

FMADIO100G Gen2

# Introduction

Hardware

FMADIO 100G Gen2 system provides lossless full 100Gbps 149Mpps line rate packet capture system in a compact 1U form factor.

## Features

- 100Gbps sustained 24/7 packet capture
- 200Gbps Burst Packet Capture
- x2 QSFP28 Ports
- SR4 / LR4 / CX4 and FEC support
- 200Gbps line rate packet generator
- 16-256TB RAID5 storage
- Hardware Timestamping
- Pre Capture Filtering
- Inline Packet Capture
- Packet Slicing
- Simple easy to use HTML GUI
- Full scriptable JSON REST API

## Specification



- Network Sustained Capture Speed:
  - 100Gbps Line Data rate
  - 148.88Mpps Packet Rate
- Network Capture Ports
  - x2 QSFP28 100G
  - 2x100G
  - 2x40G
  - 8x10G
  - 4x25G
- Packet Size
  - 64B- 9200B
- Management Interface
  - 2 x 10G SFP+
  - 2 x 40G QSFP+
  - 4 x 10G QSFP+ (with breakout cable)
  - 2 x 10/100/1000M RJ45
- Hardware Time Stamp Resolution
  - 3.2 ns
- Clock Synchronization
  - PTPv2
  - NTP
  - PPS
- Disks
  - 10 x U.2 2.5" Disks
- Storage Size
  - 16TB - 156tB
- Storage Redundancy
  - Full 100Gbps RAID5
- Chassis Size
  - 1U Rack muontable
  - 482 mm x 787mm x 44mm
  - 19" x 31" x 1.75"
- Rack mount
  - 31" Tool less rails
- Cooling
  - x12 Hot swap 40mm fans

- Power
  - Dual 850W Power Supply
  - Hot Swap Power Supply
  - (Operates with single PSU)
- Weight
  - 25.8KG
  - 54lbs

# System SKU

FMADIO 100G Gen2 System,

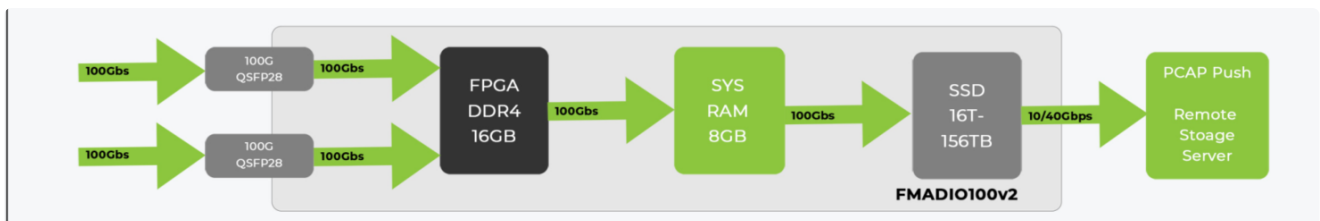
## SKU Capture (64G RAM + 32 CPUs)

- FMADIO100GV2-1U-16T
- FMADIO100GV2-1U-38T
- FMADIO100GV2-1U-76T
- FMADIO100Gv2-1U-156T

## SKU Analytics ( 576GB RAM + 96CPU)

- FMADIO100GV2-1U-16T-ANALYTICS
- FMADIO100GV2-1U-38-ANALYTICS
- FMADIO100GV2-1U-76T-ANALYTICS
- FMADIO100Gv2-1U-156T-ANALYTICS

## High Level Architecture



# Network Port Layout

## FMADIO100G Gen2 Capture

Layout of the network ports is as follows. for FMADIO100G v2 1U Capture System (10G SFP+ management)



FMADIO100G Gen2 1U Capture Port Layout

## FMADIO100G Gen2 Analytics

Layout of the network ports for FMADIO100G v2 1U Analytics System (40G QSFP+ Management)



FMADIO100G Gen2 1U Analytics Port Layout

## Port Description

### 1Gbe Management Ports

Systems shipped Prior to April 2021, default 1G management port is "L2"

System shipped after April 2021 default 1G management port is "L1"

This was made as L1 port can bridge the IPMI/BMC port (single RJ45 connection for both Server and BMC), the L2 port can not bridged. Please contact [support@fmad.io](mailto:support@fmad.io) on how to swap.

