. Note that the value selected for this parameter should complement the reassembly mode set in step 3.

```
OmniCore/system/jumbo> frag-type ipandraw

OmniCore/system/jumbo> frag-type show
Fragmentation Type                      :ipandraw
```

3. Set the reassembly mode to determine how jumbo packets will be handled after they have been processed. This example specifies the *ipAndRaw* option. Depending on your needs, you may want to leave this parameter set to the default value of *none*. Refer to the *system jumbo reassembly-mode* command for more information.

```
OmniCore/system/jumbo> reassembly-mode ipandraw

OmniCore/system/jumbo> reassembly-mode show
Reassembly Mode                         :ipandraw
```

4. (Optional) Set the fragment size for IP jumbo packets if you want smaller fragments than the standard size of 1,518 bytes.

```
OmniCore/system/jumbo> ipfrag-size 555

OmniCore/system/jumbo> ipfrag-size show
IP Fragment Size                        :555
```

Configuring Port Mirroring

Port mirroring is a useful tool for locating network problems in an efficient and simple manner. Port mirroring involves the copying of traffic from one port (a source, or target, port) to another port (a destination, or mirror, port) for network monitoring purposes.

The OmniCore routing switch supports wire-speed port mirroring without affecting switch performance. An RMON probe can therefore be used to gather statistics other than Statistics, History, Alarms, and Events. Note that one ingress analyzer port, and up to five egress analyzer ports, are allowed at any one time on the OmniCore routing switch. In addition, up to 64 target ports can exist at any one time. The major port mirroring commands in the OmniCore CLI are listed in the following table. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual*.

Port Mirroring Commands

Command	Default	Description
port-mirror map	no default	Creates a map instance between a target port and mirror port.
port-mirror settings mode	ingress	Designates a mirror port's operation mode.
port-mirror settings status	disable	Enables port mirroring on the specified port.

To configure Port Mirroring:

1. Enable port mirroring on the desired destination (mirror) port. For this example, port 4 on slot 5 is the mirror port.

```
OmniCore> port-mirror settings 5 4

OmniCore/port-mirror/settings=5,4> status enable
```

2. Determine the mode of traffic (ingress or egress) that the mirror port will copy from a target port. If the port's mirror mode is ingress, then packets entering the source (target) ports will be transmitted out of the mirror port. If the port's mirror mode is egress, then packets exiting the target ports will be transmitted out of the mirror port.

```
OmniCore/port-mirror/settings=5,4> mode egress

OmniCore/port-mirror/settings=5,4> show
Slot Number                              :5
Port Number                              :4
Mirror Mode                              :egress
Mirror Status                            :enable
```

3. Display the capabilities of the desired port to ascertain whether or not it can act as a source (target) port. If it can, Yes will appear next to Copy Source Port, as shown in the following example. For this example, port 2 on slot 7 is the target port.

```
OmniCore/port-mirror/settings=5,4> ..

OmniCore/port-mirror> copy-caps 7 2 show
Slot Number                              :7
Port Number                              :2
Copy Source Port                         :Yes
Copy Dest Port                           :Yes
Copy Source TX                           :Yes
Copy Source RX                           :Yes
Copy Errored Frames                      :No
Copy Unaltered Frames                    :No
Copy All Good Frames                     :Yes
```

4. Map a target port to a mirror port. The traffic on the target port will then be copied to the mirror port. For this example,

```
OmniCore/port-mirror> map 7 2 5 4 create

OmniCore/port-mirror> map show
Source Port  Dest Port   Current State
-----------  ---------   -------------
7/2          5/4         active
```

Configuring RMON

Remote Monitoring (RMON) allows each port in a LAN segment to be monitored and managed. Statistics about network traffic on a LAN segment, both remotely and independently, can therefore be amassed for later delivery to the management console. In addition, an external RMON probe can gather these statistics from its part of the network without affecting the network performance in any way.

The OmniCore routing switch supports four group RMON (RFC 1757), which includes statistics, history, alarms, and events. RMON is disabled by default. Using the OmniCore routing switch to view RMON statistics requires that the following steps be performed first.

The major RMON commands in the OmniCore CLI are listed in the following table. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual.*

RMON Commands

Command	Default	Description
rmon status	enable	Enables RMON operation.
rmon slot port stats	no default	Displays RMON statistics for the specifyed port.

