MXK-F Management Guide

For software version 3.3
Aug, 2020
Document Part Number: 830-04154-06

Enabling the
Hyper-Connected World

DZS AMERICAS
Global Headquarters &
Regional Headquarters
Plano, TX, USA
info@dzsi.com
www.dzsi.com/contact-us/

DZS-KEYMILE EMEA
Regional Headquarters
Hanover,
Germany
info.emea@dzsi.com
www.keymile.com/en/web/keymile/contact_sales

DZS KOREA-APAC
Regional Headquarters
Seongnam-si, Geyonggi-do,
South Korea
info@dzsi.com
www.dzsi.com/contact-us/

TABLE OF CONTENTS

ABOUT THIS GUIDE

This guide is intended for use by installation technicians and system and network administrators. It explains how to configure the MXK-F, provision cards, create IP interfaces, configure bridges, and other system administration and networking tasks.

This chapter describes:

Style and Notation Conventions

The following conventions are used in this document to alert users to information that is instructional, warns of potential damage to system equipment or data, and warns of potential injury or death. Carefully read and follow the instructions included in this document.

Caution: A caution alerts users to conditions or actions that could damage equipment or data.

Note: A note provides important supplemental or amplified information.

Tip: A tip provides additional information that enables users to more readily complete their tasks.

WARNING! A warning alerts users to conditions or actions that could lead to injury or death.

WARNING! A warning with this icon alerts users to conditions or actions that could lead to injury caused by a laser.

Typographical Conventions

Table 1describes the typographical styles that this guide uses to represent specific types of information.

Table 1: Typographical Styles

Bold	Used for names of buttons, dialog boxes, icons, menus and profiles when placed in body text, and property pages (or sheets). Also used for commands, options, parameters in body text, and user input in body text.
`Fixed`	Used in code examples for computer output, file names, path names, and the contents of online files or directories.
`Fixed Bold`	Used in configuration examples for text entered by users.
Italic	Used for book titles, chapter titles, file path names, notes in body text requiring special attention, section titles, emphasized terms, and variables.
PLAIN UPPER CASE	Used for environment variables.
Command Syntax	Brackets [ ] indicate optional syntax. Vertical bar | indicates the OR symbol.

Related Documentation

Refer to the following documents for additional information:

MXK-F Hardware Installation Guide — contains information about the MXK-F chassis including environmental and power requirements, procedures on how to prepare, install, and maintain the MXK-F chassis, install and remove slot cards, and to add them to the system to make them available for configuration.

MXK-F Configuration Guide — explains how to configure bridging, link aggregation, and other configuration tasks.

MXK-F Release Notes— refer to the release notes for software download, software installation, and software upgrade information, and for changes in features and functionality of the product.

Acronyms

Table 2 provides a description of the acronyms that are related to DZS products and may be found in this manual.

Table 2: Acronym Definitions

Acronym	Description
ARP	Address Resolution Protocol
CPE	Customer Premise Equipment - equipment that is installed at the customer's location
GPON	Gigabit Passive Optical Network defined by ITU G.984; also used in this document to represent the newer, related protocols XG-PON, XGS-PON and NG-PON2
MIB	Management Information Base
NG-PON2	Next Generation Passive Optical Network 2 defined by ITU G.989; within this document (unless otherwise noted) referred to as GPON, meaning "one of the GPON related protocols."
OLT	Optical Line Terminal. An electronic device/equipment at the beginning (core/network side) of the optical access network that connects to ONT/ONUs. Sometimes used to refer to an OLT port other times to refer to an OLT Line Card.
ONT	Optical Network Terminal (a type of ONU). An electronic device at the end (subscriber side) of the access optical network located on the subscriber side.
ONU	Optical Network Unit. An electronic device at the end of the access optical network located on the subscriber side.
SFP	Small Form factor Pluggable
SLMS	Single Line Multi-Service
SNMP	Simple Network Management Protocol
TFTP	Trivial File Transfer Protocol
XG-PON	Asymmetric 10 G Passive Optical Network defined by ITU G.987; within this document (unless otherwise noted) referred to as GPON, meaning "one of the GPON related protocols."
XGS-PON	Symmetric 10 G Passive Optical Network defined by ITU G.9807; within this document (unless otherwise noted) referred to as GPON, meaning "one of the GPON related protocols."
ZMS	Zhone Management System
zNID	A DZS manufactured ONT/ONU.

Contacting DZS Quality & Service

All new DZS equipment purchases include one year of HW Warranty and 90 days of Bronze-level Technical Support.

If your product is not within 90 days of the purchase date or you do not have a valid support contract, please contact your local sales representative to get a quote on a support contract.

Customers with a valid support contract or are eligible for 90 days technical support associated with a new product purchase can request technical support by opening a case at:

https://dzsi.com/support/#TAC

Customers with a valid support contract have access to technical product documentation, software downloads, knowledge base and consultation on the covered DZS product at the same support portal.

For repair services within the HW Warranty period or under an Extended Warranty support contract, a Return Material Authorization (RMA) must be obtained before sending the equipment for repair. RMA requests can be submitted at:

https://dzsi.com/support/#RMA

Technical support

The Technical Assistance Center (TAC) is available with experienced support engineers who can handle questions, assist with service requests, and help troubleshoot systems.

Hours of operation	Monday - Friday, 8 a.m. to 6 p.m, Eastern Time (excluding U.S. holidays)
Telephone (North America)	877-946-6320, prompt #3, #1
Telephone (International)	510-777-7133, prompt #3, #1
E-mail	support@dzsi.com
Web - available 24 x 7 to submit and track field issues/ problem reports	https://dzsi.com/support/#TAC Click **DZS Problem Reporting System**

If you purchased the product from an authorized dealer, distributor, Value Added Reseller (VAR), or third party, contact that supplier for technical assistance and warranty support.

Hardware repair

If the product malfunctions, all repairs must be authorized by DZS with a Return Merchandise Authorization (RMA) and performed by the manufacturer or a DZS-authorized agent. It is the responsibility of users

requiring service to report the need for repair to DZS Quality & Service as follows:

- Complete the RMA Request form (https://dzsi.com/support/#RMA) or contact DZS Quality & Service via phone or email:

  Hours of operation: Monday - Friday, 8 a.m. to 6 p.m, Eastern Time (excluding U.S. holidays)

  E-mail: support@dzsi.com (preferred)

  Phone:877-946-6320 or 510-777-7133, prompt #3, #2

- Provide the part numbers and serial numbers of the products to be repaired.
- All product lines ship with a minimum one year standard warranty (may vary by contract). DZS warrants all repairs for 90 days or the remainder of the standard warranty (whichever is greater).
- DZS will verify the warranty and provide the customer with a repair quote for anything that is not under warranty. DZS requires a purchase order or credit card for out of warranty fees.

CHAPTER 1 MXK-F MANAGEMENT

This chapter provides an overview of MXK-F and includes:

1-1 MXK-F OVERVIEW

The MXK-F platform is an intelligent terabit access concentrator that provides a scalable multi-service architecture on the SLMS access operating system.

The MXK-F, in conjunction with zNIDs, provides a complete end-to-end access solution for optical fiber, Access Network deployments that provide triple-play services to subscribers (e.g. using GPON, XGS-PON or Active Ethernet). zNIDs at customer sites extend network intelligence all the way to subscribers with the ability to fine-tune performance.

MXK-F fabric cards are the primary communication channel between subscribers and upstream networking devices.

The MXK-F can be deployed in Central Office environments or outdoor controlled environmental vaults for remote terminal applications. The MXK-F is intended for restricted access locations only.

Note: The term “GPON” is used throughout this document to represent GPON and the more recent related protocol/products XG-PON, XGS-PON and NG-PON2, unless noted otherwise. The CLI command syntax for all of these protocols always use “gpon” (e.g. the argument “xgspon” is not used in CLI syntax).

Note: The terms “ONU” and “ONT” have been used interchangeably throughout this document unless noted otherwise. The CLI command term “onu” is unique and can not be substituted with “ont”.

1- 2 MXK-F CHASSIS

The MXK-F has two basic chassis:

- MXK-F14xx Chassis, page 16

  There are two 14RU chassis, The MXK-F1421 and the MXK-F1419 which support 16 and 14 line cards, respectively.

- MXK-F219 Chassis, page 18

  A 2U chassis which supports two line cards.

The line cards for the MXK-F14xx and F219 are the same. The Fabric and Management cards are different as described in the following sub-sections.

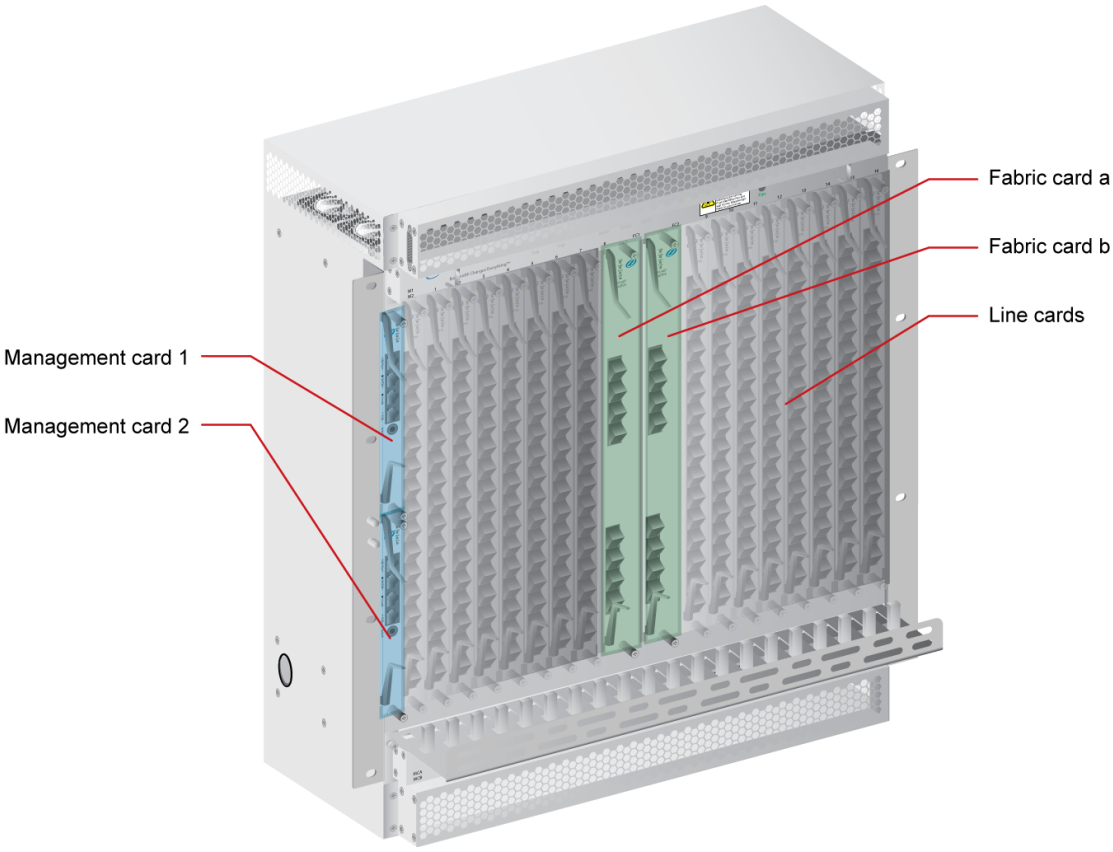
Both chassis use the same Port Naming conventions that are explained here: *Port Naming Convention* on page 19.

1- 2.1 MXK-F14xx Chassis

The MXK-F 1421 has 18-service slots; 16-slots are dedicated for subscriber facing Line Cards (LCs), and 2-Fabric Card (FC) slots initially dedicated to provide redundant network facing interfaces. Additionally, MXK-F 1421 has two half-height slots dedicated for high-availability redundant Management Cards (MCs). The MXK-F 1421 chassis is 21 inch wide and 14U high (25U metric), and is designed to be installed within a 600mm wide 300mm deep ETSI-compliant rack or cabinet.

The MXK-F 1419 chassis is similar, but narrower with two less card slots on the right hand side of the chassis. The MXK-F 1419 chassis is 19 inch wide and 14U high (25U metric), and is designed to be installed in a standard 19" rack, or within a 600mm wide 300mm deep ETSI-compliant rack or cabinet using mounting adapter brackets.

Figure 1: MXK-F Basic Chassis Cards



The types of cards supported on the MXK-F14xx are management, fabric, and line cards as shown in Figure 1. There are removable fan trays on the top and bottom of the chassis. Both fan trays are required to provide adequate cooling. The top fan tray assembly also includes an alarm input/output connector.

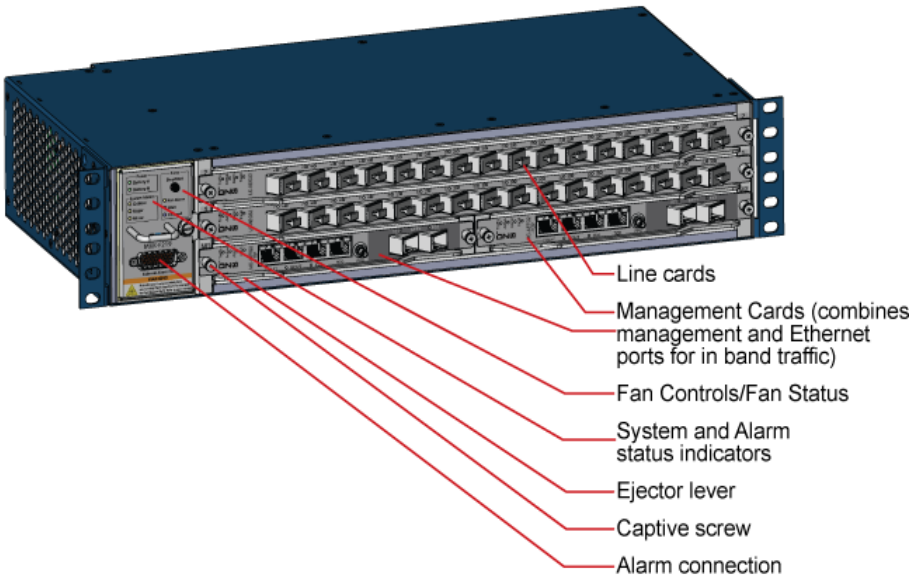
1- 2.2 MXK-F219 Chassis

MXK-F219 is a 2U chassis with two service slots and two management card slots. The management slots may contain two high-availability redundant Management Cards (MCs), though only one is required. The management cards include uplink fabric interfaces, though the upstream processing is done on the line cards which keeps the management plane and data plane more independent. The MXK-F219 management cards look similar to but are different from the MXK-F14xx management cards (different part numbers).

The 2U chassis is 19 inch wide and 2U high, and is designed to be installed in a standard 19" rack, or within a 600mm wide 300mm deep ETSI-compliant rack or cabinet using mounting adapter brackets.

The MXK-F219 chassis has one removable fan tray for cooling. Alarm inputs and outputs are provided on a connector on the fan tray.

Figure 2: The MXK-F219 chassis with management and line cards.



1- 2.3 Port Naming Convention

The MXK-F uses the following internal Port/Interface Naming Convention:

chassis # - card slot # - port # - sub-port # / media type (e.g. 1-1-5-0/eth)

- "chassis #" = "1"
- "card slot #" = # or letter identifier on chassis slot (e.g. "1" or "m1")
- "port #" = # on card faceplate that identifies the port
- "sub-port # / media type"
  - For Uplink & Active Ethernet port: sub-port#/media type = "0/eth"
  - For GPON port there are two syntax options:
    - > sub-port # / media type = "(GEM#+ONU#)/gponport"
      The GEM and ONT IDs are combined into a single field.
      (e.g. 701/gponport; GEM base # = 7xx; xx = ONU # = 01)
    - > sub-port # / media type = "ONU#/gpononu gem GEM#"
      The GEM and ONT IDs are independently specified.
      (e.g. "1/gpononu gem 703"; GEM # = 703; ONU # = 1)

The sub-port field of a GPON port name depends on several factors, which are in part determined by each Service Provider and by each ONU manufacturer's design (ONU capabilities).

The MXK-F provides a wide and flexible range of ways to assign ONU and GEM port IDs that allow for automatically and manually configured ONU and GEM port IDs. Each Service Provider independently decides how to use/ assign IDs to ONUs and GEM Ports within their network.

Some GPON and XGS-PON zNID/ONUs are pre-configured with GEM Ports for services like the CPE Manager, HSI, IPTV, and VoIP services.

- GPON pre-configured range: 501 to 564
- XGS-PON pre-configured range: 1101 to 1356
- Check the GPON and XGS-PON zNID/ONU Configuration Guide to learn which GEM Port IDs are available for Unified Service Provisioning.

There are three Provisioning systems that the MXK-F supports to configure GPON OLT, ONU and GEM ports: Smart OMCI, Dynamic OMCI and Universal Service Provisioning (USP). These three systems are explained in detail in the *MXK-F Configuration Guid*e. Table 3 and Table 4 provide a brief summary of how GPON and XGS-PON ONU and GEM IDs can be assigned.

Table 3: GPON and XGS-PON GEM Port ID Assignments with Dynamic OMCI and USP Provisioning Systems

Gem Port Range Category	GPON	XGS-PON	Gem Port Range Category
Reserved		1024 - 1035	Reserved
Available for any use on any ONU	257 - 499	1036 - 1099	Available for any use on any ONU
Preferred/Reserved (sometimes pre-configured by some ONU types): GEM # = 500 + ONU # with ONU = 1 to 128 (e.g. 547 = ONU 47)	500 - 628	1100 - 1356	Preferred/Reserved (sometimes pre-configured by some ONU types): GEM # = 1100 + ONU # with ONU = 1 to 256 (e.g. 1147 = ONU 47)
Available for any use on any ONU	629 - 3828	1357 - 8556	Available for any use on any ONU
Reserved: Internally assigned GEM # when SNMP is used with USP (one VLAN is also reserved for each GEM assignment)	3829 - 3956	8557 - 8812	Reserved: Internally assigned GEM # when SNMP is used with USP (one VLAN is also reserved for each GEM assignment)
Reserved		8813 - 8959	Reserved

Table 4: GPON and XGS-PON GEM Port ID Assignments with Smart OMCI Provisioning Systems

Gem Port Range Category	GPON	XGS-PON	Gem Port Range Category
	-	1024 - 1099	Reserved
GEM Port # Available for use. GEM# = x00 + ONU#, where x = 5, 7, … 35 in steps of 2, and with ONU = 1 to 128. The GEM base “x00” value can be used to identify a service type that is used commonly across similar ONUs (e.g. 500 for Data service; 700 for Voice; 900 for Video; e.g. GEM 907 = Video on ONU 7).	500 - 3628	1100 - 8556	GEM Port # Available for use. GEM# = x00 + ONU#, where x = 11, 14 … 83 in steps of 3, and with ONU = 1 to 256. The GEM base “x00” value can b e used to identify a service type that is used commonly across similar ONUs (e.g. 1100 for Data service; 1400 for Voice; 1700 for Video; e.g. GEM 1707 = Video on ONU 7).
	-	8557 - 8959	Reserved

1- 3 MANAGEMENT CARDS

The management controller cards provide management access to the MXK-F14xx and the MXK-F219.

- *MXK-F14xx Management Controller Card* on page 21
- *MXK-F219 Management Controller Card* on page 22

1- 3.1 MXK-F14xx Management Controller Card

The MXK-F14xx management cards support:

- User Interfaces: CLI and ZMS (the ZMS GUI application is external to the MXK-F14xx but interacts with the internal management processor)
- Northbound Interfaces
  - File Transfer Protocol (FTP) RFC 959
  - Secure File Transfer Protocol (SFTP) RFC 2228
  - Simple Network Management Protocol (SNMPv2c,v3), RFC 3411–RFC 3418
  - HTTP / HTTPS
  - Telnet
  - SSH
  - ZMS Database Synchronization
- Database
  - MIB
  - Relational Database
  - Unified Services Provisioning for Residential Gateway Support
- Physical Interfaces
  - Ethernet and serial interfaces for out-of-band management
  - CLOCK input port for TI/E1 or BITS support
  - Time Of Day (TOD) interface to provide a time of day signal
  - Pulse Per Second (PPS) interface to provide a once per second signal

1- 3.2 MXK-F219 Management Controller Card

The MXK-F219 management cards provide:

- User Interfaces, CLI and ZMS
- Northbound Interfaces
  - File Transfer Protocol (FTP) RFC 959
  - Secure File Transfer Protocol (SFTP) RFC 2228
  - Simple Network Management Protocol (SNMPv2c,v3), RFC 3411–RFC 3418
  - HTTP / HTTPS
  - Telnet
  - SSH
  - ZMS Database Synchronization
- Database
  - MIB
  - Relational Database
  - Unified Services Provisioning for Residential Gateway Support
- Physical Interfaces
  - Two 10G SFP+ AE NNI ports
  - Ethernet and serial interfaces for out-of-band management
  - CLOCK input port for TI/E1 or BITS support
  - Time Of Day (TOD) interface to provide a time of day signal
  - Pulse Per Second (PPS) interface to provide a once per second signal

CHAPTER 2 SYSTEM & CARD ADMIN

This chapter explains the system-wide and card administration functions. Administration is another word for management, but is used here to mean “highest-level” or “initial” management functions. System Administration Access deals with Management port setup, login and session logging. Card administration deals with setting up cards so that they are enabled, managed and (if needed) paired with another card for redundancy.

2- 1 SYSTEM ADMINISTRATION ACCESS

The MXK-F supports local and remote management. The first login must be done using CLI commands on the local, Serial, RS232 (hereafter called Craft/Console) Port.

The Craft/Console port can be used to set up other in-band or out-of-band management interfaces that either use CLI commands or an IP-based GUI interface (in-band = using a port that mixes management and subscriber packets; out-of-band = management packets only):

- Management Port options
  - Local, out-of-band, Craft/Console and MGMT Ethernet port
  - In-band on an uplink Ethernet port
- Management applications
  - CLI commands with a Serial/Telnet/SSH (e.g. PuTTY) shell interface

    Out-of-band management on the Craft/Console or MGMT port (*Out-of-band (Local) Management Ports* on page 29)

    In-band management on an Ethernet uplink port (*Uplink Port In-band IPoB Management Interfaces* on page 33)
  - IP-based GUI applications: Zhone Management System (ZMS) or Web-UI using out-of-band or in-band Ethernet ports. See *IP-based Management System Applications* on page 45

Figure 3 shows the MXK-F14xx out-of-band Craft/Console and MGMT management ports. MXK-F14xx in-band management interfaces can be set up using Fabric card uplinks (fabric cards are depicted in Figure 1 on page 17).

Figure 3: Ports Available for MXK-F Management for the MXK-F14xx

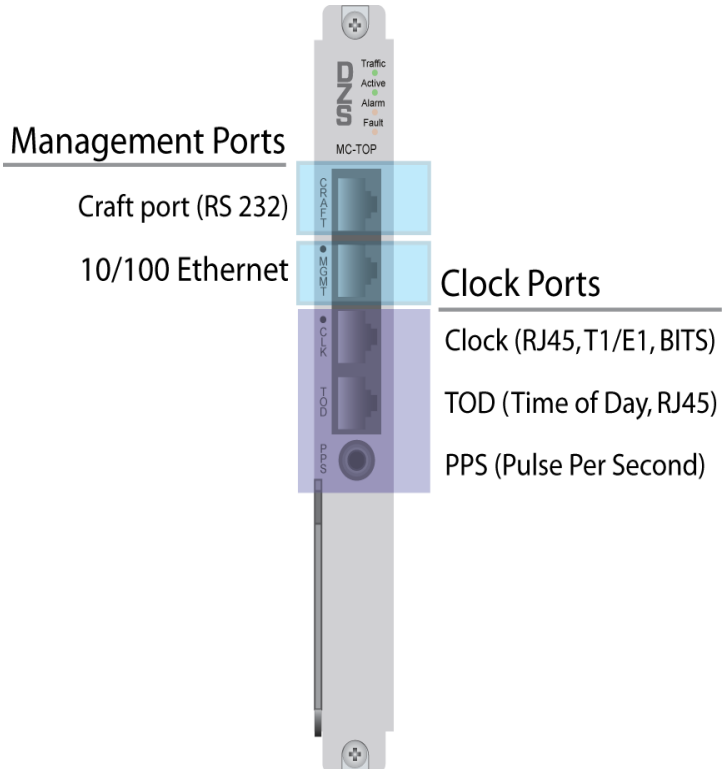
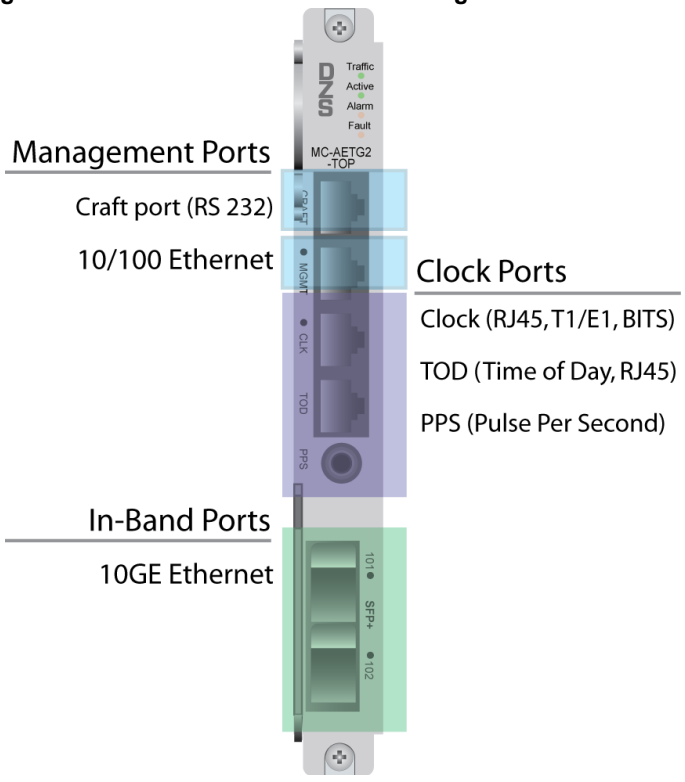


Figure 4 shows the MXK-F219 out-of-band Craft/Console and MGMT ports and the uplink ports that can be used for in-band management interfaces.

Figure 4: Ports Available for MXK-F Management for the MXK-F219



Note: The MXK-F can support up to six concurrent management sessions, five Telnet or SSH sessions and a single local session through the serial Craft/Console port.

The sections that follow cover the following topics:

2- 1.1 Admin/Management Access Default Configuration

The following defaults are used after the initial successful boot up.

- The default user name is **admin** and the password is **zhone**.
- The factory Default Configuration only includes the m1 management card. No other slot, card or port profiles are included in the Default Configuration file. Initially only the m1 card is enabled and will boot up.
- All other cards must be enabled in a **card-profile** before they will boot up (m2 management and line cards and on MXK-F14xx the a/b fabric cards).
- The Default Configuration file factory settings enable the Craft/Console management port. This is the only active port when a system is started up for the first time.
- A default **system 0** profile includes the following:
  - Authentication traps are not enabled
  - ZMS communication is not configured
  - Alarm notification and output are enabled for all severity levels

2- 1.2 CLI Login & Configure Admin/Management Interfaces

The MXK-F can be managed using a local port connection, or using an In-band interface on a network-facing, uplink port.

There are two local, out-of-band, management ports on each (m1/m2) management card: the Craft/Console port and the MGMT Ethernet port. The MGMT port requires an IP address. The Craft/Console port does not.

An in-band interface can be configured on an uplink port using the “IP on a Bridge” feature. This section explains how to:

2- 1.2.1 CLI Login/out, Response Dialog & Session Settings

The first MXK-F login must be done on the Craft/Console Port. Other management interfaces can only be used after they are enabled from the Craft/Console Port. All enabled management interfaces accept/use the same CLI system commands.

2- 1.2.1: 1 First Time Login

Use a serial port on a laptop or PC (e.g. DB9 connector) and connect it to the Craft/Console port of the MXK-F. To login for the first time, the CLI shell tool on the PC or laptop (e.g. Putty) must be setup to use the following settings (these are the MXK-F default settings).

- 57600 bps
- 8 data bits
- No parity
- 1 stop bit
- No flow control

The port speed can be configured to a different value (*Craft/Console Management Port on page 29*). If it is changed, then the PC/Laptop CLI shell setting must also be changed to equate to the value that was configured into the MXK-F.

Note: The default speed differs on the various DZS products (MXK = 9600; MXK-F108 = 9600; MXK-F multi-card chassis = 57600).

Note: Do not use the serial craft port of a standby card to modify its configuration.

2- 1.2.1: 2 Login and Logout

Procedure:

Log Into and Out of the System

When you first log in to the MXK-F, the default login is **admin** and the default password is **zhone**:

```
login:admin
password:
zSH>
```

To change the password, add users or delete users see *User Account Administration on page 67*.

To log out of the system, enter the **logout** command:

```
zSH> logout
```

Note: The maximum number of concurrent management sessions is five Telnet sessions and a single local session through the Craft/ Console (serial) port.

Tip: The system automatically logs you out after a period of inactivity. The default logout time is 10 minutes, but can be changed with the **timeout** command.

Procedure: *Change Automated Logout Time (Timeout)*

The system automatically logs you out after a period of inactivity. The default logout time is 10 minutes.

To change the Logout time use the timeout command with time in minutes:

```
zSH> timeout 120
CLI time-out value is now at 120 minutes.
```

To turn time-out off enter:

```
zSH> timeout off
CLI timer turned off.
```

To reset time-out to the default enter:

```
zSH> timeout -d
CLI time-out value reset to default of 10 minutes.
```

2- 1.2.1: 3 CLI Response Line Dialog

Procedure: *Change Max CLI Response Lines (Setline)*

The **setline** command sets the maximum lines to be displayed at once.

Entering the **setline** command without an argument displays the current number of lines per page.

```
zSH> setline
lines/page = 19
```

Use the **setline** command to set the number of lines displayed per page.

```
zSH> setline 50
cli lines per page changed to: 50
```

View the change.

```
zSH> setline
lines/page = 50
```

Use the **setline** command with "*0*" to set continuous scrolling.

```
zSH> setline 0
0 was entered, setting continuous scroll mode.
```

2- 1.2.1: 4 CLI Session Logs

The **log session** command enables/disables logging messages for that session only when connected to the device through a Telnet session. If the user logs out, the logging setting returns to the default. By default, log messages are enabled on the Craft/Console port. To enable/disable logging for the current session only enter:

```
zSH> log session on
Logging enabled.

zSH> log session off
Logging disabled.
```

This command setting does not persist across system reboots.

The **log serial** command enables/disables logging messages for the session. This command can be used in both Telnet connections and Craft/Console port connections to turn on and off the port logs. To enable/disable logging enter:

```
zSH> log serial on
Serial port logging enabled.

zSH> log serial off
Serial port logging disabled.
```

This command setting persists across system reboots and serial logging is *on* by default.

2- 1.2.2 Out-of-band (Local) Management Ports

2- 1.2.2: 1 Craft/Console Management Port

The Craft/Console port is a serial, RS-232 D compatible port that connects directly to the MXK-F internal management processor (CPU).

Only the Craft/Console port can be used to make the initial configuration settings and to enable the other management ports.

Note: The first time a system is powered up the m1 management card's Craft/Console port must be used to login and configure the system.

Note: Do not use the serial craft port of a standby card to modify its configuration.