### IPMI Port

The IPMI Port is used for out of band communication with the system. It allows power on/off and KVM capabilities. Highly recommend connecting this. Default IP address is 192.168.0.93/24

## Management Port

The Management ports can be 10G SFP+ or 40G QSFP+ depending on the system configuration. These can be run in standard, or link bonded / redundant setup.

FMADIO100Gv2 Capture systems,

Port Count	Port Speed	Interface
2	10G	SFP+
2	1G	SFP

FMADIO100G Gen2 Analytics Systems

Port Count	Port Speed	Interface
2	40Gbps	QSFP+
4	10Gbps	QSFP+ Breakout Cables

## Capture Port

Capture ports can be configured in the following way

Port Count	Port Speed	Interface
2	100Gbps	QSFP28 (FEC or no FEC)
2	40Gbps	QSFP+
8	25Gbps	QSFP28 (Not released yet)
8	10Gbps	QSFP+ Breakout Cables

## PPS Connector

The PPS connector is a 1 Pulse Per Second time synchronization cable. It runs on a 3.3V trigger signal the interface is SMA Coaxial cable

## Serial Port

Standard RJ45 Serial port connector

# Hardware

FMADIO 100G Hardware looks as follows



FMADIO100G Gen2 1U Packet Capture System



FMADIO 100G Gen2 1U Packet System Side

## Spec Sheet



100G Gen2 Datasheet v5.pdf 1MB  
PDF

FMADIO100G Gen2 Spec Sheet



10G 40G 100G PACKET CAPTURE

## FMADIO100G Gen2

Full Line Rate Sustained 100Gbit Packet Capture

## 100Gbps Full Packet Capture

Advanced 100Gbps Line Rate Packet Capture system. Full packet capture at sustained 100Gbps Line Rates with 64Byte packets at 148.8Mpps.

FMADIO Advanced Technology group has created the best 100Gbps packet capture system utilizing proprietary FPGA hardware and bleeding edge software, at a clear and affordable price.

All FMADIO appliances come with a fast, easy to use UI, that's simple to learn. Our PacketScope software gives you the power to quickly dissect data to find the packets you need to solve problems. Familiar BPF filtering reliably reduces the size of packet downloads.

A single device supports 2x100G, 2x40G, 8x25G, 8x10G operating modes, for all network topologies.