To configure RMON:

1. Enable RMON.

```
OmniCore> rmon

OmniCore/rmon> status enable
OmniCore/rmon> show
Status                                    :enable
```

2. Specify the desired port you wish to monitor. Note that the link status for a port must be active (up) to view its RMON statistics.

```
OmniCore> ..

OmniCore> slot 4 port 1 linkstat show
Link Status                                    :up

OmniCore> rmon slot 4 port 1 show
Port Number                                    :1
Drop Events                                    :0
Octets                                         :39430516
Packets                                        :382451
Broadcast Packets                              :306140
Multicast Packets                              :46689
CRC Align Errors                               :1
Undersize Packets                              :0
Oversize Packets                               :0
Fragments                                      :0
Jabbers                                        :0
Collisions                                     :0
64 Length Octets                               :242740
65 to 127 Length Octets                        :76930
128 to 255 Length Octets                       :50481
256 to 511 Length Octets                       :10171
512 to 1023 Length Octets                      :604
1024 to 1518 Length Octets                     :1526
```

Configuring Syslog

Syslog is a de facto standard method of logging events and errors, either locally on a single machine or at a logging machine over a network. The Syslog feature allows you to specify the network location of the remote host and also to determine what severity level of messages are sent. All OmniCore messages are written to an internal circular file (system log) residing on the EMM module. The Syslog output sends messages to a remote host using UDP. The Syslog utility provides a history of system events that may not be available within the OmniCore internal system log since all messages in the circular system log are eventually overwritten.

Please note that in order for Syslog entries to be written to the remote host's log, the Syslog utility must be enabled and configured on both the OmniCore switch and the remote host. In addition, the remote host must be configured to receive incoming Syslog messages. The following instructions assume that you have already configured the remote host to receive Syslog messages. The major Syslog commands in the OmniCore CLI are listed in the following table. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual*.

Syslog Commands

Command	Default	Description
system log mask	0xFF	Defines a bit mask value for Syslog message filtering purposes. Bit Priority Priorities included Use this mask 0 Emergency 0 0x01 1 Alert 0,1 0x03 2 Critical 0,1,2 0x07 3 Error 0,1,2,3 0x0F 4 Warning 0,1,2,3,4 0x1F 5 Notice 0,1,2,3,4,5 0x3F 6 Info 0,1,2,3,4,5,6 0x7F 7 Debug 0,1,2,3,4,5,6,7 0xFF
system log trap-mask	0xFF	Defines a bit mask value for generating traps upon the occurrence of important events. Bit Priority Priorities included Use this mask 0 Emergency 0 0x01 1 Alert 0,1 0x03 2 Critical 0,1,2 0x07 3 Error 0,1,2,3 0x0F 4 Warning 0,1,2,3,4 0x1F 5 Notice 0,1,2,3,4,5 0x3F 6 Info 0,1,2,3,4,5,6 0x7F 7 Debug 0,1,2,3,4,5,6,7 0xFF
system syslog debug	disable	Enables debugging of the Syslog utility.
system syslog port	514	Specifies the remote host UDP port to which Syslog messages are sent.
system syslog primary	Broadcast address of the EMM's primary interface.	Defines the primary remote host.
system syslog secondary	0.0.0.0	Defines the secondary remote host.
system syslog status	enable	Enables the transmission of Syslog messages.

To configure Syslog:

1. Enable the Syslog function.

```
OmniCore> system syslog

OmniCore/system/syslog> status enable
```

2. Define the primary remote host to which Syslog messages will be sent.

```
OmniCore/system/syslog> primary 10.0.101.89
```

3. (Optional) Define the secondary remote host.

```
OmniCore/system/syslog> secondary 10.0.45.45

OmniCore/system/syslog> show
Log to Remote Syslogd             :enable
Primary Address of Syslogd Host   :10.0.101.89
Secondary Address of Syslogd Host :10.0.45.45
UDP Port Num of Syslogd on Host   :514
Syslog Debug Status               :disable
```

4. Define a bit mask value for Syslog message filtering purposes. This example sets the 0 bit, ensuring that only messages of *emergency* level will be sent.

```
OmniCore/system/syslog> ..

OmniCore/system> log

OmniCore/system/log> mask 0x01

OmniCore/system/log> mask show
Syslog Priority Mask                      :0x01
```

5. Define a bit mask value for generating traps upon the occurrence of important events. This example sets the 2 bit, ensuring that traps of *emergency*, *alert*, and *critical* priorities will be generated.

```
OmniCore/system/log> trap-mask 0x07

OmniCore/system/log> trap-mask show
Mask of Priorities to Trap                :0x7
```

Configuring Trunk Groups

Trunking is the process whereby a number of physical connections can be grouped together (as a trunk group) to act as one logical connection between two directly connected OmniCore routing switches. This process increases the effective available bandwidth between the two switches. Up to four groupings with up to four ports apiece can exist at any one time.