Procedure:

Configuring the Craft/Console Management Port

The Craft/Console port settings can be changed by modifying the **rs232-profile**. The factory-default speed is described here: First Time Login, page 26. In general, the lower speed settings can support longer serial cables than the faster speed settings.

Normally, if the serial cable, from the Management PC to the MXK-F chassis, is shielded and less than 10 feet in length, then the fastest speed, 57600, can be used. In contrast, the 9600 speed setting can commonly accommodate a 50 foot shielded cable (some special, shielded, low capacitance cables can support up to 100 feet).

The limiting issue is actually not cable length, but rather is the capacitance of the cable if the cable is shielded, or can be noise if the cable is not shielded (RS232 rating = 2500pF; cable capacitance increases with length). There are too many types of cables, connectors and adapters to provide a simple explanation of what is possible. The most reliable setting is 9600, while the setting that allows for the fastest transfer of data (at a shorter distance) is 57600.

Caution: The Craft/Console port supports speeds of 9600, 19200, 38400, and 57600. Do not set the speed to an unsupported value. Doing so could render the Craft/Console port inaccessible.

Note: DZS recommends setting the port speed to 57600 on the MXK-F for faster CLI response times, as long as the cable length is not greater than 10 feet.

To change the **rs232-profile** port speed from 57600 to 9600 (if needed), enter:

The CLI syntax for the Craft/Console Port is:

1-m1-1-0/rs232 (or 1-m2-1-0/rs232)

```
zSH> update rs232-profile 1-m1-1-0/rs232
rs232-profile 1-m1-1-0/rs232
Please provide the following: [q]uit.
rs232PortInSpeed: -------> {57600}:9600
rs232PortOutSpeed: ------> {57600}:9600
rs232PortInFlowType: ----> {none}:
rs232PortOutFlowType: ---> {none}:
rs232AsyncPortBits: -----> {8}:
rs232AsyncPortStopBits: -> {one}:
rs232AsyncPortParity: ---> {none}:
rs232AsyncPortAutobaud: -> {disabled}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

Caution: When the MXK-F Craft/Console port speed is changed, then the management PC's serial port speed (e.g. PuTTY shell settings) must also be changed to match the newly configured MXK-F Craft/Console port speed.

Tip: If the management PC's serial port settings are not the same as the Craft/Console port (e.g. port speed), the connection will fail and CLI commands will be ignored. If another MXK-F management port has been enabled (e.g. MGMT port), that port can be used to read the MXK-F Craft/Console port settings.

To read the current **rs232-profile** enter:

```
zSH> get rs232-profile 1-m1-1-0/rs232
rs232-profile 1-m1-1-0/rs232
rs232PortInSpeed: -------> {9600}:
rs232PortOutSpeed: ------> {9600}:
rs232PortInFlowType: ----> {none}:
rs232PortOutFlowType: ---> {none}:
rs232AsyncPortBits: -----> {8}:
rs232AsyncPortStopBits: -> {one}:
rs232AsyncPortParity: ---> {none}:
rs232AsyncPortAutobaud: -> {disabled}:
```

2- 1.2.2: 2 MGMT Management Port

The MGMT, port has a default IP address = 192.168.10.1. In most applications this address should be changed to an IP address that is appropriate for the local IP address hierarchy where the MXK-F is located. To configure the management IP address use the following CLI command example on the Craft/Console port, but with an IP address that is appropriate for the subnet where the MXK-F is installed (instead of 10.50.1.35).

The CLI syntax for the MGMT Port is:

1-m1-1-0/eth (or 1-m2-1-0/eth)

```
zSH> interface add 1-m1-1-0/eth 10.50.1.35/24
Created ip-interface-record ethernetm-1/ip.
```

The m1 MGMT interface's **ip-interface-record** is **ethernetm-1** and is shared between the redundant MGMT ports on the m1/m2 management cards (when redundancy is enabled). The system can be reached using the IP address in the **ethernetm-1 ip-interface-record** no matter which card is active.

After an IP address has been assigned, a PC can be connected to the MGMT port to locally manage the MXK-F system.

Note: IPv4 is required for all IP interfaces on the MXK-F, including management interfaces. IPv6 is not supported as an IP interface on the MXK-F.

Caution: The MGMT port must be configured before any other interfaces on the system, even if you do not intend to use the MGMT port.

Note: The MXK-F requires the following Reserved IP addresses internally:

Subnet	Host Address Range
10.1.1.0/30	10.1.1.0 to 10.1.1.3
127.30.0.1/32	127.30.0.1

Users must avoid assigning these IP addresses to other interfaces to prevent conflicting IP address issues.

Note: The MGMT IP address is reset (returned to factory default IP address) when a **set2default** is performed without the restore option.

2- 1.2.2: 2.1 Change MGMT Port IP Address

To change the MXK-F management port IP address (e.g. from its default value or if the equipment is moved or replaced).

Procedure: *Change MGMT Port IP Address*

1 To change the MGMT IP address, delete the default interface, then configure a new IP address.

```
zSH> interface delete 1/m1/1/0/ip
Delete complete

zSH> interface add 1-m1-1-0/eth 10.50.1.35/24
Created ip-interface-record ethernetm-1/ip.
```

When new MGMT IP address is changed, the IP address used by the attached management PC will no longer match so the IP address that is used by the management PC must also be changed to match the newly configured MGMT IP address.

2 Use the **interface show** command to verify that the Ethernet interface was configured correctly:

```
zSH> interface show
1 interface
Interface       Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip     UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
-------------------------------------------------------------------------------
```

2- 1.2.3 Uplink Port In-band IPoB Management Interfaces

The MXK-F can also be managed using an In-band Management Interface on an uplink port by creating an “IP on a Bridge” bridge (IPoB bridge). An In-band Management Interface plays the same role as an Out-of-band Craft/ Console or MGMT port (provides a connection to the MXK-F Management Processor to execute management commands).

An IPoB Bridge is made of two parts: an upstream facing bridge on an uplink port (facing the network core) and a internal downstream facing bridge interface (facing the local, MXK-F internal Management Processor).

An Asymmetric IPoB bridge is created with an uplink bridge and an IPoB interface. A Symmetric IPoB is created with a symmetric TLS-GW bridge on an uplink port and an IPoB TLS interface.

A Symmetric bridge is used when the service provider’s management system is interconnected on a TLS network (TLS = Transparent LAN Service; this is the DZS recommended approach). An Asymmetric bridge is used for management systems that are not on a TLS network. If an MXK-F is added to a network, use the existing management network interconnect approach.

A VLAN ID is assigned to create a Management VLAN that is shared by other management interfaces in the network (e.g. other MXK-Fs and/or one or more remote, management servers).

The MXK-F can support up to six IPoB Management Interfaces per chassis.

The CLI syntax for an IPoB interface is:

1-m1-6-0/ipob (or 1-m2-6-0/ipob)

Note: The MGMT interface must be configured before in-band management interfaces can be added to an MXK-F system.

Note: IPv4 is required for all IP termination on the MXK-F, including management interfaces. IPv6 is not supported for IP termination on the MXK-F.

This section describes:

2- 1.2.3: 1 In-band Management - Uplink (Asymmetric) IPoB Interface

To create an In-band Management interface with an Uplink bridge you first create an Uplink Bridge on an Uplink port and then an “IP on a Bridge” Interface to the internal Management Processor.

An error message is displayed if an **interface add** command is executed for an “IP on a Bridge” Interface before an Uplink (or TLS) bridge VLAN exists.

```
zSH> interface add 1-m1-6-0/ipobbridge vlan 3002 1 10.50.2.35/24
Error: Couldn't determine type of IPOBRIDGE!
       Create an 'uplink' or 'tls' bridge(s) first.
```

Procedure: *Create an In-band Management Bridge using an Uplink bridge*

1 Create an Uplink Bridge for the In-band Management Interface. The specified VLAN will become the “Management VLAN”:

– For MXK-F14xx

```
zSH> bridge add 1-a-2-0/eth uplink vlan 3002 tagged
Adding bridge on 1-a-2-0/eth
Created bridge-interface-record ethernet2-3002/bridge
Bridge-path has been added successfully

zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical          Bridge                 St Table Data
-------------------------------------------------------------------------------------------
upl             Tagged 3002  1/a/2/0/eth       ethernet2-3002/bridge  UP S VLAN 3002 default
1 Bridge Interfaces displayed
```

– For MXK-F219

```
zSH> bridge add 1-1-101-0/eth uplink vlan 3002 tagged
Adding bridge on 1-1-101-0/eth
Created bridge-interface-record ethernet1-101-3002/bridge
Bridge-path has been added successfully

zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN    Physical         Bridge                      St  Table Data
-------------------------------------------------------------------------------------------------
upl             Tagged 3002  1/1/101/0/eth    ethernet1-101-3002/bridge   UP S VLAN 3002 default
1 Bridge Interface displayed
```

Note: If **uplink** is not included in the **bridge add** command, then tls is assumed and the subsequent **interface add** command will create a TLS IPoB interface (instead of a Downlink IPoB).

2 To add an “IP on a Bridge” Interface to the MXK-F Management Processor the **interface add 1-m1-6-0 /ipobridge** command must be used with an IP address and the Management VLAN (for MXK-F14xx and 219).

Note: The system auto-generates a downstream-facing asymmetric IPoB interface (ipobdwn) to match the added upstream-facing uplink bridge (upl) on the uplink port.

```
zSH> interface add 1-m1-6-0/ipobridge vlan 3002 10.50.2.35/24 uplink
Created ip-interface-record ipobridge-3002/ip.
```

Verify the “IP on a Bridge” interface addresses:

```
zSH> interface show
2 interfaces
Interface      Status  Rd/Address           Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24      00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24      00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

The 1/m1/1/0/ip interface is the MGMT port that was configured here: Change MGMT Port IP Address, page 32.

The management card is now reachable through the uplink port using IP 10.50.2.35. Use the same steps to create an IPoB and bridges for downstream devices.

3 Verify the management bridge (*upl* and *ipobdwn*) and the auto-generated bridge path parameters:

– For MXK-F14xx

```
zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical            Bridge                        St Table Data
-------------------------------------------------------------------------------------------
upl             Tagged 3002 1/a/2/0/eth         ethernet2-3002/bridge         UP S VLAN 3002 default
ipobdwn         Tagged 3002 1/m1/6/0/ipobridge  ipobridge-3002/bridge         UP S 00:01:47:7f:e1:b2
                                                                                 S 10.50.2.35
2 Bridge Interfaces displayed

zSH> bridge-path show
VLAN/SLAN  Bridge                  Address
-------------------------------------------------------------------------------
     3002 ethernet2-3002/bridge    Default, Age:3600, MCAST Age:250, IGMP Query Interval:0,
IGMP DSCP:0, Flap Mode: Default, Block: Asym/Auto
     3002 ipobridge-3002/bridge    00:01:47:7f:e1:b2
     3002 ipobridge-3002/bridge    10.50.2.35
```

– For MXK-F219

```
zSH> bridge show
       Orig
Type    VLAN/SLAN  VLAN/SLAN   Physical            Bridge                        St  Table Data
-------------------------------------------------------------------------------------------------
upl               Tagged 3002  1/1/101/0/eth       ethernet1-101-3002/bridge  UP  S VLAN 3002 default
ipobdwn           Tagged 3002  1/m1/6/0/ipobridge  ipobridge-3002/bridge      UP  S 00:01:47:e6:2e:81
                                                                                   S 10.50.2.35
```

```
2 Bridge Interfaces displayed


zSH> bridge-path show
VLAN/SLAN  Bridge                          Address
--------------------------------------------------------------------------------
     3002  ethernet1-101-3002/bridge       Default, Age: 3600, MCAST Age: 250, IGMP Query Interval: 0,
IGMP DSCP: 0, Flap Mode: Default, Block: Asym/Auto
      3002 ipobridge-3002/bridge           00:01:47:7f:e1:b2
      3002 ipobridge-3002/bridge           10.50.2.35
```

4 To configure a Default Route see: Configuring a Default Route, page 39.

2- 1.2.3: 2 In-band Management - TLS-GW (Symmetric) IPoB Interface

To create an In-band Management interface with a TLS bridge you first create a TLS bridge on an uplink port and then a downstream-facing “IP on a Bridge” TLS Interface.

Procedure: *Create an In-band Management Bridge using a TLS Interface*

1 Create a TLS-GW Bridge for the In-band Management Interface. The specified VLAN will become the “Management VLAN”:

– For MXK-F14xx

```
zSH> bridge add 1-a-2-0/eth tls-gw vlan 3002 tagged
Adding bridge on 1-a-2-0/eth
Created bridge-interface-record ethernet2-3002/bridge
Bridge-path has been added successfully

zSH> bridge show
     Orig
Type VLAN/SLAN  VLAN/SLAN     Physical      Bridge                      St  Table Data
-------------------------------------------------------------------------------------------------------
tls-gw          Tagged 3002   1/a/2/0/eth   ethernet2-3002/bridge       UP  S VLAN 3002 default
1 Bridge Interfaces displayed
```

– For MXK-F219

```
zSH> bridge add 1-1-101-0/eth tls-gw vlan 3002 tagged
Adding bridge on 1-1-101-0/eth
Created bridge-interface-record ethernet1-101-3002/bridge
Bridge-path has been added successfully

zSH> bridge show
     Orig
Type VLAN/SLAN  VLAN/SLAN   Physical      Bridge                      St  Table Data
-------------------------------------------------------------------------------------------------------
tls-gw        Tagged 3002   1/1/101/0/eth   ethernet1-101-3002/bridge  UP  S VLAN 3002 default
1 Bridge Interfaces displayed
```

2 To add an “IP on a Bridge” Interface to the MXK-F Management Processor the **interface add 1-m1-6-0 /ipobridge** command must be used with an IP address and the Management VLAN (for MXK-F14xx and 219).

Note: The system auto-generates a downstream-facing symmetric IPoB interface (ipobtls) to match the added upstream-facing TLS-GW bridge (tls-gw) on the uplink port.

```
zSH> interface add 1-m1-6-0/ipobridge vlan 3002 10.50.2.35/24 tls
Created ip-interface-record ipobridge-3002/ip.
```

Verify the “IP on a Bridge” interface addresses:

```
zSH> interface show
```

```
2 interfaces
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24       00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

The management card is now reachable through the uplink port using IP 10.50.2.35. Use the same steps to create an IPoB and bridges for downstream devices.

3 Verify the Management Bridge (*tls-gw* and *ipobtls*) and the auto-generated bridge path parameters:

– For MXK-F14xx

```
zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical            Bridge                    St Table Data
-------------------------------------------------------------------------------------------
tls-gw           Tagged 3002  1/a/2/0/eth         ethernet2-3002/bridge   UP  D 00:01:47:00:06:c0
ipobtls          Tagged 3002  1/m1/6/0/ipobridge  ipobridge-3002/bridge   UP  S 00:01:47:7f:e1:b2
                                                                             S 10.50.2.35
2 Bridge Interfaces displayed

zSH> bridge-path show
VLAN/SLAN  Bridge                  Address
-------------------------------------------------------------------------------
     3002 ethernet2-3002/bridge    Default, Age:3600, MCAST Age:250, IGMP Query Interval:0,
IGMP DSCP:0, Flap Mode: Default, Block: Asym/Auto

     3002 ipobridge-3002/bridge              00:01:47:7f:e1:b2
     3002 ipobridge-3002/bridge              10.50.2.35
```

– For MXK-F219

```
zSH> bridge show
        Orig
Type    VLAN/SLAN  VLAN/SLAN   Physical            Bridge                     St  Table Data
-------------------------------------------------------------------------------------------------
tls-gw           Tagged 3002  1/1/101/0/eth       ethernet1-101-3002/bridge  UP  D 00:01:47:00:06:c0
ipobtls          Tagged 3002  1/m1/6/0/ipobridge  ipobridge-3002/bridge      UP  S 00:01:47:c6:15:f9
                                                                                S 10.50.2.35
2 Bridge Interfaces displayed

zSH> bridge-path show
VLAN/SLAN  Bridge                          Address
-------------------------------------------------------------------------------
     3002  ethernet1-101-3002/bridge       Default, Age: 3600, MCAST Age: 250, IGMP Query Interval: 0,
IGMP DSCP: 0, Flap Mode: Default, Block: Asym/Auto
     3002 ipobridge-3002/bridge            00:01:47:7f:e1:b2
     3002 ipobridge-3002/bridge            10.50.2.35
```

4 To configure a Default Route see: Configuring a Default Route, page 39.

2- 1.2.3: 3 Configuring a Default Route

A Default Route can be configured to provide a known upstream IP destination. With a configured Default Route the MXK-F is allowed to use dynamic routes, that are learned through the use of IP routing protocols, but fallback to the configured Default Route when the MXK-F does not know how to reach a destination (e.g. a centralized management server).

A Static Route is the opposite of a Dynamic (learned) Route. A Static Route is a configured (fixed) route that does not use Routing Protocols. A Default Route is a special type of Static Route, since it is a configured (persistent) parameter, but is used in conjunction with Dynamic Routing.

Procedure: *Create a Default Route*

The following example creates a Default Route that identifies the next hop (Default Gateway) and uses a "cost" of "1" (the cost is used by routing protocols to determine which route is the shortest/best route):

```
zSH> route add default 10.50.1.254 1
```

Procedure: *Verifying the Default Route (for MXK-F14xx and 219)*

```
    zSH> route show
Destination Routing Table
Dest                    Nexthop          Cost     Owner      Fallback
-------------------------------------------------------------------------------
0.0.0.0/0               10.50.1.254      1        STATICLOW
10.50.1.0/24            1/m1/1/0/ip      1        LOCAL
10.50.2.0/24            1/m1/6/0/ip      1        LOCAL
```

Use the **ping** command to verify connectivity to the Default Gateway:

```
zSH> ping 10.50.1.254
PING 10.55.1.254: 64 data bytes
!!!!!
----10.55.1.254 PING Statistics----
5 packets transmitted, 5 packets received
round-trip (ms)  min/avg/max = 0/1/5
```

To stop the ping, press any key.

2- 1.2.3: 4 In-band IPoB Management on a LinkAgg Group

Link Aggregation provides a method to combine multiple Ethernet ports into a LinkAgg Group to act as a single port (e.g. two 10GE ports act as a 20G port). A LinkAgg Group can be thought of as a physical port. Like physical ports, VLANs and IPoB interfaces can be added to LinkAgg Groups.

This section provides an example for how to add a management vlan and IPOB to a LinkAgg Group that already exists in case the idea of adding a management vlan and IPoB to a LinkAgg Group is confusing.

There are several LinkAgg applications and settings that are not "management" or "IPoB" specific, so LinkAgg is not explained in detail here. LinkAgg Groups are explained in the MXK-F Configuration Guide.

Since this example assumes an existing LinkAgg Group, the first step is to view the established LinkAgg group info, then add a management vlan and IPoB to the LinkAgg Group and then view the bridge and bridge-path info.

Procedure: *Add an In-band Management Bridge and IPoB to a LinkAgg Group*

1 View info about the existing LinkAgg Group.

– For MXK-F14xx

```
zSH> linkagg show
LinkAggregations:
slot unit ifName        partner: Sys            Pri     grp ID  status  agg mode
-------------------------------------------------------------------------------
 a    1   LAG1          00:26:51:cb:7c:c1       0x8000  0x8000   ACT    Active

           links         slot    port    subport         status
          ------------------------------------------------------------
          1-a-1-0         1       2       0              ACT
          1-a-2-0         1       3       0              ACT
```

– For MXK-F219

```
zSH> linkagg show
LinkAggregations:
slot unit ifName        partner: Sys            Pri     grp ID  status  agg mode
-------------------------------------------------------------------------------
 1    1   LAG1          00:26:51:cb:7c:c1       0x8000  0x8000   ACT    Active

         links          slot    port    subport         status
        ------------------------------------------------------------
         1-1-101-0       1       101     0               ACT
         1-1-102-0       1       102     0               ACT
```

In this example, the LinkAgg Group name is LAG1. The LinkAgg Group name is used (instead of a port name), when adding the management vlan. The uplink ports that are members of this group are in the "links" column.

2 Create an uplink (asymmetric) bridge for the Management VLAN.

```
zSH> bridge add LAG1/linkagg uplink vlan 3002 tagged
Adding bridge on LAG1/linkagg
Created bridge-interface-record LAG1-3002/bridge
```

```
Bridge-path has been added successfully
```

Note: To instead create a TLS vlan use **tls-gw** instead of **uplink** in the **bridge add** command.

3 To create an "IP on a Bridge" Interface to the MXK-F Management Processor use the **interface add 1-m1-6-0 /ipobridge** command with an IP address and Management VLAN. The system generates an asymmetric/downlink IPoB interface to match the uplink bridge.

```
zSH> interface add 1-m1-6-0/ipobridge vlan 3002 10.50.2.35/24
Created ip-interface-record ipobridge-3002/ip.

zSH> interface show
2 interfaces
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24       00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

4 Verify the bridge.

– For MXK-F14xx

```
zSH> bridge show
       Orig
Type    VLAN/SLAN  VLAN/SLAN   Physical              Bridge                  St  Table Data
-------------------------------------------------------------------------------------------------------
upl              Tagged 3002   1/a/1/0/linkagg       LAG1-3002/bridge        UP  S VLAN 3002 default
ipobdwn          Tagged 3002   1/m1/6/0/ipobridge    ipobridge-3002/bridge   UP  S 00:01:47:e6:2e:81
                                                                                 S 10.50.2.35
2 Bridge Interfaces displayed
```

– For MXK-F219

```
zSH> bridge show
       Orig
Type   VLAN/SLAN  VLAN/SLAN   Physical             Bridge                   St  Table Data
-------------------------------------------------------------------------------------------------------
upl            Tagged 3002   1/1/1/0/linkagg      LAG1-3002/bridge         UP  S VLAN 3002 default
ipobdwn        Tagged 3002   1/m1/6/0/ipobridge   ipobridge-3002/bridge    UP  S 00:01:47:e6:2e:81
                                                                                 S 10.50.2.35
2 Bridge Interfaces displayed
```

5 View the auto-generated bridge path parameters.

```
zSH> bridge-path show
VLAN/SLAN   Bridge               Address
-------------------------------------------------------------------------------
     3002 LAG1-3002/bridge       Default, Age: 3600, MCAST Age: 250, IGMP Query Interval: 0,
IGMP DSCP: 0, Flap Mode: Default, Block: Asym/Auto
     3002 ipobridge-3002/bridge  10.50.2.35
     3002 ipobridge-3002/bridge  00:01:47:e6:2e:81
```

2- 1.2.3: 5 Deleting IPoB Management Bridges

2- 1.2.3: 5.1 Delete an IPoB Management Bridge

This procedure is for a bridge on a single 1G/10G Ethernet uplink port (not on a LinkAgg group). The same steps are used for (Asymmetric) uplink/ downlink bridges and for (Symmetric) tls-gw/tls bridges.

Procedure: *Deleting an IPoB Management Interface*

1 Verify the interface.

```
zSH> interface show
2 interfaces
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24       00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

2 Delete the Management downstream facing “IP on a Bridge” Bridge Interface and its associated IP address.

```
zSH> interface delete 1-m1-6-0/ipobridge vlan 3002
Delete complete
```

3 Verify the interface was deleted. This example is for an asymmetric uplink/downlink bridge. The ipobdwn bridge interface has been deleted and the upl bridge type remains. If this was a symmetric TLS bridge, the (downstream) tls-gw type would be deleted and the *tls* type would remain.

– For MXK-F14xx

```
zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical           Bridge                      St Table Data
-------------------------------------------------------------------------------------------------
upl             Tagged 3002 1/a/2/0/eth       ethernet2-3002/bridge       UP  S VLAN 3002 default
1 Bridge Interface displayed
```

– For MXK-F219

```
zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical           Bridge                      St Table Data
-------------------------------------------------------------------------------------------------
upl           Tagged 3002   1/1/101/0/eth      ethernet1-101-3002/bridge   UP  S VLAN 3002 default
1 Bridge Interface displayed
```

4 Delete the Management upstream facing Bridge Interface.

– For MXK-F14xx

```
zSH> bridge delete ethernet2-3002/bridge vlan 3002
```

```
Bridge-path deleted successfully
ethernet2-3002/bridge delete complete
```

– For MXK-F219

```
zSH> bridge delete ethernet1-101-3002/bridge vlan 3002
Bridge-path deleted successfully
ethernet1-101-3002/bridge delete complete
```

2- 1.2.3: 5.2 Delete a LinkAgg IPoB Management Bridge

Use this procedure if the Management Bridge is on a 1G/10G Ethernet uplink port LinkAgg group. The same steps are used for (Asymmetric) uplink/ downlink bridges and for (Symmetric) tls-gw/tls bridges.

Procedure: *Deleting a LinkAgg IPoB Management Bridge*

1 Delete the Management "IP on a Bridge" downstream Bridge Interface:

```
zSH> interface delete 1-m1-6-0/ipobbridge vlan 3002
Delete complete
```

2 Verify the interface and bridge were deleted. This example is for an asymmetric uplink/downlink bridge (for a TLS bridge the upl type would be replaced with tls-gw).

```
zSH> interface show
1 interface
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
-------------------------------------------------------------------------------
```

– For MXK-F14xx

```
zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical          Bridge                   St Table Data
-------------------------------------------------------------------------------------------
upl             Tagged 3002  1/a/1/0/linkagg   LAG1-3002/bridge         UP
1 Bridge Interfaces displayed
```

– For MXK-F219

```
zSH> bridge show
      Orig
Type  VLAN/SLAN  VLAN/SLAN   Physical          Bridge                   St Table Data
-------------------------------------------------------------------------------------------
upl             Tagged 3002  1/1/1/0/linkagg   LAG1-3002/bridge         UP
1 Bridge Interfaces displayed
```

3 Delete the Management upstream facing LinkAgg Bridge Interface.

– For MXK-F14xx

```
zSH> bridge delete 1/a/1/0/linkagg vlan 3002
Bridge-path deleted successfully
linkagg-1-1-3002/bridge delete complete
```

– For MXK-F219

```
zSH> bridge delete 1/1/1/0/linkagg vlan 3002
Bridge-path deleted successfully
linkagg-1-1-3002/bridge delete complete
```

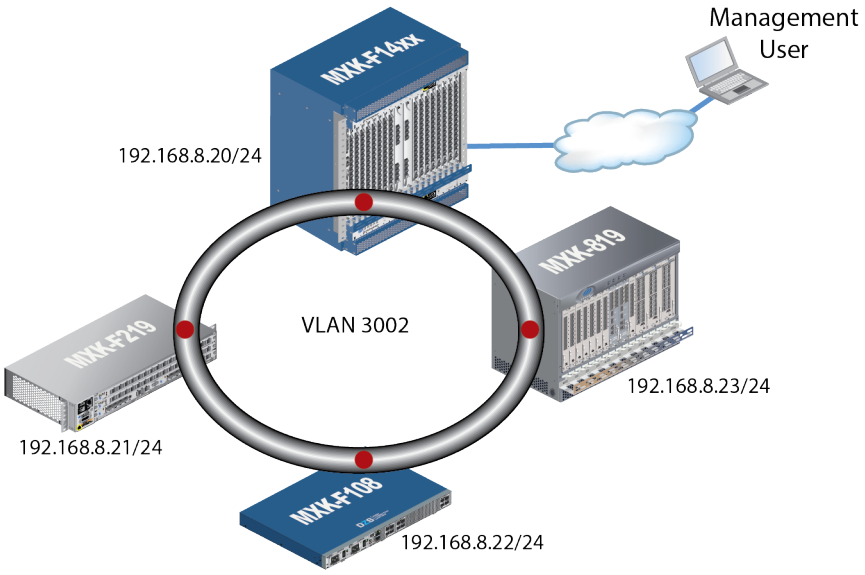
2- 1.2.3: 6 In-band IPoB Management on Multi-chassis Systems

The MXK-F is commonly installed as a stand alone system. It can, however, also be interconnected with other MXK/MXK-F/MXK-F108 devices in an EAPS Ring topology. An EAPS ring might be used because the network where they are installed uses Metro rings, or to increase reliability with Active and Standby paths, or to share a single connection to the network/core.

Figure 5 depicts a a ring with four MXK/MXK-F/MXK-F108 devices on a ring. Each device has its own IPoB bridge and IP address and all devices use the same management vlan (3002) to simplify the connection to the remote management server/controller.

The EAPS Ring application and how to setup management interfaces on the ring are explained in the MXK-F Configuration Guide in the EAPS section of the *MXK-F Configuration Guide*.

Figure 5: Ring interconnected chassis



2- 1.3 IP-based Management System Applications

Before using the Zhone Management System (ZMS), Web UI or any remote management, an In-band management interface must first be configured (see *Uplink Port In-band IPoB Management Interfaces* on page 33).

For OSS Gateway, refer to OSS Gateway documentation.

- Management Using ZMS, page 45
- Using the DZS Web UI, page 51

2- 1.3.1 Management Using ZMS

ZMS manages an MXK-F using SNMP, either SNMPv2 or SNMPv3.

The MXK-F **system 0** profile contains parameters that configure the MXK-F contact information and information to enable the ZMS interface.

Note: For details on using ZMS, refer to the *ZMS Administrator's Guide*, *ZMS Installation Guide* and the *NetHorizhon User's Guide*.

Either of the two following methods can be used to add an MXK-F to ZMS or to change the SNMP version that is used.

Procedure: *Enable ZMS Management of an MXK-F From the ZMS Interface*

At the ZMS interface:

– Add an MXK-F device to a ZMS server:

In ZMS, open the region, select the correct region, then right-click **Add Device**.

From the **SNMP Version** drop-down menu in the **Add Device Configuration** dialog box, select the SNMP version to agree with the setting in the MXK-F **system 0** profile (**SNMP V2** or **SNMP V3**). The factory default setting in an MXK-F is SNMPv2.

– Change the SNMP version that is used at the MXK-F and ZMS:

In ZMS, open the region, select the correct region, then right-click on the MXK-F device and bring up the **Modify** window. In the **Modify** window, select the **Identity** tab and change the SNMP version.

ZMS will change the parameters at the MXK-F and at ZMS.

Procedure: *Enable ZMS Management of an MXK-F Using the CLI and ZMS Interfaces*

This method provides the same results but requires entries at both the MXK-F and ZMS interfaces.

1 For MXK-F systems with an existing/running ZMS interface the interface must be disabled at the MXK-F and ZMS (otherwise jump to the next step): Start at the ZMS server by deleting the MXK-F device from ZMS. Then at the MXK-F disable the ZMS interface with system 0 profile zmsexists = *false*. (if zmsexists is already false, then leave it unchanged).