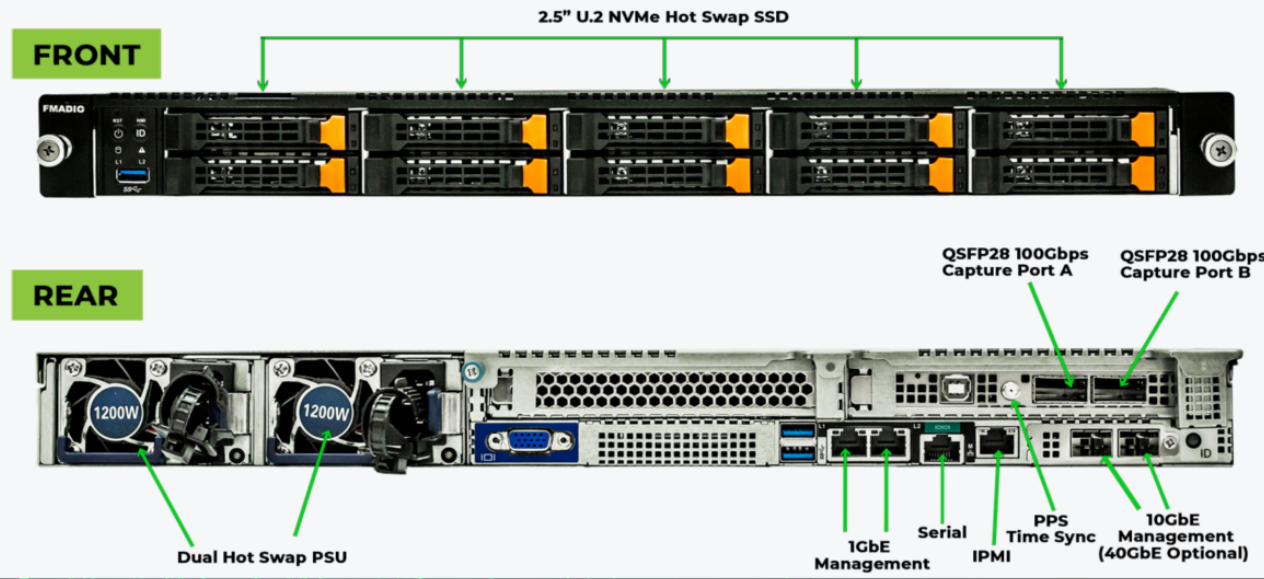
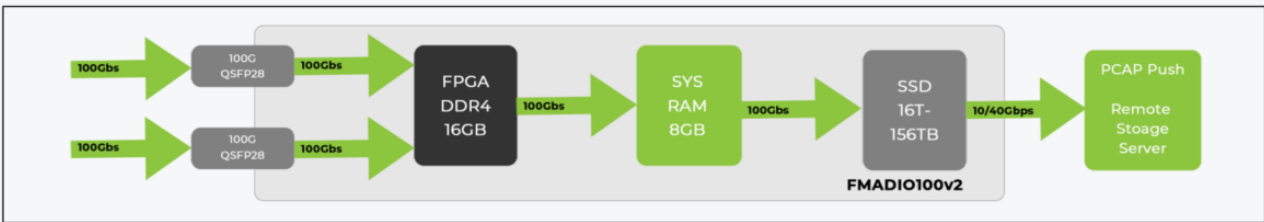
In addition the system has a built in 100Gbps packet generator and replay feature, enabling capture and replay using the same device.

### FEATURES

- 100Gbps Sustained Packet Capture
- 200Gbps Burst Packet Capture
- 2x QSFP28 Ports
- SR4 / LR4 / CX4 + FEC support
- 200Gbps Line Rate Packet Generator
- 16TB ~ 156TB RAID5 Storage
- Hardware Time Stamping
- Pre-capture Filtering
- Inline Packet Capture
- Packet Slicing
- Packet Replay + Blaster
- Simple easy to use Web GUI
- Fully scriptable JSON API

### SPECIFICATION

<b>Network Sustained Capture Speed:</b> 100Gbps Line Rate Packet Capture 148.8Mpps Packet Capture	<b>Hardware Time Stamp Resolution:</b> 3.2ns	<b>Rackmount Rails:</b> 31" Long Tool-less Rails
<b>Network Capture Ports:</b> 2x QSFP28 100Gbps Capture Port 2x100G / 2x40G / 8x25G / 8x10G SR4 / CDWM4 / LR4 / CX4 + FEC Support	<b>Clock Synchronization:</b> PTPv2, NTP, PPS	<b>Cooling:</b> 12x Hot swap 40mm Fans
<b>Packet Size:</b> 64B-9218B	<b>Hot Swap Drive:</b> x10 U.2 NVMe SSD drives	<b>Power:</b> Dual 1200W Power Supply Hot Swap Power Supply (Operates with Single PSU)
<b>Management Interfaces:</b> 2x 10GbE SFP+ 2x 10/100/1000Mbe RJ45 2x 40GbE QSFP (optional)	<b>Storage Size:</b> 16TB - 156TB Total Raw Storage	<b>Size:</b> 1U Rack mountable 482mmx787mmx44mm 19" x 31" x 1.75"
	<b>Storage Mode:</b> Full 100Gbps RAID5	<b>Total Weight:</b> 24.8kg / 55lbs



fmad engineering llc  
 4023 Kennett Pike  
 Wilmington, DE 50194  
 sales@fmad.io

fmad engineering llc  
 東京都渋谷区1-8-3渋谷  
 TOC 第1ビル8F  
 support@fmad.io

## Transceiver Support

List of known and tested transceivers that work with the system. Other transceiver types should be ok, please check with support@fmad.io for clarification.