When trunking links together, you must do the following:

- Define which ports are to be trunked.
- Ensure those ports are members of the same VLANs (an error message will appear if they are not).
- Ensure the connections exist between exactly two OmniCore routing switches. For resiliency purposes, the connections should not be connected via additional devices (such as repeaters) as this may impede the trunking software from recognizing link failures.
- Ensure the ports are of the same type, i.e., they must all share the same speed, duplex nature, flow control, etc. characteristics (an error message will appear if they are not). For example, it is not possible to combine a 10/100BASE-TX and a 1000BASE-X port into a trunk group.

- Ensure that 10/100BASE-TX ports in the same trunk group are located on the same interface module.
- Ensure that 10/100BASE-TX ports in the same trunk group fall into a port range of either 1-10 or 11-20. For example, ports 2-5 could be added to the same trunk group, but ports 9-12 could not.
- Before deleting a truck group, ensure that all affected ports are disabled. Otherwise, a data loop may occur.

❖ Tagging Over a Trunk Link ❖

Never use protocol-based VLANs on trunk links. If trunks need to carry multiple VLANs, use 802.1Q tagging. All ports in the trunck group must be tag-enabled if 802.1Q is used.

The major trunk group commands in the OmniCore CLI are listed in the following tables. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual*.

Trunk Group Global Command

Command	Default	Description
trunk	no default	Creates a trunk group.

Trunk Group Interface Command

Command	Default	Description
trunk slot port add	no default	Adds a port to a trunk group.

To configure a Trunk Group:

1. Define the trunk group. The creation of trunk group 2 is used for this example.

```
OmniCore> trunk 2 create
```

2. (Optional) Specify a name for the trunk group. Note that text strings with embedded spaces must be enclosed within quotations.

```
OmniCore> trunk 2
OmniCore/trunk=2> name “Building 2 PRs”
```

3. Add ports to the trunk group. The first port to be added to a trunk group is called the “lead port,” which in effect sets precedence for that group. All subsequent ports added to that group must therefore share the same characteristics as the lead port. In this example, port 2 on slot 4 is the lead port.

```
OmniCore/trunk=2> slot 4 port 2 add
OmniCore/trunk=2> slot 6 port 10 add
OmniCore/trunk=2> show
Trunk Id Port Members   Name
-------- ------------   ----
2        4-2, 6-10      Building 2 PRs
```

Configuring LACP

Link Aggregation Control Protocol (LACP) is the control protocol which establishes and maintains link aggregation. Link aggregation is a method of combining muliple links between systems to increase bandwidth, enhance resiliency, and provide load sharing. LACP controls the exchange of identity and the state information between links by exchanging PDUs (Protocol Data Unit). The information carried through the PDUs determine the next action. When two systems are connected by more than one link, LACP checks the properties of each port, such as port speed, type, and VLAN membership, to determine whether the links can be aggregated. If enabled, LACP aggregates any compatible links automatically. All ports are enabled for LACP by default. However, LACP is disabled globally by default.

The major LACP commands in the OmniCore CLI are listed in the following tables. For more information regarding these commands or other related commands, please refer to the *OmniCore CLI Reference Manual*.

LACP Global Commands

Command	Default	Description
lacp status	disable	Enables or disables LACP globally.
lacp linkagg name	Trunk Group <num>	Modifies the name of a link aggregation.

LACP Port Commands

Command	Default	Description
port lacp activity	active	Sets the LACP activity mode for a port.
port lacp stats show	no default	Displays statistics for the specified port.
port lacp status	enable	Enables or disables LACP for a port.
port lacp timers	long	Sets the LACP timer parameter for a port.