```
zSH> update system 0
system  0
Please provide the following: [q]uit.
syscontact: ----------------------> {}:
sysname: -------------------------> {}:
syslocation: ---------------------> {}:
enableauthtraps: -----------------> {disabled}:
setserialno: ---------------------> {0}:
zmsexists: -----------------------> {true}: false
zmsconnectionstatus: -------------> {inactive}:
zmsipaddress: --------------------> {0.0.0.0}:
configsyncexists: ----------------> {false}:
configsyncoverflow: --------------> {false}:
configsyncpriority: --------------> {high}:
configsyncaction: ----------------> {noaction}:
configsyncfilename: --------------> {}:
configsyncstatus: ----------------> {syncinitializing}:
configsyncuser: ------------------> {}:
configsyncpasswd: ----------------> {** private **}:  ** read-only **
numshelves: ----------------------> {1}:
shelvesarray: --------------------> {}:
numcards: ------------------------> {3}:
ipaddress: -----------------------> {0.0.0.0}:
alternateipaddress: --------------> {0.0.0.0}:
countryregion: -------------------> {us}:
primaryclocksource: --------------> {0/0/0/0/0}:
ringsource: ----------------------> {internalringsourcelabel}:
revertiveclocksource: ------------> {true}:
voicebandwidthcheck: -------------> {false}:
alarm-levels-enabled: ------------> {critical+major+minor+warning}:
userauthmode: --------------------> {local}:
radiusauthindex: -----------------> {0}:
secure: --------------------------> {disabled}:
webinterface: --------------------> {enabled}:
options: -------------------------> {NONE(0)}:
reservedVlanIdStart: -------------> {0}:
reservedVlanIdCount: -------------> {0}:
snmpVersion: ---------------------> {snmpv2}:
persistentLogging: ---------------> {disabled}:
outletTemperatureHighThreshold: --> {65}:
outletTemperatureLowThreshold: ---> {-12}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

2 To set the SNMP version and enable the interface at the MXK-F: Configure two **system 0** profile parameters. For SNMPv2 make zmsexists = *true* and snmpVersion = snmpv2. For SNMPv3 make zmsexists = *true* and snmpVersion = *snmpv3includingZMS*. If the SNMP version setting is already the desired setting then leave it unchanged. The following example is for SNMPv3.

```
zSH> update system 0
system  0
Please provide the following: [q]uit.
syscontact: ----------------------> {}:
sysname: -------------------------> {}:
syslocation: ---------------------> {}:
enableauthtraps: -----------------> {disabled}:
setserialno: ---------------------> {0}:
zmsexists: -----------------------> {false}: true
zmsconnectionstatus: -------------> {inactive}:
zmsipaddress: --------------------> {0.0.0.0}:
configsyncexists: ----------------> {false}:
configsyncoverflow: --------------> {false}:
configsyncpriority: --------------> {high}:
configsyncaction: ----------------> {noaction}:
configsyncfilename: --------------> {}:
configsyncstatus: ----------------> {syncinitializing}:
configsyncuser: ------------------> {}:
configsyncpasswd: ----------------> {** private **}:  ** read-only **
numshelves: ----------------------> {1}:
shelvesarray: --------------------> {}:
numcards: ------------------------> {3}:
ipaddress: -----------------------> {0.0.0.0}:
alternateipaddress: --------------> {0.0.0.0}:
countryregion: -------------------> {us}:
primaryclocksource: --------------> {0/0/0/0/0}:
ringsource: ----------------------> {internalringsourcelabel}:
revertiveclocksource: ------------> {true}:
voicebandwidthcheck: -------------> {false}:
alarm-levels-enabled: ------------> {critical+major+minor+warning}:
userauthmode: --------------------> {local}:
radiusauthindex: -----------------> {0}:
secure: --------------------------> {disabled}:
webinterface: --------------------> {enabled}:
options: -------------------------> {NONE(0)}:
reservedVlanIdStart: -------------> {0}:
reservedVlanIdCount: -------------> {0}:
snmpVersion: ---------------------> {snmpv2}: snmpv3includingZMS
persistentLogging: ---------------> {disabled}:
outletTemperatureHighThreshold: --> {65}:
outletTemperatureLowThreshold: ---> {-12}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

3 At the ZMS server add the MXK-F device.

In ZMS, open the region, select the correct region, then right-click **Add Device**.

From the **SNMP Version** drop-down menu in the **Add Device Configuration** dialog box, select the SNMP version: **SNMP V2** or SNMP V3.

2- 1.3.1: 1 Mass CLI Provisioning when Using ZMS

Procedure:

CLI provisioning and ZMS

To use a script to provision the MXK-F from the CLI when an MXK-F is managed by ZMS you can optionally disable the ZMS interface until you are finished using the CLI interface. The CLI interface can be used while ZMS is enabled, however, when ZMS is enabled, ZMS periodically auto-updates (refreshes/synchronizes) the configured parameters. In some cases, the ZMS auto-update may undo some configuration settings that have been implemented on the CLI. By disabling and later re-enabling ZMS, ZMS will not interrupt your CLI activities:

1 To disable ZMS. When the **zmsexists** parameter is set to *true*, update the **system 0** profile by changing the **zmsexists** parameter to *false* to disable partial config syncs to ZMS:

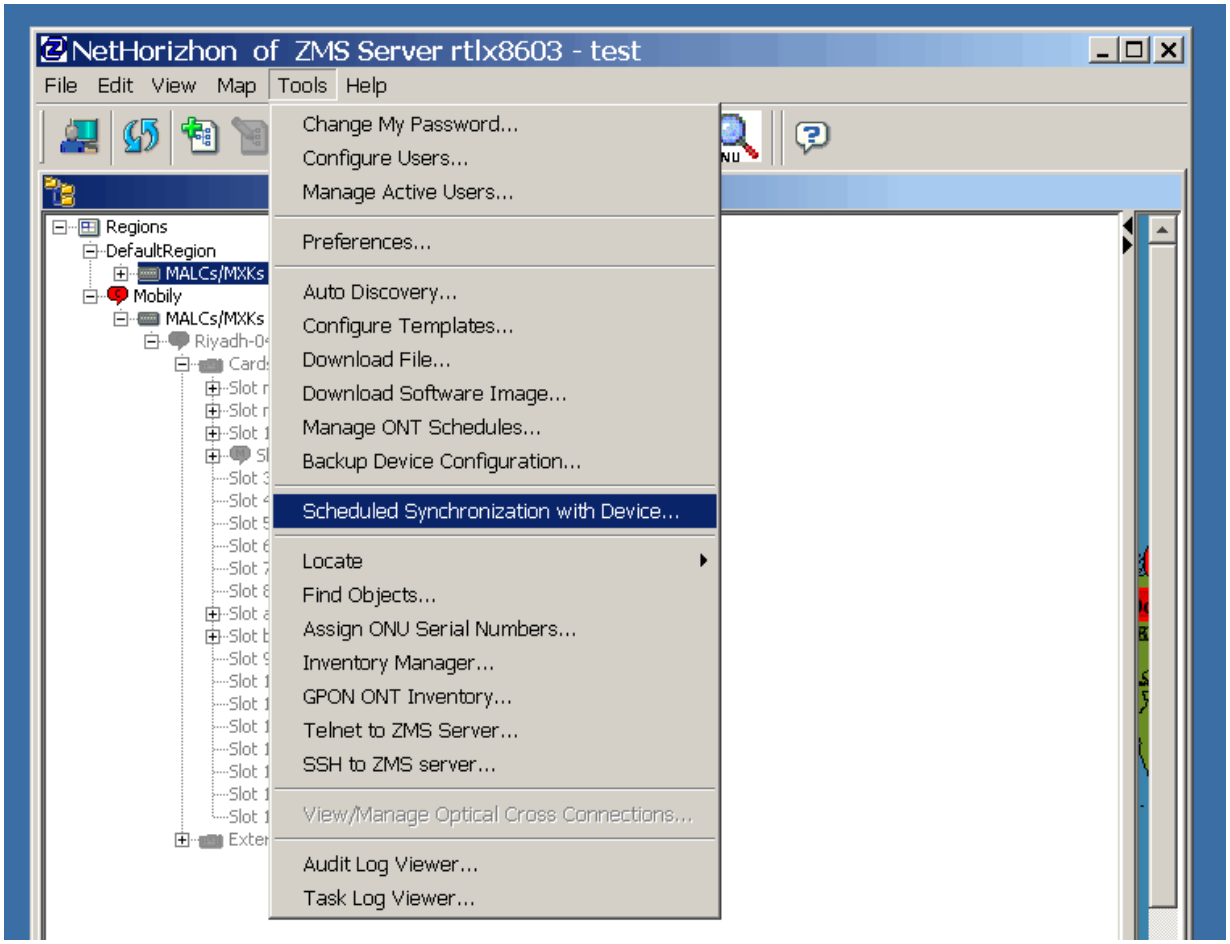
```
zSH> update system 0
system  0
syscontact: ---------------------->  {}
sysname: ------------------------->  {}
syslocation: --------------------->  {}
enableauthtraps: ----------------->  {disabled}
setserialno: --------------------->  {0}
zmsexists: ----------------------->  {true} false
zmsconnectionstatus: ------------->  {inactive}
zmsipaddress: -------------------->  {0.0.0.0}
configsyncexists: ---------------->  {false}
configsyncoverflow: -------------->  {false}
configsyncpriority: -------------->  {high}
configsyncaction: ---------------->  {noaction}
configsyncfilename: -------------->  {}
configsyncstatus: ---------------->  {synccomplete}
configsyncuser: ------------------>  {}
configsyncpasswd: ---------------->  ** private **
numshelves: ---------------------->  {1}
shelvesarray: -------------------->  {}
numcards: ------------------------>  {1}
ipaddress: ----------------------->  {192.168.10.1}
alternateipaddress: -------------->  {0.0.0.0}
countryregion: ------------------->  {us}
primaryclocksource: -------------->  {0/0/0/0/0}
ringsource: ---------------------->  {internalringsourcelabel}
revertiveclocksource: ------------>  {true}
voicebandwidthcheck: ------------->  {false}
alarm-levels-enabled: ------------>  {critical+major+minor+warning}
userauthmode: -------------------->  {local}:
radiusauthindex: ----------------->  {0}:
secure: -------------------------->  {disabled}
webinterface: -------------------->  {enabled}
options: ------------------------->  {NONE(0)}
reservedVlanIdStart: ------------->  {0}
reservedVlanIdCount: ------------->  {0}
snmpVersion: --------------------->  {snmpv2}
persistentLogging: --------------->  {disabled}
outletTemperatureHighThreshold: ->  {65}
outletTemperatureLowThreshold: -->  {-12}
```

```
tacacsauthindex: ----------------->  {0}:
maxIgmpResponseDelay: ----------->  {250}
specIgmpResponseDelay: ---------->  {1}
sshServerPort: ------------------>  {22}
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record created.
```

2 When the CLI provisioning is complete, update the **system 0** profile, this time by changing the **zmsexists** parameter to *true*.

```
zSH> update system 0
system  0
syscontact: ---------------------->  {}
sysname: ------------------------->  {}
syslocation: --------------------->  {}
enableauthtraps: ----------------->  {disabled}
setserialno: --------------------->  {0}
zmsexists: ----------------------->  {false} true
...
```

3 At the ZMS server, perform a full config sync with the MXK-F device.



Note: For details on using ZMS, refer to the *ZMS Administrator's Guide* and the *NetHorizhon User's Guide*.

Note: When using the CLI to configure an MXK-F that is managed by ZMS, an error may occur when ZMS is getting updated. The error blocks CLI commands for up to 10 seconds, after which the CLI commands must be re-issued.

2- 1.3.2 Using the DZS Web UI

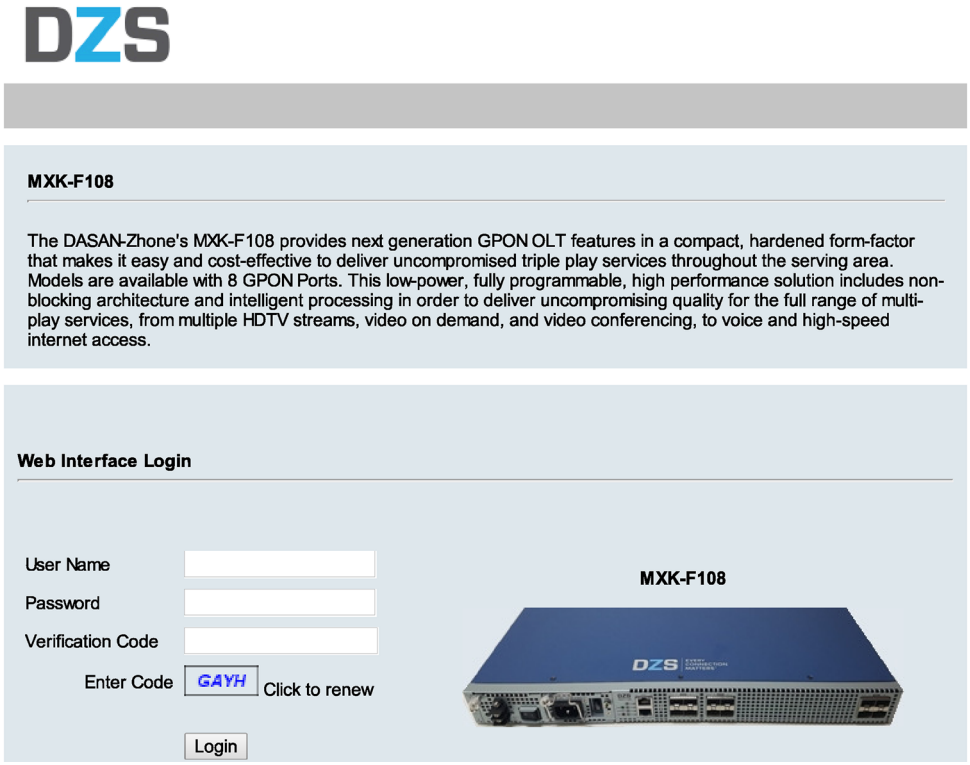
The MXK-F supports Web configuration using a DZS Web Graphical User Interface (GUI; a.k.a DZS Web UI, hereafter referred to as “Web UI”).

To use the Web UI to remotely manage an MXK-F, the local Craft/Console and MGMT ports and the remote In-band management properties must first be configured (see *Out-of-band (Local) Management Ports* on page 29 and *Uplink Port In-band IPoB Management Interfaces* on page 33).

To launch the Web UI, enter the IP address of the MXK-F on a browser URL address space. The Web UI launches and displays the Login window.

Note: Web UI only supports Internet Explorer and Mozilla browsers.

Figure 6: Web UI Login Screen



On the Login page, enter the user name, password and the verification code. The default user name is *admin* and the default password is *zhone*. The displayed verification code is only valid for approximately 30 seconds.

Click the desired menu to display the management options. For online help, click the Help icon or product title in any window.

Note: The **del** command can be used to delete all of the Web User Interface files if needed.

2- 2 CARD ADMINISTRATION

This section describes provisioning the management, fabric and line cards:

2- 2.1 Management Card (m1/m2) Provisioning for Redundancy

The MXK-F14xx and F219 chassis provide two slots, *m1* and *m2*, for redundant management cards that deliver the controller and database functions to the chassis. The default state of the management card in slot M1 is RUNNING and Active to allow access for admin/management access.

The management card in slot m2 can be provisioned to act as a redundant backup by configuring it to be in the same (redundancy) card group.

If both cards are installed and configured for redundancy, when power is turned on, by default, both come up in the RUNNING state, m1 is assigned as Active and m2 as Standby. If one card is not installed, fails or is disabled and the other is RUNNING, the RUNNING card will become Active. In other situations when both cards are RUNNING, the two management cards auto determine the Active/Standby roles based on the operational conditions.

The next two sub-sections explain how to provisions the management cards for redundancy.

2- 2.1.1 Management (m1/m2) Card Redundancy - MXK-F14xx

Procedure: *Provisioning Management Cards on the MXK-F for Redundancy Protection*

This procedure provisions the management card in the M2 slot for redundancy protection.

1 Enter the **slots** command to view the initial state of the management cards, fabric cards, and the line cards.

```
zSH> slots
MXK 1421
Management Cards
 m1:*MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
 m2: MXK-MC-TOP, 14U MGMT W/ TOP (NOT_PROV)
Fabric Cards
  a: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (NOT_PROV)
  b: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (NOT_PROV)
Line Cards
  3: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
  4: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
```

2 Provision the management card in slot m2 for redundancy.

The group ID of the management card in slot M1 and slot M2 must match. To find the Group ID, enter the **card add** <*card-profile-address*> **group** <*card-group-id*> command. The M1 group number in the following example is group *1*.

```
zSH> card add m2 group 1
new card-profile 1/m2/20001 added, sw-file-name "mxkmc.bin", 2 options:  card-group-id 1
```

If needed, to determine the group ID of the management card in slot M1 enter the **slots** <*slotNum*> command then the **get card-profile** <*profile-type*> command.

```
zSH> slots m1
MXK 1421
Type            :*MXK-MC-TOP, 14U MGMT W/ TOP
Card Version    : 800-03404-02-A
EEPROM Version  : 1
Serial #        : 7986440
CLEI Code       : No CLEI
Card-Profile ID : 1/m1/20001
Shelf           : 1
Slot            : m1
ROM Version     : MXK 3.1.1.141
Software Version: MXK 3.1.1.205
State           : RUNNING
Mode            : FUNCTIONAL
Heartbeat check : enabled
Heartbeat last  : WED JUN 17 17:17:02 2015
Heartbeat resp  : 60195
Heartbeat late  : 0
Hbeat seq error : 0
Hbeat longest   : 30
Fault reset     : enabled
Power fault mon : supported
Uptime          : 16 hours, 42 minutes

zSH> get card-profile 1/m1/20001
card-profile  1/m1/20001
sw-file-name: ------------> {mxkmc.bin}
admin-status: ------------> {operational}
upgrade-sw-file-name: ----> {}
upgrade-vers: ------------> {}
admin-status-enable: -----> {enable}
sw-upgrade-admin: --------> {reloadcurrrev}
sw-enable: ---------------> {true}
sw-upgrade-enable: -------> {false}
card-group-id: -------> {1}
hold-active: -------------> {false}
weight: ------------------> {nopreference}
card-line-type: ------> {unknowntype}
card-atm-configuration: --> {notapplicable}
card-line-voltage: -------> {not-used}
maxvpi-maxvci: -----------> {notapplicable}
card-init-string: --------> {}
wetting-current: ---------> {disabled}
pwe-timing-mode: ---------> {none}
```

3 Show the redundancy state.

```
zSH> slots
MXK 1421
Management Cards
 m1:*MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
 m2: MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
Fabric Cards
  a: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (NOT_PROV)
  b: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (NOT_PROV)
Line Cards
  3: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
  4: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)

zSH> showredundancy
Redundancy status for card 01:m1 - Safe, all services have redundant peers
01:m1 is active storage
01:m2 is standby storage

zSH> showredundancy -d
Redundancy status for card 01:m1 -
Taskname            Active Addr     Standby Addr   Stdby Ready?
========            ===========     ============   ============
InfoServer          01:m1:02        01:m2:02       Yes
RdsServer           01:m1:03        01:m2:03       Yes
tNumSrv             01:m1:1043      01:m2:1032     Yes
tShelfRR            01:m1:1044      01:m2:1033     Yes
tMAXTask            01:m1:1045      01:m2:1034     Yes
zCardRed            01:m1:26        01:m2:26       Yes
trapSrv             01:m1:25        01:m2:25       Yes
tFTD                01:m1:67        01:m2:67       Yes
TadSrvTask          01:m1:1047      01:m2:1036     Yes
ifcfgtask           01:m1:78        01:m2:78       Yes
L-RR-1/m1           01:m1:79        01:m2:79       Yes
LogServer           01:m1:08        01:m2:08       Yes
_RedSpawnSvrTask    01:m1:1052      01:m2:1039     Yes
gponOltMibHdlr      01:m1:1062      01:m2:1044     Yes
DhcpServerTask      01:m1:90        01:m2:90       Yes
bridgeMibHdlr       01:m1:1070      01:m2:1057     Yes
tEtherOamRp         01:m1:83        01:m2:83       Yes
RlyAlmHdlr          01:m1:1074      01:m2:1042     Yes
tIPSIM              01:m1:75        01:m2:75       Yes
Safe, all services have redundant peers
01:m1 is active storage
01:m2 is standby storage
```

2- 2.1.2 Management (m1/m2) Card Redundancy - MXK-F219

The MXK-F219 chassis also supports *m1* and *m2* management cards. However unlike the MXK-F14xx, the MXK-F219 m1/m2 cards must be properly combined with the two LC1/LC2, line cards to provide management card redundancy. All four cards are required for management card redundancy (m1, m2, LC 1 and LC2).

The line card in slot 1 is connected (paired) via hardware to the management card in the m1 slot for uplink port packet processing. Line card 2 is similarly connected (paired) to the management card in m2. The m1 card requires line card 1 and m2 requires line card 2.

Both line cards can be managed/controlled by either management card (m1 can control line card 1 and 2; m2 can control line card 1 and 2). Only one management card can be the Active chassis controller/database at a time.

The possible card configurations are:

M1, LC1 and optional LC2

M2, LC2 and optional LC1

M1, M2, LC1 and LC2 (if M1/M2 in same group = redundant management)

The following procedure explains how to enable management card redundancy.

Procedure: *Provisioning Management Cards on the MXK-F219 for Redundancy Protection*

1 Enter the **slots** command to view the initial state of the management cards and the line cards.

```
SH> slots
MXK 219
Management Cards
 m1:*MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP (RUNNING)
 m2: MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP card(NOT_PROV)
Line Cards
  1: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
  2: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
```

2 Provision the management card in slot M2 for redundancy.

To find the group ID of the card in slot M1 (if needed), enter the **slots** <*slotNum*> command to find the Card-Profile ID and then use that with the **get card-profile** <*profile-type*> command to find the card-group-id.

```
zSH> slots m1
MXK 219

Type            :*MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP
Card Version    : 800-03432-01-A
EEPROM Version  : 1
Serial #        : 15381158
CLEI Code       : No CLEI
```

```
Card-Profile ID : 1/m1/20002
Shelf           : 1
Slot            : m1
ROM Version     : MXK 3.1.2.101
Software Version: MXK 3.1.2.120
State           : RUNNING
Mode            : FUNCTIONAL
Heartbeat check : enabled
Heartbeat last  : WED SEP 21 20:41:23 2016
Heartbeat resp  : 1351
Heartbeat late  : 0
Hbeat seq error : 0
Hbeat longest   : 10
Fault reset     : enabled
Power fault mon : supported
Uptime          : 22 minutes
```

```
zSH> get card-profile 1/m1/20002
card-profile  1/m1/20002
sw-file-name: ------------> {mxkmc.bin}
admin-status: ------------> {operational}
upgrade-sw-file-name: ----> {}
upgrade-vers: ------------> {}
admin-status-enable: -----> {enable}
sw-upgrade-admin: --------> {reloadcurrrev}
sw-enable: ---------------> {true}
sw-upgrade-enable: -------> {false}
card-group-id: -----------> {1}
hold-active: -------------> {false}
weight: ------------------> {nopreference}
card-line-type: ----------> {unknowntype}
card-atm-configuration: --> {vbnrt65rt30}
card-line-voltage: -------> {not-used}
maxvpi-maxvci: -----------> {notapplicable}
card-init-string: --------> {}
wetting-current: ---------> {disabled}
pwe-timing-mode: ---------> {none}
```

The group ID of the management cards in slots M1 and M2 must match.

Enter the **card add** <*card-profile-address*> **group** <*card-group-id*> command and set the m2 card-group-id to match m1 (group *1*).

.

```
zSH>  card add m2 group 1
new card-profile 1/m2/20002 added, sw-file-name "mxkmc.bin", 1 option:  card-group-id 1
```

3 Show the redundancy state.

```
zSH> slots
MXK 219
Management Cards
 m1:*MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP (RUNNING)
 m2: MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP (RUNNING)
Line Cards
  1:*MXK-LC-GP16, LINE CARD W/ 16 GPON (RUNNING)
  2: MXK-LC-GP16, LINE CARD W/ 16 GPON (RUNNING)
```

```
zSH> showredundancy
Redundancy status for card 01:m1 - Safe, all services have redundant peers
01:m1 is active storage
01:m2 is standby storage
```

The “detail” (-d) command option shows if all of the process/services on a card are ready (available for redundancy; if one or more processes are not ready, repeat this command to see if/when all processes are ready).

```
SH> showredundancy -d
Redundancy status for card 01:m1 -
Taskname            Active Addr     Standby Addr   Stdby Ready?
========            ===========     ============   ============
InfoServer          01:m1:02        01:m2:02       Yes
RdsServer           01:m1:03        01:m2:03       Yes
tNumSrv             01:m1:1043      01:m2:1032     Yes
tShelfRR            01:m1:1044      01:m2:1033     Yes
tMAXTask            01:m1:1045      01:m2:1034     Yes
zCardRed            01:m1:26        01:m2:26       Yes
trapSrv             01:m1:25        01:m2:25       Yes
tFTD                01:m1:67        01:m2:67       Yes
TadSrvTask          01:m1:1047      01:m2:1036     Yes
L-RR-1/m1           01:m1:79        01:m2:79       Yes
LogServer           01:m1:08        01:m2:08       Yes
_RedSpawnSvrTask    01:m1:1052      01:m2:1040     Yes
ifcfgtask           01:m1:78        01:m2:78       Yes
gponOltMibHdlr      01:m1:1066      01:m2:1054     Yes
DhcpServerTask      01:m1:90        01:m2:90       Yes
RlyAlmHdlr          01:m1:1072      01:m2:1039     Yes
tIPSLM              01:m1:75        01:m2:75       Yes
tEtherOamRp         01:m1:83        01:m2:83       Yes
bridgeMibHdlr       01:m1:1081      01:m2:1061     Yes
tDS1RP              01:m1:1080      01:m2:1060     Yes
Safe, all services have redundant peers
01:m1 is active storage
01:m2 is standby storage
```

2- 2.2 MXK-F14xx Fabric Card (a/b) Provisioning

Slots *a* and *b* on the MXK-F14xx chassis provide slots for a pair of fabric cards that provide eight 10 GE Gigabit Ethernet interfaces with active/standby redundancy. Provision both cards to provide equipment protection. When both cards are provisioned, one card is in the Active state, the other card is in the Standby state.

Procedure: *Provisioning Fabric Cards in a Redundant Configuration*

1 Add the fabric card in slot *a*. The group ID must be 2.

```
zSH> card add a group 2
new card-profile 1/a/20104 added, sw-file-name "mxkfcae.bin", 1 option:  card-group-id 2
```

The card state changes to RUNNING.

```
zSH> slots
MXK 1421
Management Cards
 m1:*MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
 m2: MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
Fabric Cards
  a:*MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING)
  b: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (NOT_PROV)
Line Cards
  3: MXK-LC-GP16, LINE CARD W/ 16 GPON (RESET)
  4: MXK-LC-GP16, LINE CARD W/ 16 GPON (RESET)
```

2 Add the fabric card in slot *b*. The group ID must be 2 to match the card in slot *a*.

```
zSH> card add b group 2
new card-profile 1/b/20104 added, sw-file-name "mxkfcae.bin", 1 option:  card-group-id 2
```

The card state changes to RUNNING.

```
zSH> slots
MXK 1421
Management Cards
 m1:*MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
 m2: MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
Fabric Cards
  a:*MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING)
  b: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING)
Line Cards
  3: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
  4: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
```

2- 2.3 Line Card Provisioning

2- 2.3.1 Provision Line Cards for the MXK-F14xx

Procedure: *Provisioning Line Cards for the MXK-F14xx*

1 Enter the **slots** command to view the initial state of the line cards.

```
zSH> slots
MXK 1421
Management Cards
 m1:*MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
Fabric Cards
  a: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING)
  b: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING)
Line Cards
  3: MXK-LC-GP16, LINE CARD W/ 16 GPON (RESET)
  4: MXK-LC-GP16, LINE CARD W/ 16 GPON (RESET)
  9: MXK-LC-AEG32, LINE CARD W/ 32 1G AE (RESET)
```

2 Enter the **card add** *slot number* command to provision the line card.

```
zSH> card add 3
new card-profile 1/3/20201 added, sw-file-name "mxklcgp.bin"
zSH> card add 4
new card-profile 1/4/20201 added, sw-file-name "mxklcgp.bin"
zSH> card add 9
new card-profile 1/9/20222 added, sw-file-name "mxklcae.bin"
```

3 Enter the **slots** command to verify the state of the line cards.

```
zSH> slots
MXK 1421
Management Cards
 m1:*MXK-MC-TOP, 14U MGMT W/ TOP (RUNNING)
Fabric Cards
  a:*MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING+TRAFFIC)
  b: MXK-FC-AETG8, 14U FABRIC W/ 8x10G AE (RUNNING)
Line Cards
  3: MXK-LC-GP16, LINE CARD W/ 16 GPON (RUNNING)
  4: MXK-LC-GP16, LINE CARD W/ 16 GPON (RUNNING)
  9: MXK-LC-AEG32, LINE CARD W/ 32 1G AE (RUNNING)
```

2- 2.3.2 Provision Line Cards for the MXK-F219

Procedure: *Provisioning Line Cards for the MXK-F219*

1 Enter the **slots** command to view the initial state of the line cards.

```
zSH> slots
MXK 219
Management Cards
 m1:*MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP (RUNNING)
 m2: MXK-MC-AETG2-TOP, 2U MGMT W/ 2x10G AE, W/ TOP (RUNNING)
Line Cards
  1: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
```

```
  2: MXK-LC-GP16, LINE CARD W/ 16 GPON (NOT_PROV)
```

2 Enter the card add slot # command to provision the line cards.

```
SH> card add 1
new card-profile 1/1/20201 added, sw-file-name "mxklcgp.bin"

zSH> card add 2
new card-profile 1/2/20201 added, sw-file-name "mxklcgp.bin"
```

CHAPTER 3 CLOCKING

The MXK-F system timing can be derived from several timing inputs:

- Management card (m1/m2) “Local” (internal) Timing Source (crystal oscillator; default source that is used when no other source is available).
- Management card (m1/m2) CLK Port Timing input (*Management Card (m1/m2) CLK Port* on page 63).
  - Building Integrated Timing input (BITS; special cable required)
  - T1/E1 Recovered Timing
- UpLink Port Recovered Timing
  - Synchronous Ethernet - *SyncE Timing Input on page 64*

The term “timing” in this section is synonymous with “clock” (“timing input” = “clock input”). The timing source signals that are used are not actual clock signals, but are used to generate an internal clock signal (e.g. T1/BITS and Synchronous Ethernet are not clock signals).

This chapter is divided into the following sections:

3- 1 PROFILE SETTINGS FOR SYSTEM TIMING INPUTS

There are three configurable profiles that are associated with managing the system timing: **system 0**, **system-clock-profile**, and **ds1-profile**,

The MXK-F creates a **system-clock-profile** for each interface that can be used as a timing input (e.g. T1, BITS or SyncE). These profiles define which inputs are eligible to provide system timing and define the priority/weight that is assigned to each eligible timing input (weight = 1 to 10).

The **system 0** profile selects the primary clock input and is given the highest possible priority weight (11). The system is always synchronized to the primary input as long as its incoming signal is valid. If the primary clock input is not valid (e.g. failed), then the system-clock-profile weight settings determine which timing input is used (higher number = higher priority).

When an incoming T1 or BITS signal is connected to the CLK Port of an m1/m2 management card, the ds1-profile is used to specify looptiming (= CLK

Port acts as an incoming timing port). A special cable must be used to connect a BITS clock to the CLK Port.

With systems that have redundancy enabled, a timing input is only eligible for use if the card that the timing input is on is Active. Cards that are in the Standby mode or are not provisioned are not eligible timing inputs.

If you assign a weight to a timing input that is higher than the currently active timing source, or if you change the primary clock input setting in the **system 0** profile, the system will switch over to the new timing input.

Table 5 describes the configurable settings for the internal timing system.

Table 5: Clocking Parameters

Parameter	Description
transmit-clock-source (**ds1-profile**)	There are two clocking options for the m1/m2 CLK (DS1) ports ("throughtiming" is for non-MXK-F Line Card applications): Values: **looptiming** The (DS1 or BITS) recovered receive clock from the CLK Port can be used as an eligible system timing input. **localtiming** The CLK port can output a DS1 signal that is synchronized to the system clock (output to external equipment) **throughtiming** (non-MXK-F option) Default: **throughtiming**
primaryclocksource (**system 0** profile)	The *shelf-slot-port-subport/type* of an interface to provide clocking for the system. **Note:** If configured, the setting in the **primaryclocksource** parameter overrides settings in the **system-clock-profile** for all interfaces that provide clocking.
system-clock-eligibility (**system-clock-profile**)	Specifies whether the interface is eligible to provide clocking for the system. Values: **true** **false** Default: **false**
system-clock-weight (**system-clock-profile**)	Assigns a weight to the timing input. If you assign weight to an input that is higher than the currently active clock source, the system will switch over to the (new) higher priority timing input. Values: **1 to 10** 1 is the lowest priority, 10 is the highest Default: **5**

3- 2 CONFIGURE THE SYSTEM TIMING INPUTS

3- 2.1 Management Card (m1/m2) CLK Port

The internal interface names for the m1/m2 management card CLK ports are:

- 1-m1-1-0/ds1 and 1-m2-1-0/dsl

Procedure: *Management card (m1/m2) CLK Port Timing*

1. Verify that the interface that is to provide clock is up and active.
2. Verify the **transmit-clock-source** parameter in the **ds1-profile** is set to **looptiming** (required for this port to be an eligible timing input).