FMADIO Packet Capture systems are **Transceiver Vendor neutral**, 3rd party optics are fully supported.

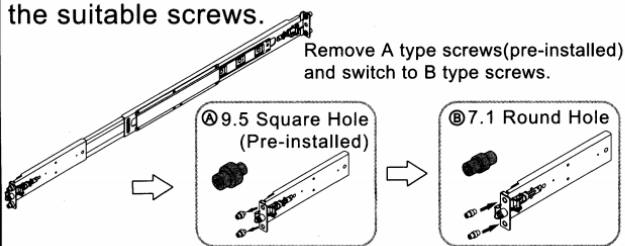
Module	Speed	Connector	Cable Type	Brand
100G SR4	100Gbps	MPO12	OM4 MMF	Generic
100G LR4	100Gbps	LC	OS2 SMF	Generic Finisar FTLC1154RD
100G CWDM4	100Gbps	LC	OS2 SMF	Generic Finisar CTLC1157RC
100G SWDM4	100Gbps	LC	OM4 MMF	Generic Finisar FTLC9152RG
100G CR4	100Gbps	DAC	Passive Twinax	Generic
40G SR4	40Gbps	MPO12	OM4 MMF	Generic Finisar FTL410QE4C
40G LR4	40Gbps	LC	OS2 SMF	Generic
40G CR4	40Gbps	DAC	Passive Twinax	Generic

## Rails installation

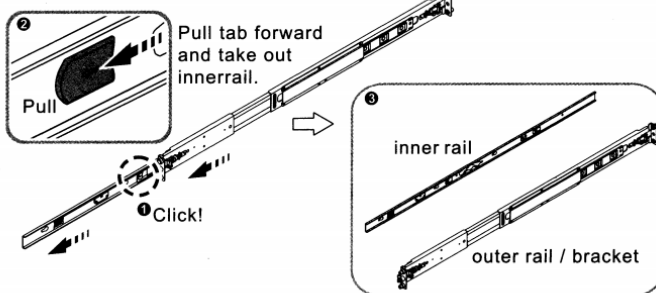
Installation of the rails is tool less please see the following instructions



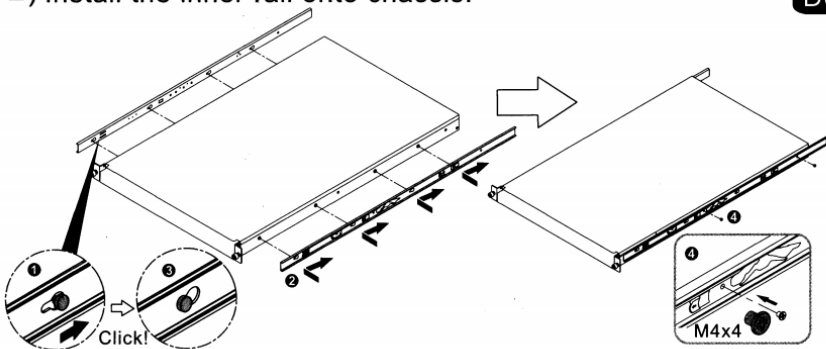
**Note:** Please check rack post type. The rail pre-installed type is for square hole post. If rack post is round hole post, please switch to the suitable screws.



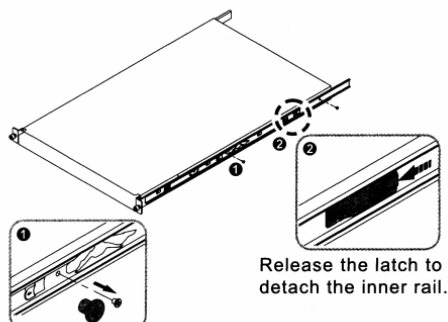
### 1) Remove the inner rail.