To configure LACP:

1. Enable LACP global status.

```
OmniCore> lacp status enable
```

2. (Optional) Specify a name for the link aggregation group. Note that text strings with embedded spaces must be enclosed within quotations.

```
OmniCore> lacp linkagg 1
OmniCore/linkagg=1> name “Accounting Mgr”
```

3. (Optional) Check the current LACP statistics for a particular port.

```
OmniCore> gigabit 5 1 lacp stats show
   Number of LACPDUs Received                :2433
   Number of Marker PDUs Received            :0
   Number of Marker Response PDUs Received   :0
   Number of Unknown PDUs Received           :0
   Number of Illegal PDUs Received           :0
   Number of LACPDUs Transmitted             :2433
   Number of Marker PDUs Transmitted         :0
   Number of Marker Response PDUs Transmitted :0
```

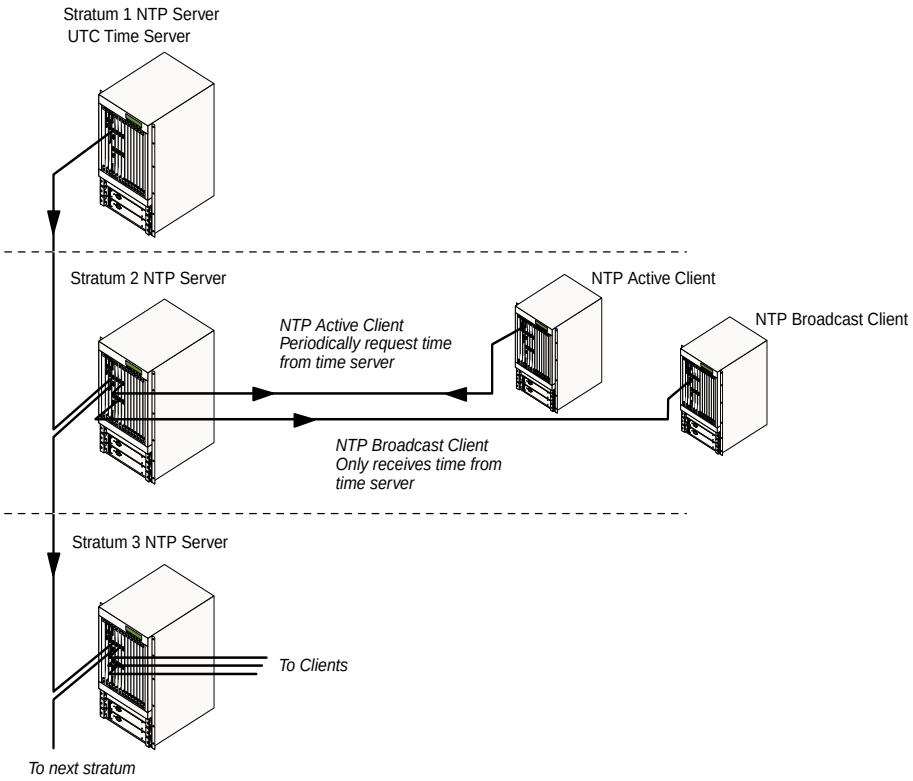
Configuring Network Time Protocol

NTP is a powerful utility for synchronizing system clocks over TCP/IP networks. It provides a precise timebase for networked workstations and servers.

Architecture

NTP defines a protocol to pass timekeeping information from primary reference sources to other time servers via the Internet, and to cross-check clocks and correct errors that may arise from equipment failures. Local net hosts or gateways, acting as secondary time servers, use NTP to communicate with one or more of the primary servers. These secondary servers distribute timekeeping information to the remaining local net hosts.

The NTP subnet can consist of a hierarchy of servers and clients, with each level in the hierarchy identified by a stratum number. The stratum number specifies the accuracy of each server. Primary servers are assigned as stratum 1 and each level downward in the hierarchy (secondary servers) are assigned as one greater than the preceding level. A stratum 1 server will most likely have a separate interface to a reliable source of time such as a radio clock or a GPS satellite. Stratum 2 servers are usually servers within a domain that obtain time from a number of primary servers over Internet paths and provide time to many local clients. These stratum 2 servers can be configured to peer with each other, comparing clocks and generating a synchronized time value.