```
zSH> update ds1-profile 1-m1-1-0/ds1
ds1-profile  1-m1-1-0/ds1
Please provide the following: [q]uit.
line-type: ------------------------>  {esf}:
line-code: ------------------------>  {b8zs}:
send-code: ------------------------>  {sendnocode}:
circuit-id: ------------------------>  {ds1}:
loopback-config: ------------------->  {noloop}:
signal-mode: ----------------------->  {none}:
fdl: ------------------------------->  {fdlnone}:
dsx-line-length: ------------------->  {dsx0}:
line-status_change-trap-enable: --->  {enabled}:
channelization: -------------------->  {disabled}:
ds1-mode: -------------------------->  {csu}:
csu-line-length: ------------------->  {csu00}:
clock-source-eligible: ------------>  {eligible}:
transmit-clock-source: ------------>  {throughtiming}:looptiming
cell-scramble: --------------------->  {true}:
coset-polynomial: ------------------>  {true}:
protocol-emulation: ---------------->  {network}:
signal-type: ----------------------->  {loopstart}:
ds1-group-number: ------------------>  {0}:
line-power: ------------------------>  {disabled}:
timeslot-assignment: --------------->  {0):
transmit-clock-adaptive-quality: -->  {stratum3}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

3. In the **system-clock-profile**, enable the timing input and set the weight (10 = most preferred; 1 = least preferred):

```
zSH> update system-clock-profile 1-m1-1-0/ds1
system-clock-profile  1-m1-1-0/ds1
Please provide the following: [q]uit.
system-clock-eligibility: ->  {false}: true
system-clock-weight: ------>  {5}:modify the weight if necessary
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

```
zSH> clkmgrshow
Primary system clock is 1/m1/1/0 : T1
Secondary system clock is LOCAL timing
```

3- 2.2 View Available Timing Inputs

To view which timing/clock inputs exist use the clkmgrshow list command. The CLI response indicates the status of the CLK ports on the m1/m2 management cards and indicates whether any other timing available.

In this example only T1/E1 timing is present and enabled on one management card (the T1/E1 port on the m2 card is OOS = Out of Service).

```
zSH> clkmgrshow list
eligible list has 1 entry
   1 *     eligible  1/m1/1/0  ( 5) :       T1 :    Active : Loop
ineligible list has 1 entry
   1   not eligible  1/m2/1/0  ( 5) :       T1 :    OOS : Loop
pending list has 0 entries
```

3- 2.3 SyncE Timing Input

The Ethernet signal on an uplink port can be used as a timing input if the incoming Ethernet signal complies with the Synchronous Ethernet (SyncE) requirements that are specified in the ITU G.8262 standard.

A SyncE signal is the same as a standard Ethernet signal (e.g. 1GE or 10GE) except that the bit timing of the signal is highly accurate (the frequency accuracy of a “standard/normal” 1GE signal is +/- 100 ppm, parts per million, as compared to +/- 4.6 ppm for SyncE). Frequency timing is recovered from the bit rate of the Ethernet signal (e.g. ~10 Gbps).

The SyncE interface names are:

- For MXK-F14xx the uplinks are on the a/b Fabric cards

  1-a-x-0/eth or 1-b-x-0/eth (where “x” = the uplink port # on the fabric card faceplate)

- For MXK-F219 there are two uplinks on the m1/m2 management cards. Each management card is “paired” with a Line Card (m1 with LC1 and m2 with LC2). The “slot” number used in the name is the Line Card #.

  1-1-x-0/eth or 1-2-x-0/eth (where “x” = 101 or 102 for the two ports on the m1/m2 card faceplate)

Procedure: *Setting the system-clock-profile for SyncE*

1 View current clock.

```
zSH> clkmgrshow
All lines are using LOCAL clock
```

2 Update the **system-clock-profile** by setting **system-clock-eligibility** to *true* on the designated Ethernet port for SyncE.

– For MXK-F14xx

```
zSH> update system-clock-profile 1-a-2-0/eth
system-clock-profile  1-a-2-0/eth
Please provide the following: [q]uit.
system-clock-eligibility: ->  {false}: true
system-clock-weight: ------>  {5}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
FEB 10 10:02:39: warning: 1/a/1051: clkmgr: Secondary clock source set to 1/a/2/0/eth Record updated.
zSH> FEB 10 10:02:40: warning: 1/a/1051: clkmgr: System clock
source set to 1/a/2/0/eth
FEB 10 10:02:40: warning: 1/a/1051: clkmgr: There is no secondary clock
```

View the primary and secondary timing sources.

```
zSH> clkmgrshow
Primary system clock is 1/a/2/0 : ETHERNET Secondary system clock is LOCAL timing BITS clock is not
present
```

– For MXK-F219

```
zSH> update system-clock-profile  1-1-101-0/eth
system-clock-profile  1-1-101-0/eth
Please provide the following: [q]uit.
system-clock-eligibility: ->  {false}: true
system-clock-weight: ------>  {5}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
FEB 10 10:02:39: warning: 1/a/1051: clkmgr: Secondary clock source set to 1/m1/101/0/eth Record updated.
zSH> FEB 10 10:02:40: warning: 1/a/1051: clkmgr: System clock
source set to 1/m1/101/0/eth
FEB 10 10:02:40: warning: 1/a/1051: clkmgr: There is no secondary clock
```

View the primary and secondary timing sources.

```
zSH> clkmgrshow
Primary system clock is 1/m1/101/0 : ETHERNET Secondary system clock is LOCAL timing BITS clock is not
present
```

3- 2.4 Select Primary Timing Input

Procedure: *Configure the Primary Clock Input*

The primary clock input is selected by configuring the **system 0** profile primaryclocksource parameter with the name of a timing source interface. The Primary Clock Input has the highest priority level (11) and takes precedence over any **system-clock-profile** settings (priority range 1 to 10). It is not necessary to configure this **system 0** parameter. If it is not configured/changed, the timing system will select the timing source that is eligible and has the highest priority level. The following example shows how to set the m1 CLK Port (T1/BITS) as the primary clock input.

```
zSH> update system 0
system  0
syscontact: ---------------------->  {}
sysname: ------------------------->  {}
syslocation: --------------------->  {}
enableauthtraps: ----------------->  {disabled}
setserialno: --------------------->  {0}
zmsexists: ----------------------->  {true}
zmsconnectionstatus: ------------->  {inactive}
zmsipaddress: -------------------->  {0.0.0.0}
configsyncexists: ---------------->  {false}
configsyncoverflow: -------------->  {false}
configsyncpriority: -------------->  {high}
configsyncaction: ---------------->  {noaction}
configsyncfilename: -------------->  {}
configsyncstatus: ---------------->  {synccomplete}
configsyncuser: ------------------>  {}
configsyncpasswd: ---------------->  ** private **
numshelves: ---------------------->  {1}
shelvesarray: -------------------->  {}
numcards: ------------------------>  {1}
ipaddress: ----------------------->  {192.168.10.1}
alternateipaddress: -------------->  {0.0.0.0}
countryregion: ------------------->  {us}
primaryclocksource: -------------->  {0/0/0/0/0} 1-m1-1-0/ds1
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
```

CHAPTER 4 SYSTEM ADMINISTRATION

This chapter describes the MXK-F system administration functions:

4- 1 USER ACCOUNT ADMINISTRATION

MXK-F users have access to the CLI and are able to configure and administer the system.

4- 1.1 Add Users

Every administrative user on the system must have a user account. The account specifies their username and password, as well as their privilege level, which determines their access to commands.

Users with **admin** privileges have access to all the administrative commands. Users with **user** privileges have access to a very limited set of commands. The highest level of access is **useradmin**, which allows the creation of user accounts.

Note: When entering access level responses, enter **yes** completely or the CLI interprets the response as **no**.

To add a user, enter the following commands:

```
zSH> adduser
Please provide the following: [q]uit.
User Name: jjsmith
User Prompt[zSH>]:

Please select user access levels.
admin: ------->  {no}: yes
zhonedebug: -->  {no}:
voice: ------->  {no}:
data: -------->  {no}:
manuf: ------->  {no}:
database: ---->  {no}:
systems: ----->  {no}:
tool: -------->  {no}:
useradmin: --->  {no}: yes
..............................
User name:(jjsmith)  User prompt:(zSH>)
Access Levels:
(admin)(useradmin)
Save new account? [s]ave, [c]hange or [q]uit: s
User record saved.
TEMPORARY PASSWORD: hmj4mxFU
```

Commands with **zhonedebug** privilege levels are intended for use by DZS development only.

Immediately after activating the user account, you should change the password something you can remember, as explained in the next section.

4- 1.2 Create an SNMPv3 User from CLI

Procedure:

Creating an SNMP v3 User

1 Use the **adduser snmp** *username* command to create an SNMPv3 user. Select the Auth protocol and the Priv protocol, then enter a password if prompted.

For example:

```
zSH> adduser snmp test
Auth protocol (md5, sha, or none): md5
Enter auth password:
Confirm auth password:
Priv Protocol (des or none): des
Enter priv password:
Confirm priv password:
Enter access group (readwrite, readonly, encrypt, admin) : readwrite
```

2 Verify the user.

```
zSH> showuser snmp
```

```
userName                        auth   priv   accessGroup
--------                        ----   ----   -----------
zmsUser                          md5    des     readwrite
test                             md5    des     readwrite
```

4- 1.3 CLI User Command Privileges - ACL Groups

The **user-acl** commands are used to configure the CLI User Privilege Access Control List feature (a.k.a. CLI ACL Privileges; ACL = Access Control List). There are three user-acl command types: user-acl list, user-acl rule and user-acl user. These commands are used to: create one or more Access Lists (ACL user groups), add privilege rules for each Access List and add users to each Access List.

Users with "useradmin" privileges can manage ACL lists but cannot be members of ACL lists (i.e. the useradmin privilege can execute **user-acl** commands but cannot be assigned to an ACL List). Users without "useradmin" privileges can be members of ACL lists but cannot manage them. Use the **showuser** command to see a user's privilege categories.

Note: CLI users that are (remotely) authenticated using RADIUS or TACACS+ do not appear in the list generated by a showuser command, cannot be added to a CLI ACL Privilege group and cannot manage CLI ACL Privileges.

User Privilege ACL rules do not affect CLI users that have not been added to a User Privilege ACL group (they are regulated by standard CLI permission categories).

4- 1.3.1 CLI ACL Privilege - Access Lists (User Groups)

Each Access List has an Access List Name (list-name) and an Access List Number (list-index) that are configured using the user-acl list commands. The name and number are pseudonym/alternative identifiers for the list.

user-acl list

[**add** < list-name >]

[**modify** < list-index | list-name > **name** < new-name >]

[**delete** < list-index | list-name >]

[**show** < list-index | list-name | **all** >]

Table 6: user-acl list Command Option Explanations

Command Option	Description
add < list-name >	Create a name for a CLI ACL Privilege group (a user/ACL group name). This command also generates a "list-index" number.
modify < list-index | list-name > **name** < new-name >	Modify a name for a CLI ACL Privilege group.

Table 6: user-acl list Command Option Explanations

Command Option	Description
delete < list-index | list-name >	Delete a CLI ACL Privilege group
show < list-index | list-name | **all** >	Show name and number for one or all CLI ACL Privilege groups

1 Use the list add command to create two Access Lists: one with the name "bridge-mon," for users that are allowed to monitor forwarding bridges and one named "onu-mon" for users to monitor the status of ONUs. The MXK auto-assigns the next available list-index # (e.g. "1" and "'2").

```
zSH> user-acl list add bridge-mon
List "bridge-mon" has been created with index 1

zSH> user-acl list add onu-mon
List "bridge-mon" has been created with index 2
```

2 View the list-name and list-index

```
zSH> user-acl list show 1
  Index   List Name
======== ============================
      1   bridge-mon
      2   onu-mon

2 lists displayed.
```

The list-index # of each Access List determines the hierarchical use-order of the Lists. Smaller numbers take precedence over larger numbers. If a user is a member of more than one Access List it is possible that rules in one Access List contradict rules in the other List. If this is the case, the rules in the user's lower numbered Access List are used and any contradictory rules in the user's higher numbered Lists are ignored.

4- 1.3.2 CLI ACL Privilege - Access Rules

Access Rules are configured to specify which CLI commands are allowed and which are denied for each Access List. When a rule is created for an Access List, the system auto-generates an "entry-id" # identifier for the rule.

user-acl rule

[**add** < list-index | list-name > **access** < **permit** | **deny** >

command < command type >]

[**insert** < list-index | list-name/entry-id > **access** < **permit** | **deny** >

command < command type >]

[**modify** < list-index/entry-id > <<**access** < **permit** | **deny** >> |

< **command** < command type >>>]

[**modify** < list-index/entry-id > **access** < **permit** | **deny** >
command < command type >]

[**delete** < list-index/entry-index >]

[**show** < < < list-index | list-name > [/ < entry-index > > | < **all** > >]

[**check** < user-name > < command type >]

Table 7: user-acl rule Command Option Explanations

Command Option	Description
add < list-index | list-name > **access** < **permit** | **deny** > **command** < command type >	Add a rule to an ACL group that permits or denies the use of a specified command. This command also auto-generates a rule "index" number using the next available number.
insert < list-index | list-name/ entry-id > **access** < **permit** | **deny** > **command** < command type >	Insert a rule for an ACL group that permits or denies the use of a specified command. The new rule is inserted at the specified entry-id, and the rule numbers for existing rules with rule number >/= the specified entry-id are incremented by 1.
modify < list-index/entry-id > **access** < **permit** | **deny** > **command** < command type >	Modify a rule for an ACL group to change the access permit/deny option and/or the command type. To only change the access option do not include: **command** < command type >. To only change the command type do not include: **access** < **permit** | **deny** >. To change both include the entire command string.
delete < list-index/entry-index >	Delete an access privilege rule from an ACL group.
show < < list-index | list-name > [/ < entry-index >] > | < **all** >	Show one or all access privilege rules for an ACL group
check < user-name > < command type >	Verify whether a CLI user can access a command

The entry-id # (index #) of each rule determines the hierarchical use-order of the rules. Smaller numbers take precedence over larger numbers (i.e. a "/1" rule over-rides a larger numbered rule like "/2").

When optional arguments for a CLI command are not included in a configured rule, the optional arguments that are not specified are included in the rule by default (like a wild card).

1 Create rules to deny the use of **bridge add** and **bridge delete** from Access List 1. In this example the system auto-assigns the entry-ids "1" and "2." "List 1" can be identified with "1" or its name "bridge-mon."

```
zSH> user-acl rule add 1 access deny command bridge add
Rule 1/1 has been created in list "bridge-mon"

zSH> user-acl rule add bridge-mon access deny command bridge delete
Rule 1/2 has been created in list "bridge-mon"
```

2 Create a rule that allows Access List 1 to use the bridge command, without any of its command options. The meaning of this rule (if there are no other associated rules) is that all bridge command options are permitted (e.g. **bridge show**, **bridge stats**, etc.).

```
zSH> user-acl rule add bridge-mon access permit command bridge
Rule 1/3 has been created in list "bridge-mon"
```

Since rules /1 and /2 (in step 1 above) have smaller entry-id numbers, they take precedence over this rule /3. The result from these three rules is that the user has access to all bridge commands except the **bridge add** and **bridge delete** that are denied.

If instead the **bridge** command with no options was entered first and auto-assigned as rule /1, and the **bridge add** and **bridge delete** commands were next entered and assigned /2 and /3, then all bridge commands would be allowed and the "deny" rules for **bridge add** and **bridge delete** would be ignored (/1 takes precedence over /2 and /3).

3 Create a rule that allows Access List 2 to use the onu showall command using the wild card symbol "*."

The showall command is normally followed by the onu interface identifier and can then be followed by the parameter enabled or disabled. The first "*" in the example below allows the user to specify any onu interface. The second "*abled" means that the user is allowed to use the enabled and disabled command options.

```
zSH> user-acl rule add 2 access permit command onu showall * *abled
Rule 2/1 has been created in list "onu-mon"
```

4 View the rules that have been set up.

```
zSH> user-acl rule show all
Index             List Name             access            command
======== ============================== ====== ==============================
   1/1    bridge-mon                     deny   bridge add
   1/2    bridge-mon                     deny   bridge delete
   1/3    bridge-mon                     permit bridge
   2/1    onu-mon                        permit onu showall * *abled
4 rules displayed.
```

5 The following shows how a new rule can be inserted into an existing rule hierarchy (inserted with a specified rule #), instead of using the auto-generated rule number that places the new rule at the next available number (which is usually at the end of the rule hierarchy).

The specified number for the inserted rule must be a number that is currently used in the hierarchy. Lower priority rules are moved one step down to make room for the newly inserted rule (e.g. the permit bridge rule in this example is moved down the hierarchy from /3 to /4).

```
zSH> user-acl rule insert 1/3 access deny command bridge unblock
Re-indexing rules, please wait ....  done.
Rule 1/3 has been inserted in list "bridge-mon"
```

```
zSH> user-acl rule show all
Index             List Name             access            command
======== ============================== ====== ==============================
   1/1    bridge-mon                     deny   bridge add
   1/2    bridge-mon                     deny   bridge delete
   1/3    bridge-mon                     deny   bridge unblock
   1/4    bridge-mon                     permit bridge
   2/1    onu-mon                        permit onu showall * *abled
5 rules displayed.
```

6 A rule that already exists can be modified. The deny/permit option can be changed, and/or the command type can be changed (in this example both the deny/permit option and the command type are changed).

```
zSH> user-acl rule modify 1/3 access permit command bridge flush
Rule has been modified.

zSH> user-acl rule show all
Index             List Name             access            command
======== ============================== ====== ==============================
   1/1    bridge-mon                     deny   bridge add
   1/2    bridge-mon                     deny   bridge delete
   1/3    bridge-mon                     permit bridge flush
   1/4    bridge-mon                     permit bridge
   2/1    onu-mon                        permit onu showall * *abled
5 rules displayed.
```

7 When a rule is deleted, the rule and rule number are deleted together.

```
zSH> user-acl rule delete 1/3
Rule has been deleted.

zSH> user-acl rule show all
Index             List Name             access            command
======== ============================== ====== ==============================
   1/1    bridge-mon                     deny   bridge add
   1/2    bridge-mon                     deny   bridge delete
   1/4    bridge-mon                     permit bridge
   2/1    onu-mon                        permit onu showall * *abled
4 rules displayed.
```

8 As explained in a previous step, the rule add command auto-generates a rule number according to the next available rule number. If there is a skipped rule number (because of a deleted rule), then the auto-generated number begins by using a skipped number before continuing at the bottom of the number list. In the previous step, rule number /3 was deleted so if another rule is added, the auto-generator will begin by using bridge-mon rule /3 and subsequently use /5 if more rules are later added

```
zSH> user-acl rule add 1 access deny command bridge modify
Rule 1/3 has been created in list "bridge-mon"

zSH> user-acl rule show all
Index             List Name             access            command
======== ============================== ====== ==============================
   1/1    bridge-mon                     deny   bridge add
   1/2    bridge-mon                     deny   bridge delete
```

```
   1/3    bridge-mon                    deny   bridge modify
   1/4    bridge-mon                    permit bridge
   2/1    onu-mon                       permit onu showall * *abled
5 rules displayed.
```

The rule insert command can only be used to replace a number that exists in the rule hierarchy, so it is not used in this example. The /3 rule does not exist, so the rule add command is used.

Rules 1/1 through 1/4 shown in this rule show all could be used to allow a group of CLI users to monitor bridge forwarding functions (e.g. performance and status) and not allow them to change the configured settings of the bridges (add/delete/modify).

4- 1.3.3 CLI ACL Privilege - Configure CLI Users to User Groups

A user can be added to a CLI ACL Privilege group if the user list generated by a showuser command includes the user and if the user does not have the "useradmin" privileges. A user can be added to more than one Access List. When a user is added to an Access List the user is only allowed to execute CLI commands that are permitted by the Access Lists that the user is associated with (i.e. any command that is not in a user's ACL rules is implicitly denied).

user-acl user

[**add** < list-index | list-name > < user-name >]

[**delete** < list-index | list-name > < user-name >]

[**show** < **all** | < **list** < list-index | list-name >> | < **user** <user-name >>>]

Table 8: user-acl user Command Option Explanations

Command Option	Description
add < list-index | list-name > < user-name >	Add an existing CLI user (user-name) to a CLI ACL Privilege group.
delete < list-index | list-name > < user-name >	Delete a user from a CLI ACL Privilege group. This does not delete the CLI user account, only the user's association with the specified CLI ACL group.
show < **all** | < **list** < list-index | list-name >> | < **user** <user-name >>>	Show one or all CLI users in a CLI ACL Privilege group.

1 Verify Chris and Meredith are valid CLI users and do not have useradmin privilege.

```
zSH> showuser
.................................
User name:(Supervisor)  User prompt:(zSH>)
Access Levels:
(admin)(voice)(data)(manuf)(database)(systems)(tool)(useradmin)
```

```
..............................
User name:(Chris)  User prompt:(zSH>)
Access Levels:
(admin)

..............................
User name:(Meredith)  User prompt:(zSH>)
Access Levels:
(systems)
```

2 Add Chris to Access List 1 and 2 and Meredith to Access List 2.

```
zSH> user-acl user add 1 Chris
User "Chris" added to list "bridge-mon".

zSH> user-acl user add 2 Chris
User "Chris" added to list "onu-mon".

zSH> user-acl user add 2 Meredith
User "Meredith" added to list "onu-mon".

zSH> user-acl user show all
List Name                      User Name
============================== ==============================
bridge-mon                     Chris
onu-mon                        Chris
onu-mon                        Meredith
3 users displayed.
```

The user named “Supervisor” cannot be added to any CLI User Access List, because “Supervisor” has “useradmin” privileges.

3 The check command can be used to see if a user has access to a particular command.

```
zSH> user-acl rule check Chris bridge
Permitted by ACL list "bridge-mon" (1), rule 4.
```

Chris can use the generic bridge command, but the bridge add and other commands may be blocked by higher priority rules (like rule 1 below).

```
zSH> user-acl rule check Chris bridge add
Error: Permission denied: ACL list bridge-mon (1), rule 1
```

Meredith is a member of a CLI ACL Privilege group, but not of the bridge-mon group. So Meredith cannot use the bridge commands.

```
zSH> user-acl rule check Meredith bridge
Error: Permission denied: This command is not on any CLI Access Control List.
```

4 The following shows an error message that would be displayed if Chris was logged in and attempted to use the bridge add command. The CLI dialog below assumes the CLI login has been changed to Chris’s login

```
zSH> bridge add 1-2-5-0/eth downlink vlan 80 tagged
Error: Permission denied: ACL list monitor (1), rule 1
```

4- 1.4 Change Default User Passwords

When adding users, the system automatically assigns a temporary password to each user. Most users will want to change their password. The **changepass** command changes the password for the current logged in user. The following is an example of changing a password:

```
zSH> changepass
Current Password:
New Password:
Confirm New Password:
Password change successful.
```

4- 1.5 Delete Users

To delete a user, enter the **deleteuser** command and specify the username:

```
zSH> deleteuser jsmith
OK to delete this account? [yes] or [no]: yes
```

4- 1.6 Delete the useradmin Account

In addition to deleting regular user accounts, you can also delete the **useradmin** account. This account is automatically created by the system and provides full access to the CLI.

Note: You cannot delete the **admin** account (or any other user account with **useradmin** privileges) if you are currently logged into it.

To delete the **admin** account:

```
zSH> deleteuser admin
```

If desired, you can recreate an account named **admin** after deleting it:

```
zSH> adduser admin
Please provide the following: [q]uit.
User Name: admin
User Prompt[zSH>]:

Please select user access levels.
admin: ------->  {no}: yes
zhonedebug: -->  {no}:
voice: ------->  {no}: yes
data: -------->  {no}: yes
manuf: ------->  {no}: yes
database: ---->  {no}: yes
systems: ----->  {no}: yes
tool: -------->  {no}: yes
useradmin: --->  {no}: yes
................................
User name:(admin)  User prompt:(zSH>)
Access Levels:
```

```
(admin) (voice) (data) (manuf) (database) (systems) (tools) (useradmin)
Save new account? [s]ave, [c]hange or [q]uit: s
User record saved.
TEMPORARY PASSWORD: hmj4mxFU
```

4- 1.7 Reset Passwords

If a user forgets their password, an administrative user can reset the password and generate a new one using the **resetpass** command, as in the following example:

```
zSH> resetpass jsmith
Password:
```

4- 2 FILE NAVIGATION SYSTEM

This section describes the MXK-F file system and includes:

- Access the MXK-F File System, page 77

4- 2.1 Access the MXK-F File System

Use the following commands to access the MXK-F file system on the management card:

- **cd** Changes directory.
- **dir** Lists the contents of the directory.
- **pwd** Displays the current working directory.
- **image** Verifies software images and downloads software images on the flash to system memory.

The management card flash memory contains DOS file system that stores the system boot code, software images, and the configuration. During system startup, the software images on the flash are decompressed and loaded into memory.

Use the **cd**, **dir**, and **pwd** commands to list the contents of the file system, as in the following example:

Change directory.

```
zSH> cd /card1
```

Print the working directory.

```
zSH> pwd
/card1
```

List the directories in the current directory.

```
zSH> dir
Listing Directory .:
```

```
-rwxrwxrwx  1 0       0             9628125 Feb  3  2015 mxkmc.bin
-rwxrwxrwx  1 0       0            14022129 Feb  2  2015 mxklcgp.bin
-rwxrwxrwx  1 0       0            12950526 Feb  2  2015 mxkfcae.bin
drwxrwxrwx  1 0       0               16384 Jan  1  1980 crash/
drwxrwxrwx  1 0       0               16384 Jan  1  1980 onreboot/
drwxrwxrwx  1 0       0               16384 Jan  1  1980 datastor/
drwxrwxrwx  1 0       0               16384 Jan 29  2015 log/
drwxrwxrwx  1 0       0               16384 Feb  3  2015 bulkstats/
drwxrwxrwx  1 0       0               16384 Jan  1  1980 pub/
-rwxrwxrwx  1 0       0               11784 Jan 14  2015 2426A-me.txt
-rwxrwxrwx  1 0       0               11784 Jan 14  2015 2426a-me
-rwxrwxrwx  1 0       0               33650 Jan 14  2015 5114-me
-rwxrwxrwx  1 0       0             8388668 Jan 24  2015 mxkfcrom.bin
-rwxrwxrwx  1 0       0             8388668 Feb  3  2015 mxkmcrom.bin
-rwxrwxrwx  1 0       0             8388668 Jan 24  2015 mxkrom.bin
-rwxrwxrwx  1 0       0             7401476 Jan 20  2015 51xx-3.4.2.272c
                                  879516068 bytes available
```

4- 3 MONITOR THE MXK-F SYSTEM WITH SYSLOGS

This section provides the following information on how syslogs work on the MXK-F.

4- 3.1 Overview

Logging enables administrators to monitor system events by generating system messages. It sends these messages to:

- A temporary management session (either on the Craft/Console port or over a Telnet session)
- Log modules to create permanent log files
- A syslog server (optional)

The type of information sent in these messages can be configured using the **log** command. By default, the system sends the same type of information to all log message destinations. If you want to send different types of messages to the syslog daemon, use the **syslog** command.

4- 3.2 Default log store level

The default log store level is now set to emergency so by default the **log display** command displays only emergency level messages. Use the **log cache** command to display all messages.

4- 3.3 User login notification

Notifications of user login are sent to the console.

```
zSH> MAR 11 17:28:20: alert  : 1/a/1031: clitask1: User admin@172.16.48.232 logg
ed in on slot a
```

4- 3.4 Enable/disable syslog

By default, log messages are enabled on the Craft/Console port. Use the **log session** command and the **log serial** command to enable/disable logging:

The **log session** command enables/disables logging messages for that session only. If the user logs out, the logging setting returns to the default. To enable logging for the current session only:

```
zSH> log session on
Logging enabled.
```

To disable logging for the session:

```
zSH> log session off
Logging disabled.
```

The **log serial** command enables/disables logging messages for all sessions on the Craft/Console port. This setting persists across system reboots. To enable/disable logging for the Craft/Console port:

```
zSH> log serial on
Serial port logging enabled.
```

To disable logging for the serial port:

```
zSH> log serial off
Serial port logging disabled.
```

4- 3.5 Syslog message format

Syslog messages contain the following information:

Table 9: Default syslog message fields

Option	Description
Date	Date stamp of log message. Enabled by default.
Time	Time stamp of log message. Enabled by default.
Ticks	Current tick count. When the **tick** option is used, the date and time fields are not displayed.
Level	Logging level of the message. Enabled by default.
Address	The shelf and slot and application identifier causing the alarm.
Logtest	Log handle.
Taskname	Name of task that generated the log message. This is generally useful only for DZS development engineers. Enabled by default.
Function	Function that generated the log message.
Line	Line in code that generated the log message. This is generally useful only for DZS support staff.
Port	Port related to the log message.
Category	Category of the log message.
System	System related to the log message.
All	Controls all log message options.
Default	Controls the default log message options.
Message text	A description of the error that caused the alarm.

To change the information displayed in the log messages, use the **log option** command. First, display the available options:

```
zSH> log option
Usage: log option  < time          | 1 >  < on | off >
                   < date          | 2 >  < on | off >
                   < level         | 3 >  < on | off >
                   < taskname      | 4 >  < on | off >
                   < taskid        | 5 >  < on | off >
                   < file          | 6 >  < on | off >
                   < function      | 7 >  < on | off >
                   < line          | 8 >  < on | off >
                   < port          | 9 >  < on | off >
                   < category      | 10 > < on | off >
                   < system        | 11 > < on | off >
                   < ticks         | 12 > < on | off >
                   < stack         | 13 > < on | off >
                   < globalticks   | 14 > < on | off >
                   < all           | 14 > < on | off >
```

```
                < default       | 15 > < on | off >
options 'time' & 'date' supercede option 'ticks'
time: date: level: address: log: port: category: system:  (0x707)
```

Then, turn the option **on** or **off**. For example, the following command will turn the task ID on or off in log messages:

```
zSH> log option taskid on
time: date: level: address: log: taskid: port: category: system:  (0x717)

zSH> log option taskid off
time: date: level: address: log: port: category: system:  (0x707)
```

The following commands will turn on or off the tick count display in log messages:

```
zSH> log option ticks on
time: date: level: address: log: port: category: system: ticks:  (0xf07)

zSH> log option ticks off
time: date: level: address: log: port: category: system:  (0x707)
```

The following command will turn all options on in log messages:

```
zSH> log option all on
time: date: level: address: log: taskname: taskid: file: function: line: port: category: system: ticks:
stack: globalticks:  (0x3fff)
```

4- 3.6 Modify logging levels

To modify logging, use the **log** command. To modify syslog messages, use the **syslog** command.

Caution: Changing the log level may generate enough output to disrupt service.

To display the current levels for all logging modules, use the **log show** command:

```
zSH> log show
MODULE                                          LEVEL         STATUS
alarm_mgr                                       error         enabled
alarmconfigmibhdlr                              error         enabled
assert                                          error         enabled
attproxy                                        error         enabled
atttree                                         error         enabled
autocfg                                         error         enabled
bds                                             error         enabled
bds_client                                      error         enabled
bridgemib                                       error         enabled
bulkstats                                       error         enabled
bulkstatshdlr                                   error         enabled
cam                                             error         enabled
card                                            error         enabled
```

```
card_resource                                   error       enabled
carddeletehdlr                                  info        enabled
cardred                                         error       enabled
cardsvchdlr                                     error       enabled
cli                                             error       enabled
clkmgr                                          warning     enabled
cpecfg                                          error       enabled
cpemgr                                          error       enabled
...
```

Logging levels determine the number of messages that are displayed on the console. The higher the log level, the more messages are displayed. The MXK-F supports the following log levels:

- 1: emergency
- 2: alert
- 3: critical
- 4: error
- 5: warning
- 6: notice
- 7: information
- 8: debug

To change the log level, use the **log** *module* **level** command. For example, the following command changes the card module logging level to emergency:

Caution: Changing the log level may generate enough output to disrupt service.