NTP performs well over non-deterministic path lengths of packet-switched networks because it makes estimates of three key variables in the relationship between a client and a time server. These variables are network delay, dispersions of time packet exchanges (clock error), and clock offset (correction to apply). The OmniCore routing switch has the ability to "slew" the local system clock by a small amount in order to keep the local clock synchronized. Additionally if the local clock exceeds the correct time by a predetermined threshold, the protocol uses a step adjustment to adjust the local clock.

Proper implementation of NTP requires the implementation of time zones and the ability of the switch to make gradual phase time adjustments when necessary. Time zone information is required for the translation of UTC time to local time. Alcatel's CLI Timezone implementation provides the interface for establishing any time zone in the world.

Client/Server Models

Alcatel's implementation of NTP Client has two association modes, Active Client mode where the protocol requests periodic time updates from a designated primary time source and Broadcast Client mode where the protocol only accepts NTP time synchronization messages from a valid time source. A host that is operating in active client mode periodically sends an NTP message to a host that is operating in server mode. The server responds by interchanging addresses and ports, filling in the required information, and returning the message to the client. Broadcast-client mode is a passive mode that only receives messages from a network time server.

Clients are free to manage the intervals between sending NTP messages to suit local conditions. The active-client and broadcast modes of operation simplify the protocol machine without significant loss of accuracy, especially while operating over high-speed LANs.

NTP Server mode will respond to requested time requests and send out periodic synchronization messages. NTP Server is a configurable option through the CLI. When enabled, NTP server will run in broadcast mode which is intended for operation on high-speed LANs with numerous workstations; this implementation is ideal where the highest accuracies are not required. Additionally, NTP server can process unicast requests for synchronization.

NTP runs on a 24 hour clock and time is stored as UTC time instead of local time. However, all access to system time is converted to display the local time based on the time zone information stored in the OmniCore routing switch. NTP is designed to synchronize time keeping among a set of distributed servers and clients. An NTP server must be accessible by the client switch. NTP runs on top of User Datagram Protocol (UDP) which in turn runs on top of Internet Protocol (IP).

An NTP network usually gets its time from an authoritative time source. The time is then distributed across the network. A stratum is used to describe how many NTP hops away a machine is from an authoritative time source. Stratum 2 servers receive time from stratum 1 servers, stratum 3 servers from stratum 2 servers, and so on.

NTP Active Client

When configured in this mode, the switch will regularly send "time-of-day" requests to an NTP server. This NTP server is a pre-configured address. Active-client mode transmits time requests to designated time server and receives its replies.

A switch configured in active-client mode will periodically poll a designated time server on the network to request time synchronization. At a predetermined interval, the NTP machine will transmit an NTP request to a designated time server and will expect an NTP response. Information included in the NTP message lets the client determine the server time with respect to local time and adjust the local clock accordingly. Additionally, active client mode has a mechanism to switch from a primary time server to a secondary time server if a response is not received back from the primary time server in the configured time frame. Once the timeout is tripped, the client attempts a pre-configured number of retries. If the number of retires is reached, the client rolls over to the secondary time server. This means that all subsequent NTP requests will be sent to the secondary time server. For all practical purposes, the primary time server and the secondary time server provide the same accuracy. Rollback to the primary only takes place if the client fails to receive a response from the secondary time server.

When an NTP message is received, the offset between the peer clock and the local clock is computed and a filter algorithm is applied, discarding invalid or incorrect version messages. Upon receiving valid data, an update procedure is called to process the appropriate synchronization technique based on the offset. This may result in either a step-phase change or a gradual phase adjustment of the local clock to reduce the offset to zero.

NTP Broadcast Client

Broadcast-client mode is a passive mode that only receives messages from a network time server. This configuration is applicable when there is an NTP broadcast server on the local network that broadcasts the "time-of-day" on the network. Note that there is a dependency on a local NTP server to synchronize. This mode of operation is intended for operation on high-speed LANs where the highest accuracy is not required. When an NTP message is received, the offset between the peer clock and the local clock is computed and a filter algorithm applied, discarding invalid or incorrect version messages. Upon receiving valid data, an update procedure is called to process the appropriate synchronization technique based on the offset. This may result in either a step-phase change or a gradual phase adjustment of the local clock to reduce the offset to zero.