```
zSH> log level card emergency
Module: card at level: emergency
```

To enable or disable log levels for a module, use the log enable or log disable commands. For example:

```
zSH> log disable card
Module: card is now disabled
```

4- 3.7 Non-persistent log messages

The **log cache** command displays the non-persistent log cache messages:

```
zSH> log cache
[1]: MAY 19 14:28:31: alert  : 1/a/1025: alarm_mgr: 01: a:06 Critical ETHERNET Down - Ethernet line down
[2]: MAY 19 14:30:19: alert  : 1/13/1025: alarm_mgr: 01:13:01 Major ETHERNET Up - Ethernet line up
[3]: MAY 19 14:32:12: alert  : 1/13/1025: alarm_mgr: 01:13:01 Major ETHERNET Down - Ethernet line down
[4]: MAY 19 14:32:26: alert  : 1/13/1025: alarm_mgr: 01:13:02 Major ETHERNET Up - Ethernet line up
[5]: MAY 19 14:33:27: alert  : 1/13/1025: alarm_mgr: 01:13:02 Major ETHERNET Down - Ethernet line down
[6]: MAY 19 14:36:23: alert  : 1/4/1025: alarm_mgr: 01: 4:01:01 Minor ONU Down
```

```
Line 1/4/1/1/gpononu CAUSE: inactive
[7]: MAY 19 14:36:32: alert  : 1/4/1025: alarm_mgr: 01: 4:01:01 Minor ONU Up
Line 1/4/1/1/gpononu CAUSE: active
[8]: MAY 19 14:36:53: critical: 1/a/1035: rebootserver:
* * * * Slot Reboot : type = 2, shelf = 1, slot = 4
[9]: JAN 01 00:00:11: error  : 1/4/9   : tnettask: Unable to find ifnet pointer for ifindex 0x2c0
[10]: JAN 01 00:00:11: error  : 1/4/9   : tnettask: Unable to find ifnet pointer for ifindex 0x2c1
[11]: JAN 01 00:00:12: error  : 1/4/9   : tnettask: Unable to find ifnet pointer for ifindex 0x2c2
[12]: MAY 19 14:40:32: notice : 1/a/12  : shelfctrl: Card in slot 4 changed state to RUNNING.
[14]: MAY 19 14:40:32: alert  : 1/4/1025: alarm_mgr: 01: 4:02 Critical OLT Up

Line 1/4/2/0/gponolt CAUSE: active
```

The **log cache max length** command sets the maximum number of log messages to store. The maximum log cache size is 2147483647, depending in the amount of memory available.

log cache max length

To change the current configured log cache size:

```
zSH> log cache max 200
Maximum number of log messages that can be saved: 200
```

The **log cache grep pattern** command searches through the log cache for the specified regular expression.

log cache grep pattern

The following example searches through the log cache for the string “Critical”:

```
zSH> log cache grep Critical
Searching for: "Critical"
[1]: AUG 02 22:37:19: alert  : 1/a/1025: alarm_mgr: 01: a:01 Critical ETHERNET Up - Ethernet line up
[2]: AUG 02 22:37:34: alert  : 1/a/1025: alarm_mgr: 01: a:02 Critical ETHERNET Down - Ethernet uplink
down
[3]: AUG 02 22:37:34: alert  : 1/a/1025: alarm_mgr: 01: a:03 Critical ETHERNET Down - Ethernet line down
```

The **log cache clear** command clears the log cache.

log cache clear

The **log cache size** command sets the maximum amount of memory for the log cache. Without options, displays the current log size.

```
zSH> log cache size
Number of log messages in the cache: 20
Total bytes used by the cache: 2052
```

The **log cache help** command displays the help information for the **log cache** command:

```
zSH> log cache help
Usage: log cache < max   > < length  >
                 < grep  > < pattern >
                 < clear >
```

```
                < size  >
                < help  >
With no arguments the 'log cache' command prints out all the
log messages currently in the cache.
The 'max' command is used to view/set the maximum number of
log messages that can be cached at one time.  If the cache is
full then the oldest log is discarded and the new log is
inserted.  If no value is given then the current setting is
displayed
The 'size' command is used to display the amount of memory
currently being used by the log cache.
The 'clear' command is used to erase the log cache.
The 'grep' command is used for searching the log cache for a
specific pattern.  Extended regular expressions are supported.
```

4- 3.8 Persistent log messages

Use the **log cache** command to view the persistent logs which only stores emergency level logs. For example:

```
zSH> log display
AUG 07 19:01:17: emergency: 1/a/12  : shelfctrl: Critical alarm set!
AUG 07 21:25:36: emergency: 1/a/12  : shelfctrl: Critical alarm set!
SEP 21 17:44:22: emergency: 1/a/12  : shelfctrl: Critical alarm set!
NOV 19 18:58:18: emergency: 1/a/12  : shelfctrl: Critical alarm set!
NOV 22 03:30:37: emergency: 1/a/12  : shelfctrl: Critical alarm set!
DEC 06 18:23:37: emergency: 1/a/12  : shelfctrl: Critical alarm set!
FEB 13 21:00:45: emergency: 1/a/12  : shelfctrl: Critical alarm set!
MAR 04 19:07:32: emergency: 1/a/12  : shelfctrl: Critical alarm set!
```

4- 3.9 Example log messages

This section provides examples of how to interpret log messages.

The following message appears when a card in the MXK chassis comes up or goes down.

The most important parts of the message are the date and time the event occurred, the shelf/slot of the event, and the message text. The remainder of the information is only useful for DZS development engineers.

For example:

```
MAR 11 17:46:20: alert  : 1/6/1025: alarm_mgr: 01: 6:01 Minor ETHERNET Down - Ethernet line down

MAR 11 17:46:21: alert  : 1/6/1025: alarm_mgr: 01: 6:01 Minor ETHERNET Up - Ethernet line up

MAR 11 17:48:30: alert  : 1/5/1025: alarm_mgr: 01: 5:03 Critical OLT Up
Line 1/5/3/0/gponolt CAUSE: active
```

4- 3.10 Log filter command

The **log filter** command is available as part of the log command functionality. This command enables users to show, set and delete log filters. Log filters limit the scope of log messages to a specific entity for troubleshooting and

diagnostics. When a log filter is set, the filter is assigned an index number and only messages relate the specified entity are displayed. Filters can be set for an specific ifindex, slot/port, VCL, or subscriber.

log filter

Restrict the display of log messages to only the log messages for a specified entity.

Syntax: **`log filter show | set (ifindex|port` *`slotport`*`|vcl` *`ifindex vpi vci`*`|subscriber` *`endpoint`*`)| delete`**

```
zSH> log filter set ifindex 12
New filter saved.

zSH> log filter set port 5 24
New filter saved.

zSH> log filter set subscriber 22
New filter saved.

zSH> log filter show
Index   Type            Filter Parameters
------  ------------    -----------------------------
 1      Port            slot=1, port=1
 2      Port            slot=1, port=4
 3      IfIndex         IfIndex=12
 4      Port            slot=5, port=24
 6      IfIndex         IfIndex=100
 7      IfIndex         IfIndex=104
 8      IfIndex         IfIndex=109
 9      IfIndex         IfIndex=103
10      IfIndex         IfIndex=107

zSH> log filter delete 10
Log filter 10 deleted
```

4- 3.11 Send messages to a syslog server

Table 10 describes the parameters in the **syslog-destination** profile you can modify to send messages to a syslog server.

Table 10: syslog-destination profile parameters

Parameter	Description
address	The IP address of the machine hosting the syslog server. Default: **0.0.0.0**
port	The UDP port to which the syslog messages will be sent. Default: **514**

Table 10: syslog-destination profile parameters (Continued)

Parameter	Description
facility	The syslog facility to which the syslog messages will be sent. Values: **local0** **local1** **local2** **local3** **local4** **local5** **local6** **local7** **no-map** Default: **local0**
severity	The severity level used to filter messages being set to the syslog server. Values: **emergency** **alert** **critical** **error** **warning** **notice** **info** **debug** Default: **debug**

```
zSH> new syslog-destination  1
Please provide the following: [q]uit.
address: -->  {0.0.0.0}: 192.200.42.5 IP address of the syslog server
port: ----->  {514}: leave at default
facility: ->  {local0}:
severity: ->  {debug}:
.................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

4- 3.12 Specify different log formats for system and syslog messages

Table 11 describes the **log-module** profile that supports the configuration of persistent log messages, syslog messages, and persistent storage levels by module. Modify this profile when you need to send different messages to admin sessions, the persistent logs, and the syslog server.

Table 11: log-module profile parameters

Parameter	Description
name	The name of the module whose logging is controlled by this profile. Default: **logtest**
display	Controls the display of messages on the system. Messages logged at this level and above will be displayed. Values: **emergency** **alert** **critical** **error** **warning** **notice** **info** **debug** Default: **error**
syslog	Controls the format of messages sent to the syslog server described in the **syslog-destination** profile. This field is similar to the display field, except for the **trackdisplay** value. Values: **emergency** **alert** **critical** **error** **warning** **notice** **info** **debug** **trackdisplay** Messages logged at, and above, the level set in the **display** parameter will also be recorded in the syslog server. Default: **trackdisplay**

Table 11: log-module profile parameters (Continued)

Parameter	Description
store	Controls the persistent storage of messages. This field is similar to the display field, except for the **trackdisplay** value. Values: **emergency** **alert** **critical** **error** **warning** **notice** **info** **debug** **trackdisplay** Messages logged at, and above, the level set in the **display** parameter will also be recorded in the syslog server. Default: **trackdisplay**

```
zSH> new log-module 1
Please provide the following: [q]uit.
name: ----> {logtest}: test1
display: -> {error}: warning
syslog: --> {trackdisplay}:
store: ---> {trackdisplay}:
.................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

In this case, the **log-module 1** will display to the screen, all messages at and above *warning*. The variable *trackdisplay* means that the same messages as defined in display are also sent to the syslog and storage. If different level of messages are needed for the different destinations, the variables for **display**, **syslog**, and **store** can be set at different levels.

4- 4 MONITOR THE MXK-F SYSTEM WITH CONSOLE LOGS

This section provides the following information on how to configure console logs on the MXK-F.

4- 4.1 Enable/ disable console logs

By default, console log is disabled. When the **consolelog on** command was given, anything displayed in the console is logged into the consolelog file. If the system is rebooted again, the console log setting returns to the default unless persistent log is enabled in the system profile.

Use the consolelog off command to turn off the console log.

4- 4.2 Display console logs and console log files history

Use the **consolelog display <filename>** command to view the contents for a consolelog file. These files are used for troubleshooting and system activity monitoring. The **consolelog display <*filename*>** where *filename* is either *consolelog1.txt*, *consolelog2.txt*. One of these files would be the current file which is receiving logging information.

The console log files are in /card1/log. The files *consolelog1.txt* and *consolelog2.txt* hold max 1 MB of console outputs each. Once the current file reaches 1 MB, the other file is renamed to old and a new *.txt* file becomes the current file. After a reboot, the *.txt* files are also saved as .old files.

4- 4.3 Persistent logging the console logs

By default, persistent logging is disabled. With persistent logging enabled in the **system** profile, the system's behavior is as if a **consolelog on** command was given and the consolelog on is persistently enabled after each reboot.

```
zSH> update system 0

system  0
Please provide the following: [q]uit.

persistentLogging: --------------> {disabled}: enabled
.....................

Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

4- 5 BASIC SYSTEM ADMINISTRATION COMMANDS

4- 5.1 Commands: new, list, show, get, update, delete

This section describes these commands:

4- 5.1.1 new Command

The **new** command creates new profiles, in this case a new GPON traffic profile.

```
zSH> new gpon-traffic-profile 1
gpon-traffic-profile  1
Please provide the following: [q]uit.
guaranteed-upstream-bw: ->  {0}:
traffic-class: ---------->  {ubr}:
compensated: ------------>  {false}:
shared: ----------------->  {false}:
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

4- 5.1.2 list Command

The **list** command displays all the profiles available on the MXK-F (partial list shown):

```
zSH> list
alarm-config: ifIndex
bridge-interface-record: ifIndex
bridge-mac-and-ip: ifIndex/recType/addrType/macAddr1/macAddr2/macAddr3/macAddr4/macAddr5/macAddr6/
vlanId/slanId/shelf/slot
bulk-statistic: index
bulk-statistics-config: index
card-profile: shelf/slot/cardType
community-access-profile: community
community-profile: community
cpe-auto-cfg-rule: index
cpe-cfg-global-settings: index
cpe-cond-dhcp-srv: listIndex/entryIndex
cpe-config-mgr: cpeConfigMgrIndex
cpe-config-mgr-dwnld-srvr: cpeCfgMgrDwnldSrvrIndex
cpe-config-mgr-member: ifIndex
cpe-config-mgr-member-stats: ifIndex
```

```
cpe-connection: ifIndex/tpType/tpIndex/vlanId/slanId
cpe-dhcp-server: index
cpe-dns-host: listIndex/entryIndex
cpe-dns-host-list: listIndex
```

The **list gpon-traffic-profile** command lists all GPON traffic profiles on the system.

```
zSH> list gpon-traffic-profile
gpon-traffic-profile  1
gpon-traffic-profile  2
gpon-traffic-profile  3
3 entries found.
```

The **list system** command displays the list of system profiles.

```
zSH> show system
syscontact:----------->   {260}
sysname:-------------->   {260}
syslocation:---------->   {260}
enableauthtraps:------>   enabled  disabled
setserialno:---------->   {0 - 2147483647}
zmsexists:------------>   true  false
zmsconnectionstatus:-->   active  inactive
zmsipaddress:--------->   {0 - 0}
configsyncexists:----->   true  false
configsyncoverflow:--->   true  false
configsyncpriority:--->   none  low  medium  high
configsyncaction:----->   noaction  createlist  createfulllist
configsyncfilename:--->   {68}
configsyncstatus:----->   synccomplete  syncpending  syncerror  syncinitializing
configsyncuser:------->   {36}
configsyncpasswd:----->   {36}
numshelves:----------->   {0 - 0}
shelvesarray:--------->   {36}
numcards:------------->   {0 - 0}
ipaddress:------------>   {0 - 0}
alternateipaddress:--->   {0 - 0}
countryregion:-------->   argentina  australia  belgium  china  costarica  finland  france  germany
hongkong  italy  japan  korea  mexico  netherlands  newzealand  singapore  spain  sweden  switzerland  uk
us  afghanistan  albania  algeria  americansamoa  andorra  angola  anguilla  antarctica  antiguabarbuda
armenia  aruba  austria  azerbaijan  bahamas  bahrain  bangladesh  barbados  belarus  belize  benin
bermuda  bhutan  bolivia  bosniaherzegovina  botswana  bouvetisland  brazil  britishindianoceanterritory
bruneidarussalam  bulgaria  burkinafaso  burundi  cambodia  cameroon  canada  capeverde  caymanislands
centralafricanrepublic  chad  chile  christmasisland  cocosislands  colombia  comoros  congo  cookislands
cotedivoire  croatia  cuba  cyprus  czechrepublic  denmark  djibouti  dominica  dominicanrepublic
easttimor  ecuador  egypt  elsalvador  equatorialguinea  eritrea  estonia  ethiopia  falklandislands
faroeislands  fiji  frenchguiana  frenchpolynesia  frenchsouthernterritories  gabon  gambia  georgia
ghana  gibraltar  greece  greenland  grenada  guadeloupe  guam  guatemala  guinea  guineabissau  guyana
haiti  heardislandmcdonaldislands  holysee  honduras  hungary  iceland  india  indonesia  iran  iraq
ireland  israel  jamaica  jordan  kazakstan  kenya  kiribati  northkorea  kuwait  kyrgyzstan  lao  latvia
lebanon  lesotho  liberia  libyanarabjamahiriya  liechtenstein  lithuania  luxembourg  macau  macedonia
madagascar  malawi  malaysia  maldives  mali  malta  marshallislands  martinique  mauritania  mauritius
mayotte  micronesia  moldova  monaco  mongolia  montserrat  morocco  mozambique  myanmar  namibia  nauru
nepal  netherlandsantilles  newcaledonia  nicaragua  niger  nigeria  niue  norfolkisland
northernmarianaislands  norway  oman  pakistan  palau  palestinianterritory  panama  papuanewguinea
paraguay  peru  philippines  pitcairn  poland  portugal  puertorico  qatar  reunion  romania  russia
rwanda  sainthelena  saintkittsnevis  saintlucia  saintpierremiquelon  saintvincentthegrenadines  samoa
```

```
sanmarino saotomeprincipe saudiarabia senegal seychelles sierraleone slovakia slovenia
solomonislands somalia southafrica southgeorgia srilanka sudan suriname svalbardjanmayen
swaziland syria taiwan tajikistan tanzania thailand togo tokelau tonga trinidadtobago tunisia
turkey turkmenistan turkscaicosislands uganda ukraine unitedarabemirates uruguay uzbekistan
vanuatu venezuela vietnam virginislandsuk virginislandsus wallisfutuna westernsahara yemen
yugoslavia zambia zimbabwe
primaryclocksource:--->   [Shelf {0-255}/Slot {0-31}/Port {0-500}/SubPort/Type] | [Name/Type]
ringsource:----------->   internalringsourcelabel  externalringsourcelabel
revertiveclocksource:->   true  false
voicebandwidthcheck:-->   true  false
alarm-levels-enabled:->   critical+major+minor+warning
userauthmode:--------->   local  radius  radiusthenlocal  radiusthencraft
radiusauthindex:------>   {0 - 2147483647}
secure:--------------->   enabled  disabled
webinterface:--------->   enabled  disabled
options:-------------->
cvlanonly+nol3bridgetable+ipg88bits+disdefpktrules+enablexcardlinkagg+fiberlan+cfmon
reservedVlanIdStart:-->   {0 - 4090}
reservedVlanIdCount:-->   {0 - 2048}
snmpVersion:---------->   snmpv2  snmpv3  snmpv3includingZMS
persistentLogging:---->   enabled  disabled
outletTemperatureHighThreshold:->   {35 - 65}
outletTemperatureLowThreshold:-->   {-40 - 0}
```

To view the card profiles existing on the system, enter **list card-profile**:

For example, on the MXK-F14xx:

```
zSH> list card-profile
card-profile  1/3/20201
card-profile  1/4/20201
card-profile  1/m1/20001
card-profile  1/m2/20001
card-profile  1/a/20104
card-profile  1/b/20104
6 entries found.
```

On the MXK-F219:

```
zSH> list card-profile
card-profile  1/1/20201
card-profile  1/2/20201
card-profile  1/m1/20002
card-profile  1/m2/20002
4 entries found.
```

To view the **bridge-interface-record** profiles of existing bridges enter **list bridge-interface-record**:

```
zSH> list bridge-interface-record
bridge-interface-record  1-0-0-3000-gponport-5/bridge
bridge-interface-record  1-0-0-3001-gponport-6/bridge
bridge-interface-record  ethernet2-3001/bridge
bridge-interface-record  ipobridge-3002/bridge
4 entries found.
```

4- 5.1.3 show Command

Use the **show** command to view all the parameters in a profile. For example, if you need to find which country codes are available on the MXK-F, use the **show system** command.

```
zSH> show system
syscontact:----------->   {260}
sysname:-------------->   {260}
syslocation:---------->   {260}
enableauthtraps:------>   enabled  disabled
setserialno:---------->   {0 - 2147483647}
zmsexists:------------>   true  false
zmsconnectionstatus:-->   active  inactive
zmsipaddress:--------->   {0 - 0}
configsyncexists:----->   true  false
configsyncoverflow:--->   true  false
configsyncpriority:--->   none  low  medium  high
configsyncaction:----->   noaction  createlist  createfulllist
configsyncfilename:--->   {68}
configsyncstatus:----->   synccomplete  syncpending  syncerror  syncinitializing
configsyncuser:------->   {36}
configsyncpasswd:----->   {36}
numshelves:----------->   {0 - 0}
shelvesarray:--------->   {36}
numcards:------------->   {0 - 0}
ipaddress:------------>   {0 - 0}
alternateipaddress:--->   {0 - 0}
countryregion:-------->   argentina  australia  belgium  china  costarica  finland  france  germany
hongkong  italy  japan  korea  mexico  netherlands  newzealand  singapore  spain  sweden  switzerland  uk
us  afghanistan  albania  algeria  americansamoa  andorra  angola  anguilla  antarctica  antiguabarbuda
armenia  aruba  austria  azerbaijan  bahamas  bahrain  bangladesh  barbados  belarus  belize  benin
bermuda  bhutan  bolivia  bosniaherzegovina  botswana  bouvetisland  brazil  britishindianoceanterritory
bruneidarussalam  bulgaria  burkinafaso  burundi  cambodia  cameroon  canada  capeverde  caymanislands
centralafricanrepublic  chad  chile  christmasisland  cocosislands  colombia  comoros  congo  cookislands
cotedivoire  croatia  cuba  cyprus  czechrepublic  denmark  djibouti  dominica  dominicanrepublic
easttimor  ecuador  egypt  elsalvador  equatorialguinea  eritrea  estonia  ethiopia  falklandislands
faroeislands  fiji  frenchguiana  frenchpolynesia  frenchsouthernterritories  gabon  gambia  georgia
ghana  gibraltar  greece  greenland  grenada  guadeloupe  guam  guatemala  guinea  guineabissau  guyana
haiti  heardislandmcdonaldislands  holysee  honduras  hungary  iceland  india  indonesia  iran  iraq
ireland  israel  jamaica  jordan  kazakstan  kenya  kiribati  northkorea  kuwait  kyrgyzstan  lao  latvia
lebanon  lesotho  liberia  libyanarabjamahiriya  liechtenstein  lithuania  luxembourg  macau  macedonia
madagascar  malawi  malaysia  maldives  mali  malta  marshallislands  martinique  mauritania  mauritius
mayotte  micronesia  moldova  monaco  mongolia  montserrat  morocco  mozambique  myanmar  namibia  nauru
nepal  netherlandsantilles  newcaledonia  nicaragua  niger  nigeria  niue  norfolkisland
northernmarianaislands  norway  oman  pakistan  palau  palestinianterritory  panama  papuanewguinea
paraguay  peru  philippines  pitcairn  poland  portugal  puertorico  qatar  reunion  romania  russia
rwanda  sainthelena  saintkittsnevis  saintlucia  saintpierremiquelon  saintvincentthegrenadines  samoa
sanmarino  saotomeprincipe  saudiarabia  senegal  seychelles  sierraleone  slovakia  slovenia
solomonislands  somalia  southafrica  southgeorgia  srilanka  sudan  suriname  svalbardjanmayen
swaziland  syria  taiwan  tajikistan  tanzania  thailand  togo  tokelau  tonga  trinidadtobago  tunisia
turkey  turkmenistan  turkscaicosislands  uganda  ukraine  unitedarabemirates  uruguay  uzbekistan
vanuatu  venezuela  vietnam  virginislandsuk  virginislandsus  wallisfutuna  westernsahara  yemen
yugoslavia  zambia  zimbabwe
primaryclocksource:--->   [Shelf {0-255}/Slot {0-31}/Port {0-500}/SubPort/Type] | [Name/Type]
ringsource:----------->   internalringsourcelabel  externalringsourcelabel
revertiveclocksource:->   true  false
voicebandwidthcheck:-->   true  false
```

```
alarm-levels-enabled:->   critical+major+minor+warning
userauthmode:--------->   local  radius  radiusthenlocal  radiusthencraft
radiusauthindex:------>   {0 - 2147483647}
secure:--------------->   enabled  disabled
webinterface:--------->   enabled  disabled
options:-------------->
cvlanonly+nol3bridgetable+ipg88bits+disdefpktrules+enablexcardlinkagg+fiberlan+cfmon
reservedVlanIdStart:-->   {0 - 4090}
reservedVlanIdCount:-->   {0 - 2048}
snmpVersion:---------->   snmpv2  snmpv3  snmpv3includingZMS
persistentLogging:---->   enabled  disabled
outletTemperatureHighThreshold:->   {35 - 65}
outletTemperatureLowThreshold:-->   {-40 - 0}
```

Use additional show commands such as **show bridge-interface-record** to view greater detail about bridges.

```
zSH> show bridge-interface-record
vpi:---------------------------------->   {0 - 4095}
vci:---------------------------------->   {0 - 65535}
vlanId:------------------------------->   {0 - 4090}
stripAndInsert:----------------------->   false  true
customARP:---------------------------->   false  true
filterBroadcast:---------------------->   false  true
learnIp:------------------------------>   false  true
learnUnicast:------------------------->   false  true
maxUnicast:--------------------------->   {0 - 2147483647}
learnMulticast:----------------------->   false  true
forwardToUnicast:--------------------->   false  true
forwardToMulticast:------------------->   false  true
forwardToDefault:--------------------->   false  true
bridgeIfCustomDHCP:------------------->   false  true
bridgeIfIngressPacketRuleGroupIndex:->   {0 - 2147483647}
vlanIdCOS:---------------------------->   {0 - 7}
outgoingCOSOption:-------------------->   disable  all
outgoingCOSValue:--------------------->   {0 - 7}
s-tagTPID:---------------------------->   {33024 - 37376}
s-tagId:------------------------------>   {0 - 4090}
s-tagStripAndInsert:------------------>   false  true
s-tagOutgoingCOSOption:--------------->   s-tagdisable  s-tagall
s-tagIdCOS:--------------------------->   {0 - 7}
s-tagOutgoingCOSValue:---------------->   {0 - 7}
mcastControlList:--------------------->   {264}
maxVideoStreams:---------------------->   {0 - 1024}
isPPPoA:------------------------------>   false  true
floodUnknown:------------------------->   false  true
floodMulticast:----------------------->   false  true
bridgeIfEgressPacketRuleGroupIndex:-->   {0 - 2147483647}
bridgeIfTableBasedFilter:------------>   none+mac+ip
bridgeIfDhcpLearn:-------------------->   none+mac+ip
mvrVlan:------------------------------>   {0 - 4090}
vlan-xlate-from:---------------------->   {0 - 4095}
slan-xlate-from:---------------------->   {0 - 4095}
bridge-type:-------------------------->  uplink  downlink  intralink  tls  rlink  pppoa  wire  mvr  user
downlinkvideo  downlinkdata  downlinkpppoe  downlinkp2p  downlinkvoice  downlinkupstreammcast  ipobtls
ipobuplink  ipobdownlink
incomingCOSOption:-------------------- >  disable  all
s_tagIncomingCOSOption:--------------- >  disable  all
```

4- 5.1.4 get Command

Use the **get** command to view the current configuration of profiles. The **get system 0** command displays information on the current MXK-F system configuration.

```
zSH> get system 0
system  0
syscontact: ----------------------> {}
sysname: -------------------------> {}
syslocation: ---------------------> {}
enableauthtraps: -----------------> {disabled}
setserialno: ---------------------> {0}
zmsexists: -----------------------> {false}
zmsconnectionstatus: -------------> {inactive}
zmsipaddress: --------------------> {0.0.0.0}
configsyncexists: ----------------> {false}
configsyncoverflow: --------------> {false}
configsyncpriority: --------------> {high}
configsyncaction: ----------------> {noaction}
configsyncfilename: --------------> {}
configsyncstatus: ----------------> {syncinitializing}
configsyncuser: ------------------> {}
configsyncpasswd: ----------------> ** private **
numshelves: ----------------------> {1}
shelvesarray: --------------------> {}
numcards: ------------------------> {3}
ipaddress: -----------------------> {0.0.0.0}
alternateipaddress: --------------> {0.0.0.0}
countryregion: -------------------> {us}
primaryclocksource: --------------> {0/0/0/0/0}
ringsource: ----------------------> {internalringsourcelabel}
revertiveclocksource: ------------> {true}
voicebandwidthcheck: -------------> {false}
alarm-levels-enabled: ------------> {critical+major+minor+warning}
userauthmode: --------------------> {local}
radiusauthindex: -----------------> {0}
secure: --------------------------> {disabled}
webinterface: --------------------> {enabled}
options: -------------------------> {NONE(0)}
reservedVlanIdStart: -------------> {0}
reservedVlanIdCount: -------------> {0}
snmpVersion: ---------------------> {snmpv2}
persistentLogging: ---------------> {disabled}
outletTemperatureHighThreshold: --> {65}
outletTemperatureLowThreshold: ---> {-12}
```

You can find the syscontact information, or whether the MXK-F is configured to communicate with the Zhone Management System (ZMS — zmsexists, zmsconnectionstatus, zmsipaddress).

4- 5.1.5 update Command

To update the **system 0** profile and all other profiles, use the **update** command.The **update system 0** command walks you through the profile to change specific fields.

Caution: You should be very careful when altering profiles. Where available you should use CLI macro commands.

For example:

```
zSH> update system 0
system  0
Please provide the following: [q]uit.
syscontact: ----------->  {}:
sysname: ------------->  {}:
syslocation: ---------->  {}:
enableauthtraps: ------>  {disabled}:
setserialno: ---------->  {0}:
zmsexists: ------------>  {true}: false
...
...
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

4- 5.1.6 delete Command

Use the **delete** command to delete profiles when needed.

```
zSH> delete gpon-traffic-profile 1
gpon-traffic-profile  1
1 entry found.
Delete gpon-traffic-profile 1? [y]es, [n]o, [q]uit : y
gpon-traffic-profile  1 deleted.
```

4- 5.2 Commands: interface show, bridge show

This section describes these commands:

- interface show Command, page 96
- bridge show Command, page 97

4- 5.2.1 interface show Command

The **interface show** command displays the numbered or unnumbered (floating) IP interfaces currently available on the MXK-F.

```
zSH> interface show
2 interfaces
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24       00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

Table 12: interface show Column

Column	Description
Interface	Shows the interface, the card and the physical port of the IP interface.

Table 12: interface show Column (Continued)

Column	Description
Status	Shows whether the interface is up or down.
Rd/Address	The IP address assigned to this gateway.
Media/Dest Address	Media/Dest Address is either the MAC address of a device.
IfName	The interface name.

4- 5.2.2 bridge show Command

The **bridge show** command displays the bridge interfaces on the MXK-F. Note that a bridge is a combination of bridge interfaces working together.