NTP Commands

Network Time Protocol has a base set of CLI commands and menus accessible under the *system* menu, and one command under *vlan ip*. The major NTP commands are listed in the following tables. For more information regarding the NTP commands, see the *OmniCore CLI Reference Manual*.

NTP Commands

Command	Default	Description
system ntp auto-disable	disable	Enables or disables auto-disable functionality.
system ntp bci	120 seconds	Set the NTP server broadcast interval.
system ntp interval	60 minutes	In active-client mode, defines the time interval to request synchronization from the primary or secondary time server.
system ntp clientmode	client	Sets the NTP client mode to active-client or broadcast mode.
system ntp primary	0.0.0.0	In active-client mode, sets the primary time server to request time from.
system ntp retries	3	In active client mode, sets the maximum number of retries for time synchronization before posting a message indicating an inability to reach the primary and secondary time server.
system ntp secondary	0.0.0.0	Sets the ip address of the secondary time server to request time from.
system ntp server	disable	Enables or disables NTP server status.
system ntp servermode	unicast	Sets NTP server mode to unicast or broadcast.
system ntp stats show	no default	Displays current system NTP synchronization information.
system ntp clientstatus	disable	Enables or disables NTP client status.

NTP Commands (Continued)

system ntp timeout	5 seconds	Sets the maximum time period to wait for a NTP reply following a time request.
system ntp unsync	reject	Rejects or permits unsynchronized NTP messages.
vlan ip ntp status	disable	Enables or disables NTP broadcasts from a VLAN IP interface.

Configuring NTP Client

Network Time Protocol has a base set of CLI commands and menus accessible under the *system* menu.

❖ Note ❖

Before configuring the OmniCore routing switch for NTP, the timezone must be set first, see *Configuring Timezone* on page 13-22.

Configuring NTP consists of the following tasks:

- Enable NTP client status.
- Set the timeout value.
- Identify the primary time server.
- Identify the secondary time server.
- Set the interval for synchronization.
- Set the number of tries for synchronization.
- Set the NTP client mode.
- Set the unsync option that rejects or permits synchronization messages.
- Set the auto-disable option that disables the protocol under certain conditions.

To configure NTP:

1. Enable NTP client status.

```
OmniCore> system ntp clientstatus enable
```

2. Set the timeout value. Set the time period (in seconds) to wait for a NTP reply following a time request. This option works in conjunction with the number of retries. Five seconds is set in the following example:

```
OmniCore> system ntp timeout 5
```

3. Identify the primary time server. Identify the primary time-server to request time from. Usually a stratum 1 or 2 time server, however, acceptable stratum servers are between 1 and 15. In the following example, 10.0.10.20 is the IP address of the primary time-server.

```
OmniCore> system ntp primary 10.0.10.20
```

4. Define the secondary time-server to request time from. Usually a stratum 1 or 2 time server, however, acceptable stratum servers are between 1 and 15. This secondary time server will be sent time requests only when the primary time source times out, or if responses are rejected. In the following example, 10.0.102.10 is the IP address of the secondary time-server.

```
OmniCore> system ntp secondary 10.0.102.10
```

5. Define the time interval (in minutes) to request synchronization from the designated primary or secondary time server. Ninety minutes is set in the following example.

```
OmniCore> system ntp interval 90
```

6. Sets the maximum number of retries for time synchronization before posting a message indicating an inability to reach primary and secondary time-server. For example:

```
OmniCore> system ntp retries 4
```

7. Set the NTP client mode to client or broadcast.

   - client - the switch is configured to NTP “active-client” mode and periodically polls a designated time server on the network to request time synchronization, i.e.,

```
OmniCore> system ntp clientmode client
```

   - broadcast - the switch only receives NTP messages from a network time server. This mode of operation is intended for operation on high-speed LANs where the highest accuracy is not required, i.e.,

```
OmniCore> system ntp clientmode broadcast
```

8. Set the **`ntp unsync`** command to reject or permit unsynchronized NTP messages.

```
OmniCore> system ntp unsync permit
```

9. Enable or disable the **`ntp auto-disable`** option.

```
OmniCore> system ntp auto-disable enable
```

   If enabled, auto-disable allows the NTP protocol to monitor itself and automatically disables the protocol under the following alarm conditions: (a.) exceeded maximum number of consecutive rejected messages, (b.) overflow calculation caused by an incorrect time or date setting, or (c.) invalid NTP message. This option can reduce network traffic when the NTP protocol has been incorrectly configured or experiences abnormal data for proper calculation.