```
zSH> bridge show
        Orig
Type    VLAN/SLAN  VLAN/SLAN  Physical            Bridge                    St  Table Data
-------------------------------------------------------------------------------------------------------
ipobtls          Tagged 3002  1/m1/6/0/ipobridge  ipobridge-3002/bridge     UP  S 00:01:47:7f:e1:b2
                                                                               S 10.50.2.35
tls              Tagged 3002  1/a/2/0/eth         ethernet2-3002/bridge     UP  D 00:01:47:00:06:c0
                                                                               D 00:01:47:04:07:3d
                                                                               D 00:01:47:17:fd:76
                                                                               D 00:01:47:1a:db:11
                                                                               D 00:01:47:1f:df:10
                                                                               D 00:01:47:24:73:a9
                                                                               D 00:01:47:2b:a5:d9
                                                                               D 00:01:47:4d:38:c1
                                                                               D 00:01:47:4d:38:c3
                                                                               D 00:01:47:4d:38:c4
                                                                               D 00:01:47:52:2b:c8
                                                                               D 00:01:47:5e:01:11
                                                                               D 00:01:47:5e:01:36
                                                                               D 00:01:47:88:da:24
                                                                               D 00:01:47:93:67:d9
                                                                               D 00:01:47:93:67:dc
                                                                               D 00:01:47:ab:86:26
                                                                               D 00:01:47:bf:be:68
                                                                               D 00:01:47:d9:9b:a8
                                                                               D 00:01:47:e1:5c:12
                                                                               D c8:4c:75:8c:ef:2d
upl              Tagged 204   1/a/4/0/eth          ethernet4-204/bridge      UP  S VLAN 204 default
3 Bridge Interfaces displayed
```

Use the **bridge show** command with a VLAN ID to view all the bridges on a VLAN.

```
zSH> bridge show vlan 999
        Orig
Type    VLAN/SLAN  VLAN/SLAN  Physical            Bridge                    St  Table Data
-------------------------------------------------------------------------------------------------------
upl               Tagged 999  1/a/3/0/eth         ethernet3-999/bridge      UP  S VLAN 999 default
1 Bridge Interfaces displayed
```

Use the **bridge show** <*bridge interface*> command to view bridge interface information.

```
zSH> bridge show ipobridge-3002/bridge
        Orig
Type    VLAN/SLAN  VLAN/SLAN  Physical           Bridge                    St  Table Data
-------------------------------------------------------------------------------------------------
ipobtls          Tagged 3002  1/m1/6/0/ipobridge  ipobridge-3002/bridge     UP  S 00:01:47:7f:e1:b2
                                                                               D 10.50.2.35
1 Bridge Interface displayed
```

4- 5.3 Commands: bridge stats

You can use the **bridge stats** command to view the packets being sent or received on bridge interfaces.

```
zSH> bridge stats
Interface                      Received Packets     Transmitted Packets      Storm Detect Pkts      Byte Count
Name                           UCast MCast BCast  UCast MCast Bcast Error  UCast  MCast  Bcast Alm  Rcv   Xmt
-------------------------------------------------------------------------------------------------------------
ethernet5-9/bridge             15004  0     5     0     0     0     0      0      0      0     0   2065k  0
ethernet5-200/bridge             0    0     0     0     0     0     0      0      0      0     0     0    0
ethernet5-201/bridge             0    0     0     0     0     0     0      0      0      0     0     0    0
ethernet5-6/bridge               0    0     0     0     0     0     0      0      0      0     0     0    0
ethernet5-800/bridge             0    0     0     0     0     0     0      0      0      0     0     0    0
ethernet5-801/bridge          4904    0  10818     0     0     0     0      0      0      0     0   1070k  0
ethernet4-802-700/bridge         0    0     0     0     0     0     0      0      0      0     0     0    0
1-3-1-701-gponport-998/bridge    0    0     0     0   601   6005    0      0      0      0     0     0    0
ethernet8-1003/bridge          775k   0     0   678k    0    85     0   1792      0      0     0   1163M 1005M
ethernet8-0-1004/bridge          0    0     0     0     0     0     0      0      0      0     0     0    0
ethernet8-1005/bridge          243k   0     0   64841   0     0     0      0      0      0     0   365M 65860k
ethernet8-0-1006/bridge          0    0     0     0     0     0     0      0      0      0     0     0    0
ethernet8-1001/bridge          6896k  0     0   5734k   0    90     0      0      0      0     0 10345M 8601M
ethernet8-0-1002/bridge          0    0     0   94068   0     0     0      0      0      0     0     0   141M
ethernet2-94/bridge            42558  16 41002    0     0     0     0      0      0      0     0   5853k  0
ethernet2-500/bridge           99133  0     5   13631k  0     0     0      0      0      0     0 66912k 932M
ethernet2-0-502/bridge         42567  0     0    690k   0     0     0      0      0      0     0 3064k 48716k
...
```

If you add the name of a bridge you can see the statistics for that bridge.

```
zSH> bridge stats ethernet2-94/bridge
Interface                      Received Packets     Transmitted Packets      Storm Detect Pkts      Byte Count
Name                           UCast MCast BCast  UCast MCast Bcast Error  UCast  MCast  Bcast Alm  Rcv   Xmt
-------------------------------------------------------------------------------------------------------------
ethernet2-94/bridge            42661  16 41106    0     0     0     0      0      0      0     0   5867k  0
```

4- 5.4 Commands: SysName and SysNameAlias

The MXK-F allows for two optional names, to identify each MXK-F system, that are configured in the system 0 profile: SysName and SysNameAlias.

Procedure: *Configure SysName and SysNameAlias*

```
zSH> update system 0
system  0
Please provide the following: [q]uit.
syscontact: ---------------------->  {}:
sysname: ------------------------->  {}: MXK-F
syslocation: --------------------->  {}:
< skip >
sysNameAlias: -------------------->  {}: Aisle3_Rack2
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.

MXK-F>
```

When the sysname parameter is changed, the CLI dialog prompt “zSH>” changes to use the sysname “MXK-F>”. This document uses “zSH>” as a generic CLI prompt to abbreviate the command line text in each CLI shell dialog.

4- 6 SNTP

4- 6.1 Set System for SNTP

To set up the system to use SNTP update the **ntp-client-config** profile:

```
zSH> update ntp-client-config 0
ntp-client-config  0
Please provide the following: [q]uit.
primary-ntp-server-ip-address: --->  {0.0.0.0}: 192.168.8.100
secondary-ntp-server-ip-address: ->  {0.0.0.0}:
local-timezone: ------------------>  {gmt}: pacific
daylight-savings-time: ----------->  {false}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

4- 6.2 Set Daylight Savings Time Begin and End Times

To set the specific date and time for the beginning and end of daylight savings time add the month, day and time in the **daylight-savings-time-start** and **daylight-savings-time-end** parameters of the **ntp-client-config profile**. Follow the MM:DD:HH:MM (month:day:hour:minute) format.

For example to set the daylight savings time to begin on March 10 at 2am and end on November 3 at 2am, the actual times for 2013 DST, you would update the **ntp-client-config** as shown below.

```
zSH> update ntp-client-config  0

ntp-client-config  0
Please provide the following: [q]uit.
primary-ntp-server-ip-address: --->  {172.16.1.53}:
secondary-ntp-server-ip-address: ->  {0.0.0.0}:
local-timezone: ------------------>  {pacific}:
daylight-savings-time: ------------>  {true}:
daylight-savings-time-start: ----->  {03:10:02:00}:
                        daylight-savings-time-end: ------->  {11:03:02:00}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

Note: The **primary-ntp-server-ip-address** parameter must be non-zero to save changes to the **ntp-client-config** profile.

Note: When testing this feature, please ensure that there is at least two hours time between the start and end times of the cycle for the feature to operate correctly.

4- 7 SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

4- 7.1 Create SNMP Community Names and Access Profiles

Note: By default, the MXK-F has a single SNMP community defined with the name **ZhonePrivate**. This community has admin access to the system. DZS recommends that you configure community names and access profiles to prevent unauthorized access to the system.

The **community-profile** specifies the community name and an access level for SNMP manager to access the system. It can also optionally specify a **community-access-profile** which is used to verify the source IP address of the SNMP manager. The system supports up to 50 different access lists.

The following community access levels are supported:

- **noaccess**—the community has no access.
- **read**—the community has read-only access to the system, with the exception of information in the **community-profile** and **community-access-profile**.
- **readandwrite**—the community has read/write access to the system, with the exception of information in the **community-profile** and **community-access-profile**.
- **admin**—the community has read and write access to the entire system, including information in the **community-profile** and **community-access-profile**. Note that the ZMS requires admin access to manage the system.

4- 7.1.1 Create a Community Profile

Note: Configuring a community profile disables the **ZhonePrivate** default community name. If you do change the community name, you must change the name in ZMS or the device will become unmanageable.

This example defines a community name **public** with read-only privileges:

```
zSH> new community-profile 1
Please provide the following: [q]uit.
community-name: -----> {}: public
permissions: --------> {read}:
access-table-index: -> {0}:
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

4- 7.1.2 Create Community Access Profiles

The following example defines a community name **private** with read/write privileges and also creates an access list to verify that the SNMP client attempting to access the MXK-F is coming from known IP addresses 192.168.9.10 and 192.168.11.12:

First, create an access list for the first IP address:

```
zSH> new community-access-profile 2
Please provide the following: [q]uit.
access-table-index: ->  {0}: 1
ip-address: --------->  {0.0.0.0}: 192.168.9.10
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

Then, create an access list for the second IP address with the same **access-table-index** (1):

```
zSH> new community-access-profile 3
Please provide the following: [q]uit.
access-table-index: ->  {0}: 1
ip-address: --------->  {0.0.0.0}: 192.168.11.12
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

Finally, create a **community-profile** that specifies the community name, and uses the same **access-table-index** (1) as defined in the two **community-access-profiles** you just created:

```
zSH> new community-profile 4
Please provide the following: [q]uit.
community-name: ----->  {}: private ZMS must include this name
permissions: -------->  {read}: readandwrite
access-table-index: ->  {0}: 1
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

4- 7.2 Configure Traps

The **trap-destination** profile defines a trap recipient the MXK-F will send traps to. To configure a trap destination you need:

- the IP address of the SNMP trap server
- the community name the trap recipient expects

The other parameters in the **trap-destination** profile can be left at their default values. The following example configures a trap recipient with the IP address 192.168.3.21:

```
zSH> new trap-destination 32
Please provide the following: [q]uit.
trapdestination: ->  {0.0.0.0}: 192.168.3.21
communityname: --->  {}: public
resendseqno: ----->  {0}:
ackedseqno: ------>  {0}:
traplevel: ------->  {low}:
traptype: -------->  {(null)}: 0
trapadminstatus: ->  {enabled}:
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

Note: When ZMS configures a device, a trap destination profile is automatically created.

CHAPTER 5 PORT MANAGEMENT

This chapter describes MXK-F port management:

5- 1 PORT COMMAND OVERVIEW

Each Port/Interface has an internal name. The CLI Port/Interface Naming Convention is briefly explained here: Port/Interface Naming Convention.

The **port** command has various administrative functions and is used to:

- alter the administrative status of a physical port or virtual interface on the MXK-F with the **port up**, **port down**, **port bounce**, or **port testing** commands. See *Port Descriptions* on page 115.
- verify the administrative status of a physical port or virtual interface on the MXK-F with the **port show** command. See *View the Administrative and Operational States of Ports* on page 106.
- View DDM data on Ethernet SFPs with the **port show** command. See *View DDM data on Ethernet SFPs with the port show Command* on page 108.
- view the operational status, speed, and duplex mode of Ethernet ports with the **port status** command. See *View the Administrative and Operational States of Ports* on page 106.
- associate a text string with a physical interface, including bond groups, with the **port description** set of commands. See *Port Descriptions* on page 115.

- display or clear various statistical information on Ethernet ports with the **port stats** command.
- set the severity level of alarms on Ethernet ports with the **port config alarm** command. See the *MXK-F Monitoring and Trouble Shooting Guide.*
- configure jumbo Ethernet frames with the port config command and verify the change with the port show command. See *Ethernet Jumbo Frames* on page 130

5- 2 VIEW THE ADMINISTRATIVE AND OPERATIONAL STATES OF PORTS

5- 2.1 port status and port show Command - MXK-F14xx

The **port status** command displays operational status, speed, and duplex mode of an Ethernet port.

Use show to display port parameters.

You can use show to see alarm severity level set for a port as well

The **port show** command displays the status of an Ethernet or GPON port.

Note: The **port status** command is only valid for Ethernet ports.

Use the **port status** command to view the operational status, speed, and duplex mode of an Ethernet port.

```
zSH> port status 1-a-1-0/eth
Operational status  : Up
Rate in Mbps        : 10000
Duplex              : Full
```

Use the **port show** command to view the administrative status and additional information of an Ethernet port.

```
zSH> port show 1-a-1-0/eth
Interface 1-a-1-0/eth
        Physical location:     1/a/1/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM data:
                Temperature:        30c
                Voltage:           3.29v
                Tx bias current:     27mA
                Transmit power:    -2.3dBm
                Receive power:     0.2dBm
```

Use the **port show** command to view the status of a GPON OLT port.

```
zSH> port show 1-3-1-0/gponolt
Interface 1-3-1-0/gponolt
        Physical location:     1/3/1/0/gponolt
        Administrative status: up
```

Use the **port show** command to view status of an ONU on a GPON OLT port.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: up
```

Use the **port show** command to view the status of a port with a configured bridge.

```
zSH> port show ethernet1-800/bridge
Interface ethernet1-800/bridge
        Administrative status: up
```

5- 2.2 port status and port show Command - MXK-F219

The **port status** command displays operational status, speed, and duplex mode of an Ethernet port.

Use show to display port parameters.

You can use show to see alarm severity level set for a port as well

The **port show** command displays the status of an Ethernet or GPON port.

Note: The **port status** command is only valid for Ethernet ports.

Use the **port status** command to view the operational status, speed, and duplex mode of an Ethernet port.

```
zSH> port status 1-m1-1-0/eth
Operational status  : Up
Rate in Mbps        : 100
Duplex              : Full

zSH> port status 1-1-101-0/eth
Operational status  : Up
Rate in Mbps        : 10000
Duplex              : Full
```

Use the **port show** command to view the administrative status and additional information of an Ethernet port or interface.

```
zSH> port show 1-m1-1-0/eth
Interface 1-m1-1-0/eth
        Physical location:     1/m1/1/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
```

```
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Egress rate: 0 Kbps burst size: 0 Kbits
        No DDM data available from ethernet port

zSH> port show 1-1-101-0/eth
Interface 1-1-101-0/eth
        Physical location:     1/1/101/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Egress rate: 0 Kbps burst size: 0 Kbits
        DDM not supported
```

Use the **port show** command to view the status of a GPON OLT port.

```
zSH> port show 1-1-1-0/gponolt
Interface 1-1-1-0/gponolt
        Physical location:     1/1/1/0/gponolt
        Administrative status: up
```

Use the **port show** command to view status of an ONU on a GPON OLT port.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: up
```

Use the **port show** command to view the status of a port with a configured bridge.

```
zSH> port show ethernet1-101-400/bridge
Interface ethernet1-101-400/bridge
        Administrative status: up
```

5- 3 VIEW DDM DATA ON ETHERNET SFPS WITH THE PORT SHOW COMMAND

This section describes DDM on SFPs for Ethernet:

- DDM Data on Ethernet SFPs Overview, page 108
- DDM Data on Fabric Card Ethernet SFPs - MXK-F14xx, page 109

5- 3.1 DDM Data on Ethernet SFPs Overview

Digital Diagnostic Monitoring (DDM) provides SFP diagnostic data. For SFPs that support DDM, the SFP transceiver measures the temperature, supply voltage, transmit bias current, transmit power, and the receive power on the SFP.

Use the **port show** *interface/type* to display DDM data on Ethernet ports using SFPs that support DDM. Table 13. describes the DDM data fields displayed.

Table 13: port show Command Output Fields for DDM data on Ethernet Ports

Field	Description
Temperature	Internally measured Transceiver Temperature in degrees celsius.
Voltage	Internally measured Transceiver Supply Voltage in hundredths of volts.
Tx Bias Current	Measured Tx Bias current in milliamperes.
Transmit Power	Measured Tx Output power in tenths of dB.
Receive Power	Measured Rx power in tenths of dB.

5- 3.2 DDM Data on Fabric Card Ethernet SFPs - MXK-F14xx

Ethernet port on fabric card with SFP that supports DDM data.

```
zSH> port show 1-a-1-0/eth
Interface 1-a-1-0/eth
        Physical location:     1/a/1/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM data:
                Temperature:        29c
                Voltage:           3.29v
                Tx bias current:      5mA
                Transmit power:    -2.1dBm
                Receive power:     -4.0dBm
```

Ethernet port on fabric card with without SFP.

```
zSH> port show 1-a-3-0/eth
Interface 1-a-3-0/eth
        Physical location:     1/a/3/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

Ethernet port on fabric card with SFP that does not support DDM data.

```
zSH> port show 1-a-5-0/eth
Interface 1-a-5-0/eth
        Physical location:     1/a/5/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
```

```
        DDM not supported
```

Ethernet management port that does not use an SFP.

```
zSH> port show 1-m1-1-0/eth
Interface 1-m1-1-0/eth
        Physical location:     1/m1/1/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        No DDM data available from ethernet port
```

5- 4 ADMIN STATES: PORT TESTING/UP/DOWN/ BOUNCE - MXK-F14XX

5- 4.1 port testing Command

Use the **port testing** command to set the administrative state to *testing* on an Ethernet port.

```
zSH> port testing 1-a-2-0/eth
1-a-2-0/eth set to admin state TESTING
```

Verify the state.

```
zSH> port show 1-a-2-0/eth
Interface 1-a-2-0/eth
        Physical location:     1/a/2/0/eth
        Administrative status: testing
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM data:
                Temperature:        29c
                Voltage:           3.29v
                Tx bias current:     28mA
                Transmit power:    -2.4dBm
                Receive power:     -4.0dBm
```

Use the **port testing** command to set the administrative state to *testing* on a GPON ONU port.

```
zSH> port testing 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state TESTING
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: testing
```

5- 4.2 port up Command

Use the **port up** command to set the administrative state to *up* on an Ethernet port.

```
zSH> port up 1-a-2-0/eth
1-a-2-0/eth set to admin state UP
```

Verify the state.

```
zSH> port show 1-a-2-0/eth
Interface 1-a-2-0/eth
        Physical location:     1/a/2/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM data:
                Temperature:        29c
                Voltage:           3.29v
                Tx bias current:     28mA
                Transmit power:    -2.4dBm
                Receive power:     -4.0dBm
```

Use the **port up** command to set the administrative state to *up* on an gpononu port.

```
zSH> port up 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state UP
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: up
```

5- 4.3 port down Command

Use the **port down** command to set the administrative state to down on an Ethernet port.

```
zSH> port down 1-a-2-0/eth
1-a-2-0/eth set to admin state DOWN
```

Verify the state.

```
zSH> port show 1-a-2-0/eth
Interface 1-a-2-0/eth
        Physical location:     1/a/2/0/eth
        Administrative status: down
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM data:
```

```
        Temperature:        29c
        Voltage:           3.29v
        Tx bias current:     28mA
        Transmit power:    -2.4dBm
        Receive power:     -4.0dBm
        ALARMS PRESENT:  txBiasLo+txPwrLo
```

Use the **port down** command to set the administrative state to down on a GPON ONU port.

```
zSH> port down 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state DOWN
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: down
```

5- 4.4 port bounce Command

Use the **port bounce** command to perform a down operation followed by an up operation on an Ethernet port.

```
zSH> port bounce 1-a-2-0/eth
1-a-2-0/eth set to admin state DOWN
1-a-2-0/eth set to admin state UP
```

Verify the state.

```
zSH> port show 1-a-2-0/eth
Interface 1-a-2-0/eth
        Physical location:     1/a/2/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM data:
                Temperature:        29c
                Voltage:           3.29v
                Tx bias current:     28mA
                Transmit power:    -2.4dBm
                Receive power:     -4.0dBm
```

Use the **port bounce** command to perform a down operation followed by an up operation on a GPON ONU port.

```
zSH> port bounce 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state DOWN
1-1-1-1/gpononu set to admin state UP
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: up
```

5- 5 ADMIN STATES: PORT TESTING/UP/DOWN/ BOUNCE - MXK-F219

5- 5.1 port testing Command

Use the **port testing** command to set the administrative state to *testing* on an Ethernet port.

```
zSH> port testing 1-2-101-0/eth
1-2-102-0/eth set to admin state TESTING
```

Verify the state.

```
zSH> port show 1-2-101-0/eth
Interface 1-2-101-0/eth
        Physical location:     1/2/101/0/eth

        Administrative status: testing
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Egressrate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

Use the **port testing** command to set the administrative state to *testing* on a GPON ONU port.

```
zSH> port testing 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state TESTING
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Administrative status: testing
```

5- 5.2 port up Command

Use the **port up** command to set the administrative state to *up* on an Ethernet port.

```
zSH> port up 1-2-101-0/eth
1-2-101-0/eth set to admin state UP
```

Verify the state.

```
zSH> port show 1-2-101-0/eth
Interface 1-2-101-0/eth
        Physical location:     1/2/101/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
```

```
        Egress rate: 0 Kbps burst size: 0 Kbits
    SFP not present
```

Use the **port up** command to set the administrative state to *up* on an gpononu port.

```
zSH> port up 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state UP
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
    Administrative status: up
```

5- 5.3 port down Command

Use the **port down** command to set the administrative state to down on an Ethernet port.

```
zSH> port down 1-2-101-0/eth
1-2-101-0/eth set to admin state DOWN
```

Verify the state.

```
zSH> port show 1-2-101-0/eth
Interface 1-2-101-0/eth
    Physical location:     1/2/101/0/eth
    Administrative status: down
    Port type specific information:
        Frame size: 0 bytes
        Ingress rate: 0 Kbps burst size: 0 Kbits
        Engress rate: 0 Kbps burst size: 0 Kbits
    SFP not present
```

Use the **port down** command to set the administrative state to down on a GPON ONU port.

```
zSH> port down 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state DOWN
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
    Administrative status: down
```

5- 5.4 port bounce Command

Use the **port bounce** command to perform a down operation followed by an up operation on an Ethernet port.

```
zSH> port bounce 1-2-101-0/eth
1-2-101-0/eth set to admin state DOWN
1-2-101-0/eth set to admin state UP
```

Use the **port bounce** command to perform a down operation followed by an up operation on a GPON ONU port.

```
zSH> port bounce 1-1-1-1/gpononu
1-1-1-1/gpononu set to admin state DOWN
1-1-1-1/gpononu set to admin state UP
```

Verify the state.

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
       Administrative status: up
```

5- 6 PORT DESCRIPTIONS

This section describes port descriptions:

5- 6.1 Port Description Rules

The MXK-F has a port description field, which provides a mapping between the physical port, or bonded interface, or bridge and a subscriber. This mapping improves MXK-F management without requiring extra documents to provide the mapping. Port description information can be entered for ports, bridges, or bond groups. Port description information is also searchable.

The rules for entering a port description are:

- Port descriptions do not have to be unique.
- The port description field is a text field 64 characters long.
- Any characters can be used including spaces,$,@,-,.,etc. The only characters not supported are the double quote, ‘ “ ‘which is a delimiter to identify the beginning and end of the text string, the carat ‘^’, and the question mark ‘?’.
- Port descriptions are associated with physical ports and not logical interfaces. For bonding technologies port descriptions are supported both on the physical port and the bond group, so if you want to use a keyword such as a company name to group interfaces.
- Even though port descriptions are searchable, you cannot perform commands using port description. For example, you can not use a command like “bridge modify circuitName …”

5- 6.2 Add, Modify, List, and Delete a Port Description MXK-F14xx

The **port description add** command associates a text string with a physical interface (which includes bond groups):

port description add <physical interface> <text string>

Note: Port descriptions do not need to be unique. If one customer has many lines, they may all have the same port description. You may also use the port description field as a means to group interfaces. See Search Port Descriptions MXK-F14xx, page 119.

5- 6.2.1 Add a Port Description to a Port

To add a port description with spaces to a port, enter:

```
zSH> port description add 1-a-7-0/eth "555 555 5555"
```

In this case, the port description has spaces so quotes are needed.

To verify the port description, enter:

```
zSH> port show 1-a-7-0/eth
Interface 1-a-7-0/eth
        Physical location:     1/a/7/0/eth
        Description:     555 555 5555
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

To add a port description without spaces to a port, enter:

```
zSH> port description add 1-a-6-0/eth BusinessPark
```

To verify the port description enter:

```
zSH> port show 1-a-6-0/eth
Interface 1-a-6-0/eth
        Physical location:     1/a/6/0/eth
        Description:     BusinessPark
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

5- 6.2.2 Add a Port Description to a GPON Port

GPON ports have one ONT and up to 64 ONUs. Both the ONT and the ONUs can have port descriptions.

To add a port description on a GPON ONT, enter:

```
zSH> port description add 1-4-1-0/gponolt SFO
```

To verify the port description, enter:

```
zSH> port show 1-4-1-0/gponolt
Interface 1-4-1-0/gponolt
        Physical location:     1/4/1/0/gponolt
        Description:     SFO
        Administrative status: up
```

To add a port description to a GPON ONU, enter:

```
zSH> port description add 1-4-1-1/gpononu "business 1 555-555-5555"
```

In this case, a port description is added to ONU 1 on OLT 1.

To verify the port description, enter:

```
zSH> port show 1-4-1-1/gpononu
Interface 1-4-1-1/gpononu
        Description:     business 1 555-555-5555
        Administrative status: up
```

5- 6.2.3 Add a Port Description to a Bridge

The port description must be add to the physical port of a bridge configuration. A port description can be added to the physical port of an existing bridge configuration or the port description can be added to the physical port that is then configured as a bridge.

View existing bridges:

```
zSH> bridge show vlan 500
        Orig
Type    VLAN/SLAN  VLAN/SLAN   Physical         Bridge                          St  Table Data
-------------------------------------------------------------------------------------------------------
dwn               Tagged 500   1/3/1/1/gpononu  1-3-1-501-gponport-500/bridge  UP
upl               Tagged 500   1/a/2/0/eth      ethernet2-500/bridge           UP  S VLAN 500 default
2 Bridge Interfaces displayed
```

Add the port description to the physical port of an existing bridge configuration, in this case the downlink bridge on Ethernet port 2:

```
zSH> port description add 1-a-2-0/eth "US Insurance Consortium, Inc."
```

Verify the port description on the downlink bridge:

```
zSH> bridge showdetail ethernet2-0-500/bridge
Bridge interface: ethernet2-0-500
        Administrative status: up        Operational status: up
        Blocked status: unblocked
        Type:upl         ST    0/500
        Data: S SLAN VLAN 0 default[U: 3600 sec, M: 250 sec, I: 0 sec]
Physical interface: 1-a-2-0/eth
        Administrative status: up        Operational status: up
Total Packet Statistics
```

```
Description: US Insurance Consortium, Inc.
Received
                 Unicast: 836
               Multicast: 0
               Broadcast: 0
                   Bytes: 60192
       Sent
                 Unicast: 0
               Multicast: 0
               Broadcast: 0
                   Bytes: 0
                  Errors: 0
       Packet Storm Blocked
                 Unicast: 0
               Multicast: 0
               Broadcast: 0
                  Alarms: 0
Delta Packet Statistics - Collecting a 1 second data interval
                    Received                          Sent
         Unicast  Multicast  Broadcast    Unicast  Multicast  Broadcast  Error
Delta          0          0          0          0          0          0      0
Rate           0          0          0          0          0          0      0
          IGMP  Received                      IGMP Transmitted
 GenQuery  SpecQuery  vxReport    Leave    GenQuery  SpecQuery  vxReport    Leave
   0/0       0/0       0/0          0      0/0       0/0       0/0          0
IGMP misc: unknown=0 errorRx=0 actChans=0 actHosts=0
```

5- 6.2.4 Modify a Port Description

The port description modify command allows you to edit an existing port description.

port description modify <**physical interface**> <**text string**>

Enter a port description:

```
zSH> port description add 1-4-1-2/gpononu "Business, Inc"
```

Verify the description:

```
zSH> port show 1-4-1-2/gpononu
Interface 1-4-1-2/gpononu
       Description:    Business, Inc.
       Administrative status: up
```

Modify the description on the same port:

```
zSH> port description modify 1-4-1-2/gpononu "Stadium, Inc."
```

Verify the change:

```
zSH> port show 1-4-1-2/gpononu
Interface 1-4-1-2/gpononu
       Description:    Stadium, Inc
       Administrative status: up
```

5- 6.2.5 Port Description List

The **port description list** command will list the descriptions on a particular port.

```
zSH> port description list 1/4/1
Interface                              Description
-------------------------------------------------------------------------------
1-4-1-0/gponolt                        SFO
1-4-1-1/gpononu                        business 1 555-555-5555
1-4-1-2/gpononu                        -
1-4-1-3/gpononu                        -
1-4-1-4/gpononu                        -
...
```

5- 6.2.6 Port Description Delete

The port description delete command removes the port description from the physical interface.

port description delete <**physical interface**>

To view the port description on a physical port enter:

```
zSH> port show 1-4-1-2/gpononu
Interface 1-4-1-2/gpononu
       Description:    Stadium, Inc.
       Administrative status: up
```

To delete the port description enter:

```
zSH> port description delete 1-4-1-2/gpononu
```

To verify the deletion enter:

```
zSH> port show 1-4-1-2/gpononu
Interface 1-4-1-2/gpononu
       Administrative status: up
```

5- 6.3 Search Port Descriptions MXK-F14xx

The **port description find** command provides a textual search which allows you search for a text string within the port description fields. The display show the description and the physical location. If multiple port descriptions have the same text string they will all be displayed

port description find <**text string**>

```
zSH> port description find SFO
Results for SFO
Description:     SFO
Interface:       1-4-1-0/gponolt

zSH> port description find "business 1 555-555-5555"
Results for business 1 555-555-5555
Description:     business 1 555-555-5555
```

```
Interface:      1-4-1-1/gpononu
```

Note: Notice that for search items which do not have spaces the quotation marks are unnecessary.

5- 6.4 Add, Modify, List, and Delete a Port Description MXK-F219

The **port description add** command associates a text string with a physical interface (which includes bond groups):

```
port description add <physical interface> <text string>
```

Note: Port descriptions do not need to be unique. If one customer has many lines, they may all have the same port description. You may also use the port description field as a means to group interfaces. See Search Port Descriptions MXK-F14xx, page 119.

5- 6.4.1 Add a Port Description to a Port

To add a port description with spaces to a port, enter:

```
zSH> port description add 1-2-101-0/eth "555 555 5555"
```

In this case, the port description has spaces so quotes are needed.

To verify the port description, enter:

```
zSH> port show 1-2-101-0/eth
Interface 1-2-101-0/eth
        Physical location:     1/2/101/0/eth
        Description:     555 555 5555
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Egress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

To add a port description without spaces to a port, enter:

```
zSH> port description add 1-2-102-0/eth BusinessPark
```

To verify the port description enter:

```
zSH> port show 1-2-102-0/eth
Interface 1-2-102-0/eth
        Physical location:     1/2/102/0/eth
        Description:     BusinessPark
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
```

```
            Engress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

5- 6.4.2 Add a Port Description to a GPON Port

GPON ports have one ONT and up to 64 ONUs. Both the ONT and the ONUs can have port descriptions.

To add a port description on a GPON ONT, enter:

```
zSH> port description add 1-1-1-0/gponolt SFO
```

To verify the port description, enter:

```
zSH> port show 1-1-1-0/gponolt
Interface 1-1-1-0/gponolt
        Physical location:     1/1/1/0/gponolt
        Description:     SFO
        Administrative status: up
```

To add a port description to a GPON ONU, enter:

```
zSH> port description add 1-1-1-1/gpononu "business 1 555-555-5555"
```

In this case, a port description is added to ONU 1 on OLT 1.

To verify the port description, enter:

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Description:     business 1 555-555-5555
        Administrative status: up
```

5- 6.4.3 Add a Port Description to a Bridge

The port description must be add to the physical port of a bridge configuration. A port description can be added to the physical port of an existing bridge configuration or the port description can be added to the physical port that is then configured as a bridge.