If the OmniCore routing switch is going to be used as a time server, see *Configuring NTP Server* on page 13-21.

Configuring NTP Server

Network Time Protocol has a base set of CLI commands and menus accessible under the *system* menu.

❖ Note ❖

Before configuring the OmniCore routing switch for NTP, the timezone must be set first, see *Configuring Timezone* on page 13-22.

Configuring NTP consists of the following tasks.

- Enable the NTP client feature.
- Enable NTP Server functionality.
- Set the NTP Server mode.
- Enable the VLAN ip interface that will be broadcasting the NTP Server messages.

To configure the OmniCore routing switch for NTP Sever:

1. Enable the NTP client feature of the switch, see *Configuring NTP Client* on page 13-19.
2. After NTP client has been enabled, enable NTP Sever functionality by entering on the CLI:

   ```
   OmniCore> system ntp server enable
   ```

3. Set the NTP Sever mode to broadcast or unicast (default), for example,

   ```
   OmniCore> system ntp servermode broadcast
   ```

   Note that even if the mode is set to broadcast, the NTP Server will still respond to unicast requests for synchronization.

4. Enable the VLAN ip interface that will be broadcasting the NTP Server messages. Use the *vlan <vlan id> ip <ip addr> ntp status {enable | disable}* command, for example,

   ```
   OmniCore> vlan 2 ip 10.2.134.19 ntp status enable
   ```

Configuring Timezone

Timezone information is required for the translation of Universal Coordinated Time (UTC) to local time and the implementation of Network Time Protocol (NTP), see *Configuring Network Time Protocol* on page 13-16. Time zone implementation is a simple solution for establishing a time zone by providing an interface that allows the user to configure the switch to any time zone in the world. Additionally, a set of default standard time zones is provided for easy setup, and include all related daylight saving time (DST) information. Setting the switch's time zone to one of the provided defaults is the easiest way to set time zone information. Since all time zones could not be incorporated into the switch, the option for creating a unique time zone is also provided.

Timezone Commands

Time zone information can be set through the command line interface (CLI). All timezone commands are located under the *system* menu. For details on the timezone commands, refer to the *OmniCore CLI Reference Manual.*

Timezone Commands

Command	Default	Description
system timezone calendar show	current month	Displays a calendar of a past, current, or future month or week.
system timezone dst-offset	60 minutes	Displays or sets the Daylight Saving Time (DST) offset from the current time zone setting.
system timezone offset	no default	Sets the time offset in minutes from Greenwich Mean Time (GMT).
system timezone summer-time	no default	Sets Daylight Saving Time (DST).
system timezone zone	no default	Displays or sets the time zone to a default time zone, or creates a custom time zone.
system timezone zone-table show	no default	Displays a list of the most common standard time zones and their offset from Greenwich Mean Time.

Configuring Timezone

The following subsections explain how to set or modify a time zone via the command line interface (time zones can also be set through a SNMP client interface). All timezone commands are located under the *system* menu. Selecting one of the provided defaults is the easiest way to set time zone information. Since all time zones could not be incorporated into the switch, the option for creating a unique time zone is also provided. Time zone modifications are saved and take immediate effect without having to save the config file.

Current Timezone

To see the current time zone information, use the *timezone show* command:

```
OmniCore> system timezone show
   Standard Timezone Name                :GMT
   Offset from GMT (minutes)             :0
   Daylight Savings Setup                :
   DST Frequency                         :recurring
   DST Offset from Std Time (minutes)    :60
   DST Starting Week                     :first
   DST Starting Day                      :Sunday
   DST Starting Month                    :April
   DST Starting Time                     :02:00
   DST Starting Year                     :0
   DST Ending Week                       :last
```

Default Timezone

To set the OmniCore routing switch to a default time zone:

1. List the default time zones by using the *timezone zone-table show* command and determine the time zone for your region. For example:

```
OmniCore> system timezone zone-table show
Zone Alias      Name      DST Name  Offset
--------------  --------  --------  ------
None                                0
MiddleEuro      MET       MET DST   60
EasternEuro     EET       EET DST   120
Atlantic        AST       ADT       -240
Eastern         EST       EDT       -300
Central         CST       CDT       -360
Mountain        MST       MDT       -420
Pacific         PST       PDT       -480
Arizona         AST                 -420
Greenwich       GMT                 0
WesternEuro     WET       WET DST   -60
AustEast        EST       EST       600
AustCentral     CST       CST       630
AustWest        WST                 480
   Number of Entries Displayed: 14
```