View existing bridges:

```
zSH> bridge show vlan 999
        Orig
Type    VLAN/SLAN  VLAN/SLAN   Physical      Bridge                      St  Table Data
-------------------------------------------------------------------------------------------------
upl               Tagged 999  1/2/102/0/eth  ethernet1-102-999/bridge    DWN S VLAN 999 default
1 Bridge Interface displayed
```

Add the port description to the physical port of an existing bridge configuration, in this case the downlink bridge on Ethernet port 2:

```
zSH> port description add 1-1-102-0/eth "US Insurance Consortium, Inc."
```

Verify the port description on the downlink bridge:

```
zSH> bridge showdetail ethernet1-102-999/bridge
Bridge interface: ethernet1-102-999
```

```
        Administrative status: up         Operational status: down
        Blocked status: unblocked
        Type:upl         Tagged 999
        Data: S VLAN 999 default[U: 3600 sec, M: 250 sec, I: 0 sec]

Physical interface: 1-1-102-0/eth
        Administrative status: up         Operational status: down


Description: US Insurance Consortium, Inc.

Total Packet Statistics
        Received
                  Unicast: 0
                Multicast: 0
                Broadcast: 0
                    Bytes: 0
        Sent
                  Unicast: 0
                Multicast: 0
                Broadcast: 0
                    Bytes: 0
                   Errors: 0
        Packet Storm Blocked
                  Unicast: 0
                Multicast: 0
                Broadcast: 0
                   Alarms: 0

Delta Packet Statistics - Collecting a 1 second data interval
                     Received                          Sent
          Unicast  Multicast  Broadcast    Unicast  Multicast  Broadcast  Error
Delta           0          0          0          0          0          0      0
Rate            0          0          0          0          0          0      0

          IGMP  Received                       IGMP Transmitted
 GenQuery  SpecQuery  vxReport    Leave    GenQuery  SpecQuery  vxReport    Leave
   0/0       0/0       0/0          0      0/0       0/0       0/0          0
IGMP misc: unknown=0 errorRx=0 actChans=0 actHosts=0
```

5- 6.4.4 Modify a Port Description

The port **description modify** command allows you to edit an existing port description.

port description modify <**physical interface**> <**text string**>

Enter a port description:

```
zSH> port description add 1-1-1-1/gpononu "Business, Inc"
```

Verify the description:

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Description:    Business, Inc.
        Administrative status: up
```

Modify the description on the same port:

```
zSH> port description modify 1-1-1-1/gpononu "Stadium, Inc."
```

Verify the change:

```
zSH> port show 1-1-1-1/gpononu
Interface 1-1-1-1/gpononu
        Description:     Stadium, Inc
        Administrative status: up
```

5- 6.4.5 Port Description List

The **port description list** command will list the descriptions on a particular port.

```
zSH> port description list 1/2/101
Interface                              Description
-------------------------------------------------------------------------------
1-2-101-0/eth                          555 555 5555
```

5- 6.4.6 Port Description Delete

The port description delete command removes the port description from the physical interface.

port description delete <**physical interface**>

To view the port description on a physical port enter:

```
zSH> port show 1-2-101-0/eth
Interface 1-2-101-0/eth
        Physical location:     1/2/101/0/eth
        Description:     555 555 5555
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Egress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

To delete the port description enter:

```
zSH> port description delete 1-2-101-0/ethu
```

To verify the deletion enter:

```
zSH> port show 1-2-101-0/eth
Interface 1-2-101-0/eth
        Physical location:     1/2/101/0/eth
        Description:     555 555 5555
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Egress rate: 0 Kbps burst size: 0 Kbits
        SFP not present
```

5- 6.5 Search Port Descriptions MXK-F219

The **port description find** command provides a textual search which allows you search for a text string within the port description fields. The display show the description and the physical location. If multiple port descriptions have the same text string they will all be displayed

port description find <**text string**>

```
zSH> port description find BusinessParK
Results for Business

Description:     BusinessPark
Interface:       1-2-102-0/eth
```

Note: Notice that for search items which do not have spaces the quotation marks are unnecessary.

5- 7 PORT MIRRORING

The MXK-F provides port mirroring as a diagnostic/troubleshooting tool that sends copied/mirrored packets to an uplink mirror port. The Mirroring function is implemented as part of the "fabric/switching" function of an MXK-F. On the MXK-F14xx the switching function is on the (a/b) Fabric Cards. On the MXK-219 the switching function is on the Line Cards (an m1 uplink mirror port uses LC1, slot 1, and an m2 uplink mirror uses LC2, slot 2).

The rules for port mirroring are:

- The MXK-F supports one mirror at a time.
- All mirrored uplink ports must be on the same card even in a redundant configuration
- Any uplink port can be mirrored to any other uplink port on the same card (between same size ports or smaller ports can be mirrored to larger ports).
- For MXK-F14xx: When a port is a link aggregration group member, either the group or one port in the group can be mirrored (the MXK-F-219 does not support LinkAgg mirroring).

Note: If more than one uplink port needs to be mirrored, you must put the ports in a link aggregration group. The ports must stay in the link aggregration group for mirroring to continue.

- For MXK-F219: A GPON port can be mirrored to an uplink port (the MXK-F-14xx does not support GPON mirroring).

5- 7.1 port mirror Command Syntax

The syntax for the **port mirror** command is:

```
port mirror <from-interface> <to-interface> <vlan <vlanId>> <in|out|both|off>
```

Table 14: Variables for the port mirror Command

Variable	Definition
from-interface	The interface to mirror.
to-interface	Where to send the packets.
vlanID	The outer VLAN tag.
in	Mirror the incoming packets.
out	Mirror the outgoing packets.
both	Mirror both the incoming and outgoing packets.
off	Disable port mirroring for the port interface.

5- 7.2 Create a Mirrored Uplink Port

Procedure: *Case 1: Configure an Uplink Mirror Port for Incoming Packets on an Uplink Port*

Create a mirror with the source port, destination mirror port, mirror vlan, and make the source packet direction **in** (for incoming packets).

– **For MXK-F14xx:**

Open a CLI connection to the card with the source and mirror ports.

```
zSH> connect a
Connecting to shelf: 1, slot: a ......
```

Mirror the incoming vlan 100 packets on the source port *1-a-3-0/eth* to uplink mirror port *1-a-4-0/eth*.

```
1/a-zSH> port mirror 1-a-3-0/eth 1-a-4-0/eth vlan 100 in
```

To turn port mirroring *off*.

```
1/a-zSH> port mirror 1-a-3-0/eth 1-a-4-0/eth vlan 100 off
```

When port mirroring is no longer needed, close CLI card connection.

```
1/a-zSH> exit
Connection closed.
zSH>
```

– **For MXK-219:**

Open a CLI connection to the Line Card that is paired with the m1/m2 uplink mirror port (the m1 fabric mirror function is on LC slot 1).

```
zSH> connect 1
Connecting to shelf: 1, slot: 1 ......
```

Mirror incoming vlan 100 packets on the source port *1-1-101-0/eth* to uplink mirror port *1-1-102-0/eth*.

```
1/1-zSH> port mirror 1-1-101-0/eth 1-1-102-0/eth vlan 100 in
```

To turn port mirroring *off*.

```
1/1-zSH> port mirror 1-1-101-0/eth 1-1-102-0/eth vlan 100 off
```

When port mirroring is no longer needed, close CLI card connection.

```
1/1-zSH> exit
Connection closed.
zSH>
```

Procedure: *Case 2: Configure an Uplink Mirror Port for Outgoing Packets on an Uplink Port*

Create a mirror with the source port, destination mirror port, vlan and specify the packet direction **out** (for outgoing packets).

– **For MXK-F14xx:**

Open a CLI connection to the card with the source and mirror ports.

```
zSH> connect a
Connecting to shelf: 1, slot: a ......
```

Mirror the outgoing vlan 200 packets on the source port *1-a-3-0/eth* to uplink mirror port *1-a-4-0/eth*.

```
1/a-zSH> port mirror 1-a-5-0/eth 1-a-6-0/eth vlan 200 out
```

To turn port mirroring *off*.

```
1/a-zSH> port mirror 1-a-5-0/eth 1-a-6-0/eth vlan 200 off
```

When port mirroring is no longer needed, close CLI card connection.

```
1/1-zSH> exit
Connection closed.
zSH>
```

– **For MXK-219:**

Open a CLI connection to the Line Card that is paired with the source m1/m2 uplink port (the m1 fabric mirror function is on LC slot 1).

```
zSH> connect 1
Connecting to shelf: 1, slot: 1 ......
```

Mirror the outgoing vlan 200 packets on source port *1-1-101-0/eth* to uplink mirror port *1-1-102-0/eth*.

```
1/1-zSH> port mirror 1-1-101-0/eth 1-1-102-0/eth vlan 200 out
```

To turn port mirroring *off*.

```
1/1-zSH> port mirror 1-1-101-0/eth 1-1-102-0/eth vlan 200 off
```

When port mirroring is no longer needed, close CLI card connection.

```
1/1-zSH> exit
Connection closed.
zSH>
```

5- 7.3 Create an MXK-F14xx Mirror Uplink Port for a LinkAgg Group

Procedure: *Case 3: Configure an Uplink Mirror Port for In & Out Pkts on a LinkAgg Group*

An uplink mirror can be created for a LinkAgg Group on an MXK-F14xx (the MXK-F219 does not have enough uplink ports to setup a LinkAgg Group and a mirror port)

View the ports in the link aggregration group.

```
zSH> linkagg show
LinkAggregations:
slot unit ifName        partner: Sys            Pri     grp ID  status  agg mode
-------------------------------------------------------------------------------
  a*  1   1-a-1-0       00:00:00:00:00:00       0x0     0x0      OOS    Active
        links           slot    port    subport         status
        ---------------------------------------------------------
         1-a-7-0         a       7       0               ACT
         1-a-6-0         a       6       0               ACT
  b   1   1-b-1-0       00:00:00:00:00:00       0x0     0x0      OOS    Active
        links           slot    port    subport         status
        ---------------------------------------------------------
         1-b-7-0         a       7       0               DSA
         1-b-6-0         b       6       0               DSA
global linkagg group red type: red
```

Open a CLI connection to the card with the source and mirror ports.

```
zSH> connect a
Connecting to shelf: 1, slot: a ......
```

Create a mirror with the source port, destination mirror port, vlan and specify the packet direction **both**. In this example the incoming and outgoing vlan 900 packets on the source LinkAgg Group *1-a-1-0/linkagg* are mirrored to the *1-a-8-0/eth* uplink mirror port.

```
1/a-zSH> port mirror 1-a-1-0/linkagg 1-a-8-0/eth vlan 900 both
```

To turn port mirroring *off*.

```
1/a-zSH> port mirror 1-a-1-0/linkagg 1-a-8-0/eth vlan 900 off
```

When port mirroring is no longer needed, close the CLI card connection.

```
1/a-zSH> exit
Connection closed.
zSH>
```

5- 7.4 Create an MXK-F219 Mirror Uplink Port for a GPON Port

Procedure: *Case 3: Configure an Uplink Mirror Port for In & Out Pkts on a GPON Port*

On the MXK-F219 a GPON port can be mirrored to an m1/m2 uplink port. The mirroring process is performed on the Line Card, so the CLI connection is made to the Line Card (not to the m1/m2 management/uplink card). A GPON port on LC 1 can be mirrored to an m1 uplink port (not m2) and a GPON port on LC 2 can be mirrored to an m2 uplink port (not m1).

Open a CLI connection to the Line Card with the source GPON port (the m1 fabric mirror function is on LC slot 1).

```
zSH> connect 1
Connecting to shelf: 1, slot: 1 ......
Connection established.
```

Create a mirror with the source port, destination mirror port, vlan and specify the packet direction **both**. In this example the incoming and outgoing vlan 998 packets on the source GPON port *1-1-1-1/gpononu* are mirrored to the *1-1-102-0/eth* uplink mirror port.

```
1/1-zSH> port mirror 1-1-1-1/gpononu 1-1-102-0/eth vlan 998 both
creating mirror port 8 bcm 9 v 998 mp 102 mbp 27 m 3
```

Verify the mirror.

```
1/1-zSH> port mirror show
port 1/1/1/0/gpononu mirrored to port 1/1/102/0/eth vlan 998 mode Both
```

To turn port mirroring *off*.

```
1/1-zSH> port mirror 1-1-1-1/gpononu 1-1-102-0/eth vlan 998 off
```

When port mirroring is no longer needed, close the CLI card connection.

```
1/1-zSH> exit
Connection closed.
zSH>
```

5- 8 ETHERNET JUMBO FRAMES

Jumbo Ethernet frames are defined as frames that exceed 1500 bytes of payload. Jumbo Ethernet frames are usually up to 9000 bytes of payload and are frequently used by data centers to provide lower overhead Ethernet connectivity. Enterprise Ethernet, carrier Ethernet, and access networks are now frequently requiring jumbo Ethernet frames.

```
zSH> port show 1-1-1-0/eth
Interface 1-1-1-0/eth
        Physical location:     1/1/1/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 0 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
       DDM not supported
```

```
zSH> port config 1-1-1-0/eth maxframe 9120
 Setting max frame size to: 9120 bytes.
  Interface 1-1-1-0/eth configured for max frame size of 9120.
```

```
zSH> port show 1-1-1-0/eth
Interface 1-1-1-0/eth
        Physical location:     1/1/1/0/eth
        Administrative status: up
        Port type specific information:
                Frame size: 9120 bytes
                Ingress rate: 0 Kbps burst size: 0 Kbits
                Engress rate: 0 Kbps burst size: 0 Kbits
        DDM not supported
```

```
zSH> get ether 1/1/1/0
ether  1/1/1/0
autonegstatus: ----> {enabled}
mauType: ----------> {mau1000basetfd}
restart: ----------> {norestart}
ifType: -----------> {mau1000basetfd}
autonegcap: -------> {b100baseTX+b100baseTXFD+b1000baseT+b1000baseTFD}
remotefault: ------> {noerror}
clksrc: -----------> {automatic}
pauseFlowControl: -> {disabled}
aggregationMode: --> {on}
linkStateMirror: --> {0/0/0/0/0}
maxFrameSize: -----> {9120}
ingressRate: ------> {0}
ingressBurstSize: -> {0}
egressRate: -------> {0}
egressBurstSize: --> {0}
```

```
zSH> port config 1-1-1-0/eth maxframe 0
  Interface 1-1-1-0/eth configured to default frame size of 2048 bytes.



zSH> get ether 1/1/1/0
ether  1/1/1/0
autonegstatus: ----> {enabled}
mauType: ----------> {mau1000basetfd}
restart: ----------> {norestart}
ifType: -----------> {mau1000basetfd}
autonegcap: -------> {b100baseTX+b100baseTXFD+b1000baseT+b1000baseTFD}
remotefault: ------> {noerror}
clksrc: -----------> {automatic}
pauseFlowControl: -> {disabled}
aggregationMode: --> {on}
linkStateMirror: --> {0/0/0/0/0}
maxFrameSize: -----> {0}
ingressRate: ------> {0}
ingressBurstSize: -> {0}
egressRate: -------> {0}
egressBurstSize: --> {0}
```

CHAPTER 6 SECURITY

This chapter describes MXK-F operations, system administration, and maintenance functions:

6- 1 SECURITY USING SSH, SFTP, AND HTTPS

This section covers the security on the MXK-F:

Note: For security reasons, host keys are not accessible via SNMP and cannot be saved/restored with the dump command.

6- 1.1 Enable Security SSH, SFTP and HTTPS

The **system 0** profile provides a **secure** parameter which allows only secure communication for management activities. When security is enabled, the MXK-F uses the following protocols:

- Secure File Transfer Protocol (SFTP)
- Secure shell (SSH)
- HTTPS (HTTP secure)

Table 15 describes which protocols are allowed when the **secure** parameter is *enabled* and which protocols are allowed when the **secure** parameter is *disabled*.

Table 15: Protocols for the Secure Parameter

Disabled	Enabled
TFTP, FTP	SFTP
Telnet, SSH	SSH
HTTP	HTTPS

Procedure: *Enabling Security on the MXK-F*

To enable the security parameter enter **update system 0** on the MXK-F, change the secure parameter from *disabled* to *enabled*, then save the file:

Note: After enabling the **secure** parameter, HTTPS and changes to the Web UI take affect after the next reboot. SSH and SFTP do not require a reboot.

```
zSH> update system 0
system  0
Please provide the following: [q]uit.
syscontact: ----------------------> {}:
sysname: -------------------------> {}:
syslocation: ---------------------> {}:
enableauthtraps: -----------------> {disabled}:
setserialno: ---------------------> {0}:
zmsexists: -----------------------> {false}:
zmsconnectionstatus: -------------> {inactive}:
zmsipaddress: --------------------> {0.0.0.0}:
configsyncexists: ----------------> {false}:
configsyncoverflow: --------------> {false}:
configsyncpriority: --------------> {high}:
configsyncaction: ----------------> {noaction}:
configsyncfilename: --------------> {}:
configsyncstatus: ----------------> {syncinitializing}:
configsyncuser: ------------------> {}:
configsyncpasswd: ----------------> {** private **}:  ** read-only **
numshelves: ----------------------> {1}:
shelvesarray: --------------------> {}:
numcards: ------------------------> {3}:
ipaddress: -----------------------> {0.0.0.0}:
alternateipaddress: --------------> {0.0.0.0}:
countryregion: -------------------> {us}:
primaryclocksource: --------------> {0/0/0/0/0}:
ringsource: ----------------------> {internalringsourcelabel}:
revertiveclocksource: ------------> {true}:
voicebandwidthcheck: -------------> {false}:
alarm-levels-enabled: ------------> {critical+major+minor+warning}:
userauthmode: --------------------> {local}:
radiusauthindex: -----------------> {0}:
secure: --------------------------> {disabled}: enabled
webinterface: --------------------> {enabled}:
options: -------------------------> {NONE(0)}:
reservedVlanIdStart: -------------> {0}:
```

```
reservedVlanIdCount: -------------> {0}:
snmpVersion: ---------------------> {snmpv2}:
persistentLogging: ---------------> {disabled}:
outletTemperatureHighThreshold: --> {65}:
outletTemperatureLowThreshold: ---> {-12}:
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

6- 1.2 Encryption-key Commands

encryption-key add

Adds an encryption key to the encryption-key profile.

Syntax: **`encryption-key add [rsa|dsa] [512|768|1024|2048]`**

Options:

rsa|dsa
Name and type of the encryption key.

512|768|1024|2048
The number of bytes the key is set to.

encryption-key delete

Deletes an encryption key from the encryption-key profile.

Syntax: **`encryption-key delete [rsa|dsa]`**

Options:

rsa|dsa
Name and type of the encryption key.

encryption-key renew

Regenerates a compromised encryption key.

Syntax: **`encryption-key renew [rsa|dsa]`**

Options:

rsa|dsa
Name and type of the encryption key.

encryption-key show

Displays the current encryption keys.

Syntax: **`encryption-key show`**

6- 1.3 DSA and RSA Keys

The MXK-F automatically creates a Digital Signature Algorithm (DSA), a standard for digital signatures, and supports RSA, an algorithm for public-key cryptography. The DSA and RSA host keys for the server are persistently

stored in the encryption-key profile. In order to manage the host keys, use the CLI command **encryption-key**.

RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key

When the system first boots, it will try to load the existing DSA and RSA keys. If they do not exist, the system creates a 512 bit DSA key.

The CLI **encryption-key** command can be used to view current keys, create a new key, regenerate keys that may have been compromised, and delete keys.

To create a new key enter:

```
zSH> encryption-key add rsa 1024
Generating key, please wait ... done.
```

Note: Generating keys is computationally intensive. The longer the key, the longer it takes to generate. Wait until the system shows that key generation is completed before you continue.

To view the new key just created enter:

```
zSH> encryption-key show
Index Type       Length
----- ---------- ------
  1   dsa           512
  2   rsa          1024
```

Note: The **encryption-key show** command displays the keys that were generated and are available for use. The command does not show the actual keys.

To regenerate a key that might have been compromised enter:

```
zSH> encryption-key renew dsa
Generating key, please wait ... done.
```

To delete an encryption key enter:

```
zSH> encryption-key delete dsa
```

6- 1.4 Secure Communications Between MXK-F and Servers

When the MXK-F is configured for secure mode, secure communication between the ZMS and other servers, and when communicating with the MXK-F for file transfers and or uploading bulk-statistics can be added via the use of digital keys (public and private) between systems.

The MXK-F can then log into the ZMS Server using zmsftp username without using a password, and all traffic between MXK-F and ZMS will be encrypted.

1 Create a RSA or DSA client key pair

a To see what is provided, first display the

By default, the MXK-F generates one server key (for SSH to the MXK-F):

```
zSH> encryption-key show
Index Type         Length Public Key MD5 Fingerprint
----- ------------ ------ -----------------------------------------------
 1   dsa             256 66:43:ec:a1:d2:39:4c:7d:3f:c5:7d:be:1e:d8:6f:7f
```

b Create the RSA or DSA client key pair:

```
zSH> encryption-key add rsa-client 1024
Generating key, please wait ... done.

zSH> encryption-key show
Index Type         Length Public Key MD5 Fingerprint
----- ------------ ------ -----------------------------------------------
  1   dsa             256 66:43:ec:a1:d2:39:4c:7d:3f:c5:7d:be:1e:d8:6f:7f
  4   rsa-client     1024 3e:2c:58:65:67:95:c5:4e:4d:82:6a:f4:47:10:2f:f9
```

2 Display the public key to copy and paste it to the SFTP servers ~/.ssh/authorized_keys file:

```
zSH>encryption-key show-pubkey rsa-client
ssh-rsa
AAAAAwEAAQAAAIBA1/SAhnZYqm/
fSA24BKoKLr3YGvDSqOLKQ6YIr6dOVkV3o9TeUOmz+JXq0zoUtv2AdMs200b0cbjQQRyuQ4x+5at0lkt5jurykOsotQYVRlHw/
jAxxy5zy5mySS8gdk/RQB6W6EUVmfSuWFqQESuz9OPHelgrsRU4DP68USgUiQ== Zhone-12865960
```

3 (Optional) The keys can be saved to a file in the MXK-F's flash memory and then copied/transferred to the server, e.g. using the **file upload** command.

```
zSH> encryption-key save-pubkey rsa-client
RSA public key save to id_rsa.pub
```

or

```
zSH> encryption-key save-pubkey rsa-client rsa_key.dat
RSA public key save to ra_key.dat

zSH> file upload 172.16.80.22 id_rsa.pub id_rsa.pub
Device is in secure mode, using SFTP protocol.
User name: zhone
Password:
Bytes copied: 228
File upload successful
```

6- 1.5 Tested MXK-F SSH Clients

Secure Shell (SSH) is a command interface and protocol for securely getting access to a remote computer. SSH commands are encrypted and secure in two ways. Both ends of the client/server connection are authenticated using a digital certificate, and passwords are protected by being encrypted. You can

now connect to an MXK-F using the SSH client of your choice to encrypt the session. The MXK-F supports SSH2 only with the following SSH clients:

- OpenSSH
  - cygwin
  - Linux
  - Solaris
- Putty
- Teraterm
- SecureCRT
- Absolute Telnet

6- 2 PORT ACCESS SECURITY

The MXK-F provides security capabilities on the UDP/TCP ports which the MXK-F uses for management. Use the **port-access** profile to define the UDP/TCP port and the IP address or IP address subnet that allows access to that port.

The port access security feature is a white list mechanism. If a host's IP address is not specified in a **port-access** profile, users from that host cannot access on that port.

The management ports are:

- Telnet, port 23
- SSH, port 22
- HTTP, port 80
- HTTPS, port 443
- SNMP, port 161

In order to restrict access to the SNMP port, there must be a rule to allow the MXK-F its own SNMP access. See *Creating a port-access Entry for the MXK-F to Maintain SNMP Access on page 140*.

By default, **port-access** profiles do not exist and all ports are open. After a **port-access** profile is configured for a port all other IP addresses or subnets are blocked. This restriction only takes effect after the first **port-access** profile is created.

Note: Port access security is not independent from enabling secure mode for SFTP and SSH in **system 0**. If secure is enabled to provide SSH and SFTP while limiting Telnet access, and you have provided access with the **port-access** profile for Telnet to a device (or range of devices), the device(s) will not have access.

Up to 100 **port-access** profile entries can be created on a SLMS device.

Procedure: *Creating port-access Profile Entries*

Create a **port-access** profile entry.

1. Create a new port-access entry by entering **new port-access n**, where n is an available entry ID number.
2. In the **portNumber** parameter enter the port number.
3. In the **srcAddr** parameter enter the IP address or first IP address of the subnet.
4. In the **netMask** parameter enter 255.255.255.255 for a single IP address mask, or a subnet mask for a subnet.

Procedure: *Creating a port-access Entry for a Specific IP Address*

Create a new **port-access** profile and specify the port number, host/network IP address to be granted access, and the one address netmask (255.255.255.255, which really means an exact mask of the IP address given) applied to the IP address to allow access to a single IP address.

Note: To create port access protection for both HTTP and HTTPS, create port access entries for port 80 and port 443.

```
zSH> new port-access 1
Please provide the following: [q]uit.
portNumber: ->  {0}: 80
srcAddr: --->  {0.0.0.0}: 172.16.42.1
netMask: --->  {0.0.0.0}: 255.255.255.255
.....................S=
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

This example creates **port-access** entry 1 on HTTP port 80 and allows the host 172.16.42.1 to have HTTP access to the MXK-F.

Procedure: *Creating a port-access Entry for a Subnet*

Create a new **port-access** profile and specify the Telnet port number, network address (enter for srcAddr field) to be granted access, and the network mask (enter for netMask field) applied to the network address to allow access to a range of IP addresses.

Note: Typically, only port 23 is used for Telnet access.

```
zSH> new port-access 2
Please provide the following: [q]uit.
portNumber: ->  {0}: 23
srcAddr: --->  {0.0.0.0}: 172.16.41.0
netMask: --->  {0.0.0.0}: 255.255.255.0
```

```
.....................S=
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

This example creates **port-access** entry 2 on Telnet port 23 and allows hosts on the 172.16.41.0 network to Telnet to the MXK-F.

Procedure: *Displaying port-access Profile Entries*

Display configured **port-access** profile entries with the **list** command:

```
zSH> list port-access
port-access 1
1 entry found.
```

Procedure: *Modifying port-access Profile Entries*

Modify a configured **port-access** profile entry with the **update** command. This example changes the entry's source IP address to 172.16.40.0:

```
zSH> update port-access 2
portNumber: ->  {23}
srcAddr: --->  {172.16.41.0} 172.16.40.0
netMask: --->  {255.255.255.0}
1 entry found.
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
Updated record saved.
```

Procedure: *Displaying port-access Profile Entries*

To display configured **port-access** profile entries use the **list** command:

```
zSH> list port-access
port-access 1
1 entry found.
```

Procedure: *Creating a port-access Entry for the MXK-F to Maintain SNMP Access*

Create a new **port-access** profile and specify the SNMP port number (161) then 127.0.0.0 as the IP address for the subnet and a subnet mask of 255.0.0.0.

```
zSH> new port-access 10
Please provide the following: [q]uit.
portNumber: ->  {0}: 161
srcAddr: --->  {0.0.0.0}: 127.0.0.0
netMask: --->  {0.0.0.0}: 255.0.0.0
.....................S=
Save new record? [s]ave, [c]hange or [q]uit: s
New record saved.
```

6- 3 RADIUS SUPPORT

The MXK-F supports local and RADIUS (Remote Authentication Dial In User Service) access authentication. The MXK-F can be configured for local authentication, RADIUS authentication, or RADIUS then local authentication. RADIUS users are configured with the Service-Type attribute as Administrative-User or NAS-Prompt-User. RADIUS is used for only login authentication, not severity levels.

Table 16 shows the mapping of service-type to MXK-F permissions.

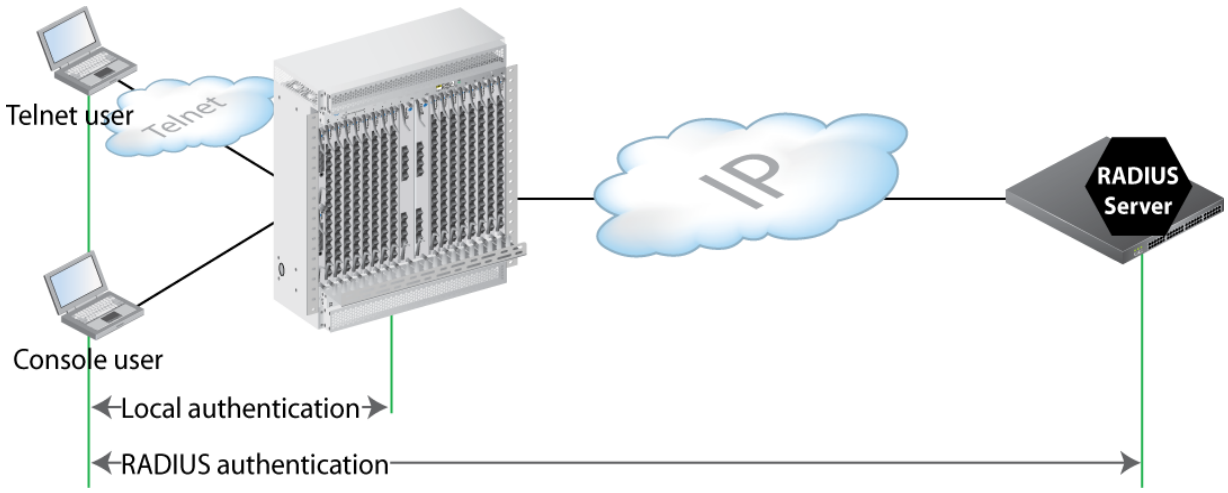
Table 16: Service Type Mapping to MXK-F Permissions

Service-Type Attribute	MXK-F permissions
Administrative-User	admin, zhonedebug, voice, data, manuf, database, systems, tools, useradmin
NAS-Prompt-User	admin, voice, data, manuf, database, systems, tools, useradmin

When establishing a connection to the MXK-F with RADIUS authentication, the MXK-F passes RADIUS information securely to the RADIUS server. The RADIUS server then authenticates the user and either allows or denies access to the MXK-F. If access is denied and the local authentication option is also configured, the MXK-F then authenticates access based on the locally configured users and passwords. For logins and failed logins, a console message is generated with user ID and IP address of the device from which the login originated. Failed logins also are logged as alert level messages in the MXK-F system log file.

By default, RADIUS access uses the UDP port 1812 for authentication.This parameter can be changed in the radius-client profile.

Figure 7: MXK-F RADIUS Authentication



Note: Follow the RADIUS server guidelines for RADIUS configuration instructions. For example, when using the MXK-F with the FreeRadius server:

- Create only one entry in the *clients.conf* file for each subnet or individual MXK-F. For individual MXK-Fs, the IP in this file must match the IP address of the outbound interface used by the MXK-F to connect to the RADIUS server.
- The MXK-F uses the value stored in the RADIUS *system.sysname* file for the NAS-Identifier attribute.
- The shared-secret in the MXK-F **radius-client** profile, must exactly match the shared-secret in the RADIUS client entry.

Procedure:

Configuring RADIUS Support

The MXK-F can be configured for local authentication, RADIUS authentication, or RADIUS then local authentication. Multiple **radius-client** profiles can be defined using the index and subindex numbers. This index scheme can be used to create index numbers for groups of RADIUS servers. When an index number is specified in the system profile, the MXK-F attempts authentication from each RADIUS server in that group in sequential order of the subindex numbers.

To configure RADIUS support:

Note: Before beginning this procedure, ensure that the MXK-F has IP connectivity to the RADIUS server.

1. Update the RADIUS server with settings for the DZS prompts.
2. Create a **radius-client** profile on the MXK-F with the desired index number and RADIUS settings for server name, shared secret, number of retries, and other parameters. The first number in the index is used to group **radius-client** profiles so multiple profiles can be assigned to an MXK-F. The second number in the index specifies the order in which **radius-client** profiles are referenced. This example specifies the **radius-client** *1/1* with server name *radius1* and a **shared-secret** of *secret*. The index *1/1* specifies that this profile is the first profile in group *1*.

   A DNS resolver record must be configured in the system to resolve (find and translate) the server name and IP address. If a DNS resolver record is not available, create a record for the IP address of the server (see *DNS Resolver Configuration* on page 147).