2. Use the *timezone zone* command to set the time zone and the *timezone show* command to view the time zone's characteristics. For example, if Greenwich (GMT) time is to be the time zone for the switch:

```
OmniCore> system timezone zone Greenwich

OmniCore> system timezone show
   Standard Timezone Name                :GMT
   Offset from GMT (minutes)             :0
   Daylight Savings Setup                :
   DST Frequency                         :recurring
   DST Offset from Std Time (minutes)    :60
   DST Starting Week                     :first
   DST Starting Day                      :Sunday
   DST Starting Month                    :April
   DST Starting Time                     :02:00
   DST Starting Year                     :0
   DST Ending Week                       :last
```

Custom Timezone

The steps below describe how to create a custom time zone, or a time zone not listed in the table of the *timezone zone-table show* command.

1. Determine a name for the time zone to create and ensure the name is not listed in the "Zone Alias" or "Name" columns in the table of the *timezone zone-table show* command (see below), otherwise, you will not be able to modify the time zone.

```
OmniCore> system timezone zone-table show
Zone Alias      Name      DST Name  Offset
--------------  --------  --------  ------
None                                0
MiddleEuro      MET       MET DST   60
EasternEuro     EET       EET DST   120
Atlantic        AST       ADT       -240
Eastern         EST       EDT       -300
Central         CST       CDT       -360
Mountain        MST       MDT       -420
Pacific         PST       PDT       -480
Arizona         AST                 -420
Greenwich       GMT                 0
WesternEuro     WET       WET DST   -60
AustEast        EST       EST       600
AustCentral     CST       CST       630
AustWest        WST                 480
   Number of Entries Displayed: 14
```

2. After determining a name, use the *timezone zone* command to create the time zone. In the following example, "MyCustomTimeZoneName" is the name of the time zone to create.

```
OmniCore> system timezone zone MyCustomTimeZoneName
```

3. Use the *timezone offset* command to set up the offset from Greenwich Mean Time (GMT), if applicable. Sixty minutes west of Greenwich is used in the following example.

```
OmniCore> system/timezone> offset -60
```

4. See the following Daylight Saving Time procedure for setting up daylight saving time.

Daylight Saving Time

To set the daylight saving time (DST), and the following information must be known.

- Is this daylight savings time to recur every year or limited to a current year?
- When does it start? For example, the first Sunday in April at 2:00 am
- When does it end? For example, the last Sunday in October at 2:00 am
- What is the offset from standard time, for example, 60 minutes.

1. Once the above information is known, use the *timezone summer-time recurring* or *timezone summer-time limited* command to setup the DST information. If daylight savings is recurring yearly, do not specify the year in the summertime command.

   An example of setting a DST recurring yearly:

```
OmniCore> system timezone summer-time MyDSTsetup_Name recurring first mon jan
3:45 last wed july 3:47
```

   An example of assigning DST to particular years:

```
OmniCore> system timezone summer-time MyDSTsetup_Name limited first mon jan
3:45 2006 last wed july 3:47 2009
```

2. Use the *timezone show* command to view the DST parameters.

```
OmniCore> system timezone show
Standard Timezone Name           :MyCustomTimeZoneName
Offset from GMT (minutes)        :-480
Daylight Savings Setup           :MyDSTsetup_Name
DST Frequency                    :limited
DST Offset from Std Time (minutes) :60
DST Starting Week                :first
DST Starting Day                 :Monday
DST Starting Month               :January
DST Starting Time                :03:45
DST Starting Year                :2006
DST Ending Week                  :last
DST Ending Day                   :Wednesday
DST Ending Month                 :July
DST Ending Time                  :3:47
DST Ending Year                  :2009
```

3. If necessary, use one of the following commands to set the DST offset.

- *timezone dst-offset* command to set the DST offset from your current time zone:

  ```
  OmniCore> system timezone dst-offset 60
  ```

- *timezone offset* command to set the DST offset from Greenwich Mean Time:

  ```
  OmniCore> system timezone offset 60
  ```