```
zSH> new radius-client 1/1
Please provide the following: [q]uit.
server-name: ----> {}: radius1.test.com [DNS resolver must be configured in the system.]
udp-port: -------> {1812}:
shared-secret: --> {** password **}: secret
retry-count: ----> {5}:
retry-interval: -> {1}:
.....................
```

```
Save new record? [s]ave, [c]hange or [q]uit: s
Record created.
```

Another method to reference the RADIUS server is by specifying the IP address. This example specifies the **radius-client** *1/1* with server IP address 172.24.36.248 and a **shared-secret** of *secret*. The index *1/1* specifies that this profile is the first profile in group *1*.

```
zSH> new radius-client 1/1
Please provide the following: [q]uit.
server-name: ----> {}: 172.24.36.248
udp-port: -------> {1812}:
shared-secret: --> {** password **}: secret
retry-count: ----> {5}:
retry-interval: -> {1}:
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
Record created.
```

3 Create another **radius-client** profile on the MXK-F with the desired RADIUS settings for server name, shared secret, number of retries, and other parameters. This example specifies the **radius-client** *1/2* with server IP address 172.24.36.249 and a **shared-secret** of *secret*. The index *1/2* specifies that this profile is the second profile in group *1*.

```
zSH> new radius-client 1/2
Please provide the following: [q]uit.
server-name: ----> {}: 172.24.36.249
udp-port: -------> {1812}:
shared-secret: --> {** password **}: secret
retry-count: ----> {5}:
retry-interval: -> {1}:
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
Record created.
```

Create additional **radius-client** profiles for each additional RADIUS server to be assigned to this MXK-F.

4 In the system profile on the MXK-F, set the desired user authentication method and specify the index of the radius profile to use. This examples specifies the **radiusauthindex** of *1*. This index is configured with two **radius-client** profiles (*1/1*, *1/2*). The MXK-F first attempts authentication using the server specified in **radius-client** *1/1*. If this authentication fails, the MXK-F attempts authentication using **radius-client** *1/2* server. If this authentication also fails, the MXK-F then attempts authentication based on the authentication mode setting in the system profile. This example uses **radiusthenlocal**.

Caution: If the *radius* authentication mode is used, local authentication is disabled so the MXK-F may become inaccessible if IP connectivity to the RADIUS server is lost or other changes prevent the MXK-F from receiving RADIUS authentication.

```
zSH> update system 0
system  0
Please provide the following: [q]uit.
syscontact: ----------->  {}:
sysname: ------------->  {}:
syslocation: ---------->  {}:
enableauthtraps: ------>  {disabled}:
setserialno: ---------->  {0}:
zmsexists: ------------>  {false}:
zmsconnectionstatus: -->  {inactive}:
zmsipaddress: --------->  {0.0.0.0}:
configsyncexists: ----->  {false}:
configsyncoverflow: --->  {false}:
configsyncpriority: --->  {high}:
configsyncaction: ----->  {noaction}:
configsyncfilename: --->  {}:
configsyncstatus: ----->  {syncinitializing}:
configsyncuser: ------->  {}:
configsyncpasswd: ----->  ** private **
numshelves: ----------->  {1}:
shelvesarray: --------->  {}:
numcards: ------------->  {3}:
ipaddress: ------------>  {0.0.0.0}:
alternateipaddress: --->  {0.0.0.0}:
countryregion: -------->  {us}:
primaryclocksource: --->  {0/0/0/0/0}:
ringsource: ----------->  {internalringsourcelabel}:
revertiveclocksource: ->  {true}:
voicebandwidthcheck: -->  {false}:
alarm-levels-enabled: ->  {critical+major+minor+warning}:
userauthmode: --------->  {local}: radiusthenlocal
radiusauthindex: ------>  {0}: 1
secure: --------------->  {disabled}:
webinterface: --------->  {enabled}:
options: -------------->  {NONE(0)}:
reservedVlanIdStart: -->  {0}:
reservedVlanIdCount: -->  {0}:
snmpVersion: ---------->  {snmpv2}:
persistentLogging: ---->  {disabled}
outletTemperatureHighThreshold: ->  {65}
outletTemperatureLowThreshold: -->  {-12}
.................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.
```

After completing the RADIUS configuration, the MXK-F displays console messages for RADIUS login and logout activity.

5 For users logging in through RADIUS, the CLI prompt appears as *username@sysname*. For example, the prompt for a basic user on an MXK-F using the default DZS MXK-F system name will appear as *basicuser@DasanZhone mxk*. The system name is configured using the **sysname** parameter in the **system 0** profile (see Commands: SysName and SysNameAlias, page 98).

```
basicuser@DasanZhone mxk> update system 0
system  0
```

```
Please provide the following: [q]uit.
syscontact: ----------------------> {}:
sysname: -------------------------> {}: MXK-F_A3-R2
syslocation: ---------------------> {}:
< skip >
sysNameAlias: --------------------> {}: Aisle3_Rack2
.....................
Save changes? [s]ave, [c]hange or [q]uit: s
Record updated.

basicuser@MXK-F_A3-R2>
```

CHAPTER 7 DNS RESOLVER

This chapter describes MXK-F DNS resolver.

7- 1 DNS RESOLVER CONFIGURATION

Domain Name System (DNS) maps domain names to IP addresses, enabling the system to reach destinations when it knows only the domain name of the destination. In other words, you can use **ping** and a name instead of an IP address. DNS configuration uses the following profiles:

- **resolver**—Configures the global DNS resolver, including the DNS search order, default domain name, and list of nameserver addresses. The DNS settings in this record can be used for local applications by administrators on the system, such as **traceroute** or **ping**.
- **host-name**—A replacement for the UNIX local hosts table. Up to four host aliases can be defined for each host entry. Settings in the **resolver** record determine whether the hosts table is searched.

Table 17 describes the configurable parameters for the **resolver** profile (all others should be left at their default values):

Table 17: Configurable Resolver Parameters

Parameter	Description
query-order	The kind of resolver query for this routing domain. Values: **hosts-first** searches the local hosts table first then the list of nameservers. **dns-first** searches the list of nameservers first then the local hosts table. **dns-only** searches only the list of nameservers. Default: **hosts-first**
domain	The routing domain to which this host parameter applies. The default is an empty string. The only routing domain supported is domain 1.
first-nameserver	The IP address of the first or primary nameserver for this routing domain. The default value is 0.0.0.0.

Table 17: Configurable Resolver Parameters (Continued)

Parameter	Description
second-nameserver	The IP address of the second or secondary nameserver for this routing domain. This nameserver is queried if the first nameserver cannot resolve the query. The default value is 0.0.0.0.
third-nameserver	The IP address of the third or tertiary nameserver for this routing domain. This nameserver is queried if the first nameserver cannot resolve the query. The default value is 0.0.0.0.

The following example creates a **resolver** record for a routing domain:

```
zSH> new resolver 1
Please provide the following: [q]uit.
query-order: -------> {hosts-first}:
domain: ------------> {}: dzsi.com
first-nameserver: --> {0.0.0.0}: 192.168.8.21
second-nameserver: -> {0.0.0.0}: 201.23.20.2
third-nameserver: --> {0.0.0.0}:
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
Record created.
```

Another way to create DNS is by creating a **hosts** profile after the **resolver** profile is created. The syntax is **new host-name** *routingdomain*/*ipoctet1*/*ipoctet2*/*ipoctet3*/*ipoctet4*.

Table 18 describes the configurable parameters in the **host-name** profile (all others should be left at their default values).

Table 18: Configurable Parameters in the host-name Profile

Parameter	Description
hostname	Client host name (if any) that the client used to acquire its address (default = empty string).
hostalias1	Host name alias for the specified host (default = empty string).
hostalias2	Secondary host name alias for specified host (default = empty string).
hostalias3	Tertiary host name alias for the specified host (default = empty string).
hostalias4	Quaternary host name alias for the specified host (default = empty string).

```
zSH> new host-name 1/192/168/8/32
Please provide the following: [q]uit.
hostname: ---> {}: www.dzsi.com
ipaddress: --> {0.0.0.0}: 192.168.8.32
hostalias1: -> {}: engineering.dzsi.com
hostalias2: -> {}: marketing.dzsi.com
hostalias3: -> {}: sales.dzsi.com
hostalias4: -> {}: gss.dzsi.com
.....................
Save new record? [s]ave, [c]hange or [q]uit: s
Record created.
```

CHAPTER 8 CPE MANAGER

This chapter describes the CLI-based CPE Manager system. In the explanatory text of this section the terms “CPE” and “ONT” are synonymous. However, these two terms cannot be interchanged in the command syntax (e.g. “ont-mgr” cannot be used in place of the CLI syntax = “cpe-mgr”).

Note: When an ONT is configured via CLI, the local config for the ONT is static. The preferred config method, for customers that use ZMS, is to use ZMS (instead of CLI) - this creates a local config for the ONT that is dynamic with a 24 hour lease. The creation in ZMS is automatic by right clicking on the ONT and selecting Launch ONT WebUI.

For details on using ZMS, refer to the *ZMS Administrator's Guide*, *ZMS Installation Guide* and the *NetHorizhon User's Guide*.

8- 1 CPE MANAGER CONFIGURATION

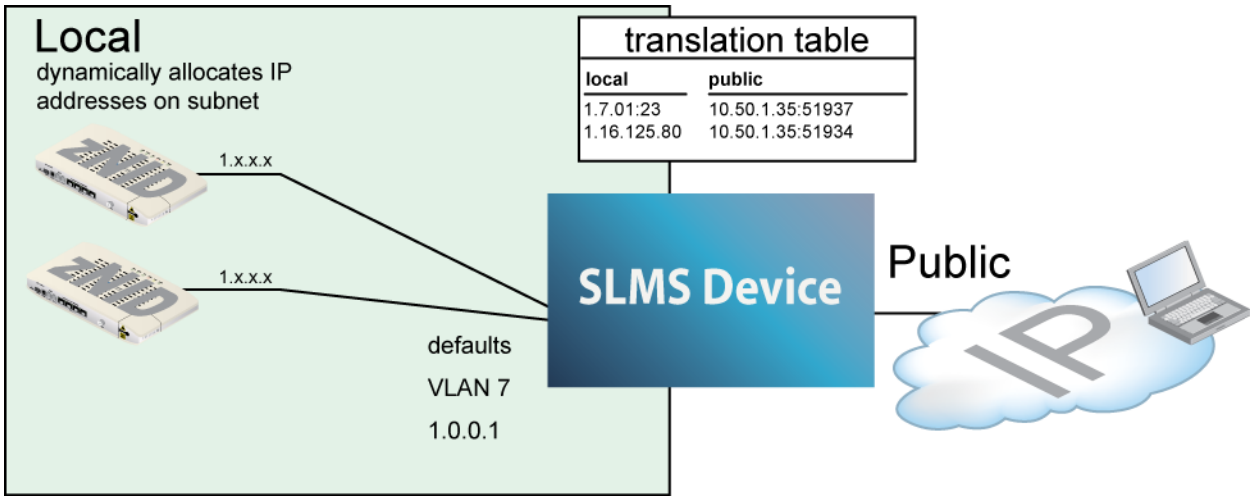
This section describes:

8- 1.1 CPE Manager Overview

The MXK-F’s CPE Manager provides a means for managing customer premises equipment (CPE) devices without requiring extra routable IP addresses to reach these CPE end-points. While the CPE Manager is specifically designed for DZS’s zNID family of CPE products, CPE Manager can be used with any CPE device which supports receiving an IP address via DHCP on a VLAN.

In many service provider networks, the increasing use of IP-aware CPE devices creates an operational challenge for service providers because the number of devices which require IP addresses cause IP address space depletion, making it hard to assign routable addresses for these devices.

A solution to this problem is the CPE Manager. CPE Manager adds proxy capability, allowing one IP interface on the DZS central office device to provide IP access to all the subtended CPE devices connected to it. This one IP interface is created on an upstream port which is routable on the service providers management network, and it provides the translation when forwarding packets to and from managed CPE devices. In this way, one MXK-F IP address can be used for all CPE management without having to consume IP address space or having to add network routes for reachability of line side CPE devices.



CPE Manager is supported on all MXK-F line cards.

The MXK-F manages ONTs using internal ONT Port/Interface Names (see: Port/Interface Naming Convention). A local (non-public) IP address can also be configured for each ONT. These ONT IP addresses can then be used from a remote location to manage each ONT through the CPE Manager system.

8- 1.2 Manage a CPE using a Local (non-public) IP Address

To access a CPE configured using CPE Manager, access the MXK-F through its IP address, however, instead of using the protocol ports, use the CPE's base public port plus an offset to the specific port used for the protocol desired. Supported protocols include Echo, FTP (data), FTP (control), SSH, Telnet, HTTP, SNMP and HTTPS.

To select the ports to make available the **cpe-mgr add** command has several options depending on the selection of **the compact and security parameters:**

- **compact [full | partial | none]**

Selection of the **compact** mode defines how many ports may be accessed using the NAT-PAT binding, the more ports are accessed per device, the fewer devices that will be able to be accessed.

- **security [enabled | disabled | default]**

  enabled uses SSH and HTTPS network protocols.

  disabled uses Telnet and HTTP protocols.

  default uses the default settings of the chassis.

A list of offsets for public ports based on the compact and security mode is given in Table 19. For more information about how offsets work, see *CPE Manager Additional Information on page 159*.

The defaults for compact mode is full mode (the three port mapping). For security mode, the default is **default**, which uses the MXK-F chassis security settings in **system 0**. For additional information about security and **system 0**, see *Chapter 6, Security,* on page 133.

Table 19: Offsets for Public Ports

Public port offset	Type	Name	Compact & Security Modes				
			Full		Partial		None
			Secure Enabled	Secure Disabled	Secure Enabled	Secure Disabled	N/A (all ports)
7	TCP, UDP	ECHO	+0	+0	+0	+0	+0
20	TCP	FTP - data	—	—	—	—	+1
21	TCP	FTP - control	—	—	—	—	+2
22	TCP, UDP	SSH	+1	—	+1	—	+3
23	TCP, UDP	Telnet	—	+1	—	+1	+4
80	TCP	HTTP	—	+2	—	+3	+5
81	TCP	HTTP	—	—	—	—	+6
161	TCP, UDP for partial and none UDP for full compact mode	SNMP	+2	+2	+2	+2	+7
162	UDP	SNMP traps (upstream only)	+0	+0	+3	+3	+1
443	TCP	HTTPS	+2	—	+3	—	+8

The default, local, class A network is set up as 1.0.0.1/8 on VLAN 7. All ZNID ONTs ship with VLAN 7 pre-configured for this use. To change the default CPE management VLAN requires changing the MXK-F CPE management VLAN and requires manually changing the management VLAN

that is used by each ZNID (see *Changing the VLAN of the Local Network* on page 154).

The IP addresses given to CPEs follow the general guidelines:

```
<Class A network>.<Slot>.<Port number: higher order byte>.<Port number: lower order byte>
```

Note that the GPON format has the port/subport encoded into the IP address which allows 12 bits for a subport and 4 bits for the port number:

```
<class A>.<slot>.<subport upper 8 bits>.<subport lower 4 bits * 16 + port>
```

The *1-1-4-501/gponport* yields an IP address of 1.1.31.84.

Procedure: *Configuring the CPE Manager for Active Ethernet*

Adding the CPE Manager.

1 View the MXK-F Management system IP addresses that will also be used by the CPE Manager system (the MXK-F management IP addressing is configured here: Change MGMT Port IP Address, page 32). In this case, 10.50.1.35 is used.

```
zSH> interface show
2 interfaces
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24       00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

2 Verify the network route.

```
zSH> route list
Domain   Dest            Mask            Nexthop          IfNum    Cost       Enable
-------------------------------------------------------------------------------------
1        0.0.0.0         0.0.0.0         10.50.1.254      0        1          enabled
```

3 Add a “public” address to the CPE Manager for the core/network side.

```
zSH> cpe-mgr add public 10.50.1.35
 CPE Manager using 10.50.1.35 for public interface.
```

4 Add the CPE Manager to an Active Ethernet port.

```
zSH> cpe-mgr add local 1-7-1-0/eth
  Created CPE Management interface: 1-7-1-0-eth-7/ip
```

Note that a default “local” IP address is automatically created for the CPE Manager system if you do not first manually assign a local address.

```
zSH> cpe-mgr show
        CPE Manager public side interface:
                IP:      10.50.1.35
        CPE Manager local management network:
                IP:      1.0.0.1/8 (default) (active)
                VlanID:  7 (default)
        Managed CPE Interface Configuration:
```

```
 Interface               Local IP        ECHO   FTP   SSH  Telnt HTTP  SNMP  HTTPS
-------------------------------------------------------------------------------------
1-7-1-0/eth[UP]          1.7.0.1         51927   -     -   51928 51929 51929   -
```

Procedure: *Configuring the CPE Manager for GPON*

Adding the CPE Manager.

1 View the MXK-F Management system IP addresses that will also be used by the CPE Manager system. In this case, 10.50.1.35 is used.

```
zSH> interface show
2 interfaces
Interface      Status  Rd/Address            Media/Dest Address      IfName
-------------------------------------------------------------------------------
1/m1/1/0/ip    UP      1 10.50.1.35/24       00:01:47:79:dd:08       ethernetm-1
1/m1/6/0/ip    UP      1 10.50.2.35/24       00:01:47:7f:e1:b2       ipobridge-3002
-------------------------------------------------------------------------------
```

2 Verify the network route.

```
zSH> route list
Domain   Dest             Mask             Nexthop          IfNum    Cost       Enable
-------------------------------------------------------------------------------------
1        0.0.0.0          0.0.0.0          10.50.1.254      0        1          enabled
```

3 Add a “public” address to the CPE Manager for the core/network side.

```
zSH> cpe-mgr add public 10.50.1.35
 CPE Manager using 10.50.1.35 for public interface.
```

4 Add the CPE Manager to a GPON port by adding it to at least one ONT on that GPON port.

In this GPON example the GEM Port range 501 to 628 is assumed to be reserved for the CPE Manager GEM Port for each ONU. The GEM ID is composed of the GEM base value 500 plus the ONU ID, which in this example is ONU 1 (500 plus 1 => **501/gponport**). This is a common GEM port assignment method. See: Port/Interface Naming Convention.

```
zSH> cpe-mgr add local 1-15-16-501/gponport gtp 512
  Created CPE Management interface: 1-15-16-501-gponport-7/ip
```

Note: For an equivalent XGS-PON example, the GEM Port range 1101 to 1356 can be reserved for the CPE Manager GEM Port for each ONU. The GEM ID is composed of the GEM base value 1100 plus the ONU ID. The CPE Manager port for ONU 1 would result in **1101/gponport** (1100 + 1).

5 View the CPE Manager configuration.

```
zSH> cpe-mgr show
        CPE Manager public side interface:
                IP:      10.50.1.35

        CPE Manager local management network:
```

```
                IP:      1.0.0.1/8 (default) (active)
                VlanID:  7 (default)

        Managed CPE Interface Configuration:
 Interface               Local IP        ECHO   FTP   SSH  Telnt HTTP  SNMP  HTTPS
-------------------------------------------------------------------------------------
1-7-1-0/eth[UP]          1.7.0.1         51927   -     -   51928 51929 51929   -
1-15-16-501/gponport[UP] 1.16.125.80     51930   -     -   51931 51932 51932   -
```

Procedure: *Changing the VLAN of the Local Network*

Ordinarily the default settings are acceptable. However if you need to change the default class A network or VLAN ID you can use the following command. If you change this MXK-F CPE management VLAN setting you must also change the management VLAN setting in all of the CPEs. VLAN 7 is the factory default management VLAN setting of DZS zNIDs.

To change the VLAN ID for the CPE Manager local network:

```
zSH> cpe-mgr add local vlan <vlan id used by the MXK-F for CPE Management>
```

To manually set the VLAN ID back to the default, enter:

```
zSH> cpe-mgr add local vlan 7
```

Note: DZS does not recommend changing the VLAN manually because DZS CPE and zNID products use VLAN 7 as the factory default management VLAN.

Procedure: *Changing the class A Network used as the CPE Manager Local Network*

The default network settings should be acceptable. However, if you need to change the default class A network setting, use the **cpe-mgr add local network** command. If you want to change network settings after CPEs are attached and configured, you must delete them before making the changes:

To manually set the local network setting enter:

```
zSH> cpe-mgr add local network <class A network used internally for all managed CPEs>
```

To manually set the local network back to the default, enter:

```
zSH> cpe-mgr add local network 1.0.0.1
```

Note: You can manually set the local network settings only when there are no CPE devices configured on the network.

By default we use the 1.0.0.0 class A network. In other words, a class A network is one that has an 8 bit mask which means only the first byte of the IP address is common between nodes in the network. If you execute the following command: **cpe-mgr add local network 2.0.0.0**, the class A network will be changed and all local IPs will start with 2.

8- 1.3 View and Ping the CPE Manager Port/Interfaces

The **cpe-mgr show** command displays all interfaces configured for CPE Manager, and provides a mapping between the interface and local IP address along with the various protocol ports.

```
zSH> cpe-mgr show
        CPE Manager public side interface:
                IP:      10.50.1.35
        CPE Manager local management network:
                IP:      1.0.0.1/8 (default) (active)
                VlanID:  7 (default)
        Managed CPE Interface Configuration:
 Interface               Local IP        ECHO   FTP   SSH  Telnt HTTP  SNMP  HTTPS
-------------------------------------------------------------------------------
1-7-1-0/eth[UP]          1.7.0.1         51924   -     -   51925 51926 51926   -
1-15-16-501/gponport[UP] 1.16.125.80     51927   -     -   51928 51929 51929   -
```

The **cpe-mgr show local** command provides information for an interface.

```
zSH> cpe-mgr show local 1-15-16-501/gponport
        Public IP address: 10.50.1.35
        Public Access Port:
                Protocol      Port
                 ECHO        51927
                 SNMP Traps  51927
                 Telnet      51928
                 HTTP        51929
                 SNMP        51929
        Local IP Address: 1.16.125.80
```

Use the **ping** command to verify the network connection.

```
zSH> ping 1.16.125.80
PING 1.16.125.80: 64 data bytes
!!!!!
----1.16.125.80 PING Statistics----
5 packets transmitted, 5 packets received
round-trip (ms)  min/avg/max = 0/1/5

zSH> cpe-mgr show local 1-7-1-0/eth
        Public IP address: 10.50.1.35
        Public Access Port:
                Protocol      Port
                 ECHO        51924
                 SNMP Traps  51924
                 Telnet      51925
                 HTTP        51926
                 SNMP        51926
        Local IP Address: 1.7.0.1

zSH> ping 1.7.0.1
PING 1.7.0.1: 64 data bytes
!!!!!
----1.7.0.1 PING Statistics----
5 packets transmitted, 5 packets received
round-trip (ms)  min/avg/max = 0/48/240
```

8- 1.4 Delete Local & Public IP Addresses from the CPE Manager

If necessary, delete the CPE Manager.

Procedure: *Deleting CPE Manager Local Side*

1 View the CPE manger configuration.

```
zSH> cpe-mgr show
        CPE Manager public side interface:
                IP:      10.50.1.35
        CPE Manager local management network:
                IP:      1.0.0.1/8 (default) (active)
                VlanID:  7 (default)
        Managed CPE Interface Configuration:
 Interface               Local IP        ECHO   FTP   SSH  Telnt HTTP  SNMP  HTTPS
-------------------------------------------------------------------------------------
1-7-1-0/eth[UP]          1.7.0.1         51924   -     -   51925 51926 51926   -
1-15-16-501/gponport[UP] 1.16.125.80     51927   -     -   51928 51929 51929   -
```

2 Delete the local interface.

```
zSH> cpe-mgr delete local 1-15-16-501/gponport

zSH> cpe-mgr delete local 1-7-1-0/eth
```

3 Verify CPE Manager.

```
zSH> cpe-mgr show
        CPE Manager public side interface:
                IP:      10.50.1.35
        CPE Manager local management network:
                IP:      1.0.0.1/8 (default) (active)
                VlanID:  7 (default)
        No CPE's currently configured in CPE Manager.
```

Procedure: *Deleting CPE Manager Public Side*

1 Delete the CPE Manager from the public side IP address.

```
zSH> cpe-mgr delete public
 CPE Manager public interface no longer configured.
```

2 Verify the deletion.

```
zSH> cpe-mgr show
        CPE Manager public side interface: Not Configured
        CPE Manager local management network:
                IP:      1.0.0.1/8 (default) (active)
                VlanID:  7 (default)
        No CPE's currently configured in CPE Manager.
```

8- 2 CPE MANAGER TROUBLE SHOOTING

To verify or troubleshoot CPE Manager, you should understand what the two commands for CPE Manager do. The first **cpe-mgr add public** command

- Sets **natenabled** to “yes” in the **ip-interface-record** for the public address (in our example, the 10.50.1.35 address).

When using the defaults and the local network has not been created, the second command, **cpe-mgr add local**:

- Creates a floating **ip-interface** record with IP address of 1.0.0.1 (only created if the defaults are being used and if the record does not already exist. In other words, the first **cpe-mgr add local** if the record wasn’t created manually).
- Creates an **ip-unnumbered-record** for the floating ip-interface record (only created if the defaults are being used and if the record does not already exist. In other words, the first **cpe-mgr add local** if the record wasn’t created manually).
- Creates a **dhcp-server-subnet** for the 1.0.0.0 network (only created if the defaults are being used and if the record does not already exist. In other words, the first **cpe-mgr add local** if the record wasn’t created manually)
- Creates a host **ip-interface-record** for the CPE on interface.

  Assigns a local IP address based on the interface description (not routable, but may be reached from the local network, or by Telnet to the MXK-F, then Telnet from the MXK-F to the device)
- Creates a **pat-bind** profile of type **cpemgr** or **cpemgrsecure**

Note: The **ip-interface-record** created is not a normal “host” record and cannot be seen using the **host show** command.

The **pat-bind 2** profile contains the local IP address (1.7.0.1) and the CPE base port (51927) for the Ethernet CPE Manager and the pat-bind 3 profile contains the local IP address 1.16.125.80 and CPE base port 51930 for the GPON CPE Manager:

```
zSH> list pat-bind
pat-bind  2
pat-bind  3
2 entries found.

zSH> get pat-bind 2
pat-bind  2
public-ipaddr: --> {10.50.1.35}
public-port: ----> {51927}
local-ipaddr: ---> {1.7.0.1}
local-port: -----> {3}
portType: -------> {cpemgr}

zSH> get pat-bind 3
pat-bind  3
```

```
public-ipaddr: --> {10.50.1.35}
public-port: ----> {51930}
local-ipaddr: ---> {1.16.125.80}
local-port: -----> {3}
portType: -------> {cpemgr}
```

The local address which is given is based on the interface in the form:

```
<local class A network>.<slot>.<port HI byte>.<port LO byte>
```

The local IP address (as shown above in the **pat-bind 2** profile) is 1.7.0.1. If you need to verify this number, enter **get pat-bind** <#>.

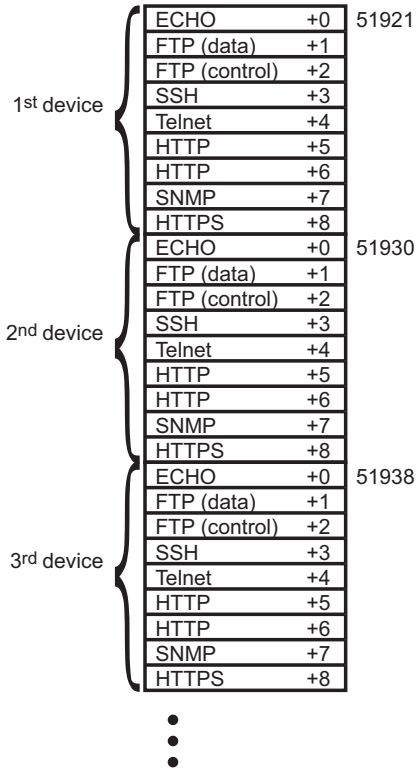
Note that GPON format allows 12 bits for a subport and 4 bits for the port number:

```
<class A>.<slot>.<subport upper 8 bits>.<subport lower 4 bits * 16 + port>
```

The *1-16-15-501/gponport* yields a local IP address of 1.16.125.80.

8- 3 CPE MANAGER ADDITIONAL INFORMATION

The first device will be accessible by the MXK-F's public IP address and the CPE base port. The CPE base port for the first device is 51921. To reach one of the well known ports you then give the offset for the public port. Well known port (7) is for echo which has an offset of zero.



Note: The examples use compact mode none. See *Configuring the CPE Manager for Active Ethernet on page 152* and *Configuring the CPE Manager for GPON on page 153*. Using different variations of **compact** mode and **security** mode requires different offsets as shown in Table 19.

To telnet to the first CPE via the well known port, 23, you would use the CPE base port plus the public port offset of 4; You would use the MXK-F's address (192.168.254.1), then 51925 (51921 + 4) to Telnet to the device. From a Unix or DOS prompt it would look like

```
telnet 192.168.254.1 51925
```

To access the second device you need to start with the CPE base port for that device. Each device consumes nine public ports, so the first device has a port range from 51921 - 51929, the second device has a port range from 51930 - 51938, the third from 51939 - 51947 and so on.

To access the HTTP port on the third device from a browser, you would start from the first public port address 51921 + 18 (the 51921 start point plus two

times nine for the first two devices to get to the third device range) + 5 (to get to port 80, a HTTP port) or 51944.

As CPE devices are deleted or added, holes will form in the list of CPE devices, so the order eventually becomes arbitrary, but is used in the discussion to elucidate how the mechanism works.

CPE base port and information for added devices is shown in the **cpe-mgr show** display.

INDEX

A

C

D

F

G

H

I

L

M

N

O

P

R

S

T

U

V

W

X

Z