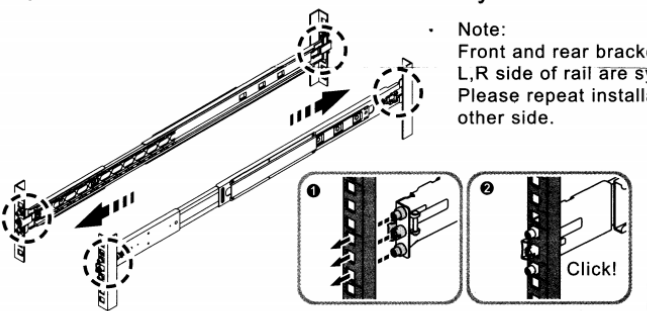
### 2) Install the inner rail onto chassis.



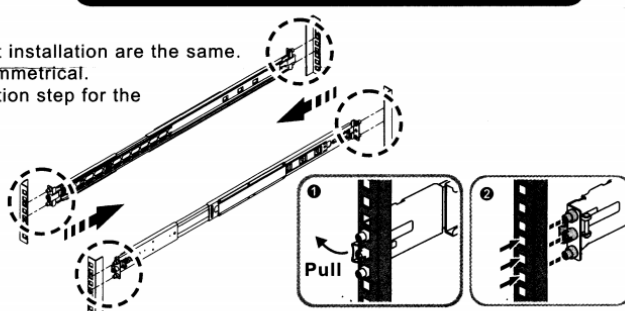
### Detach Front Bracket from Rack Post



### 3) Fix the outer rail/bracket assembly to the frame.



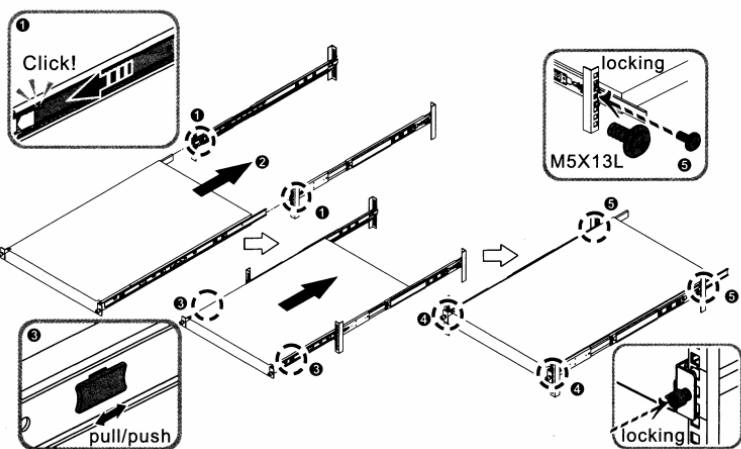
### Detach Front Bracket from Rack Post



**Note:**  
Front and rear bracket installation are the same.  
L,R side of rail are symmetrical.  
Please repeat installation step for the other side.

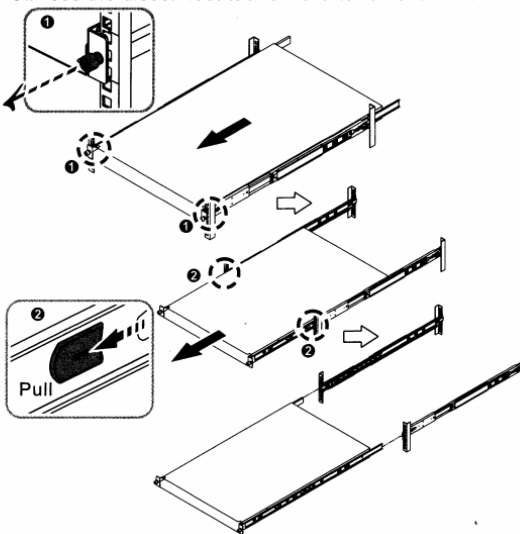
### 4) Insert chassis to complete the installation.

1. Ensure ball bearing retainer is located at the front of the rail.
2. Insert the chassis into outer rails.
3. When hit a stop, please pull / push the release tab on the inner rails.
4. Tighten chassis with shipping screws.



### Detach Front Bracket from Rack Post

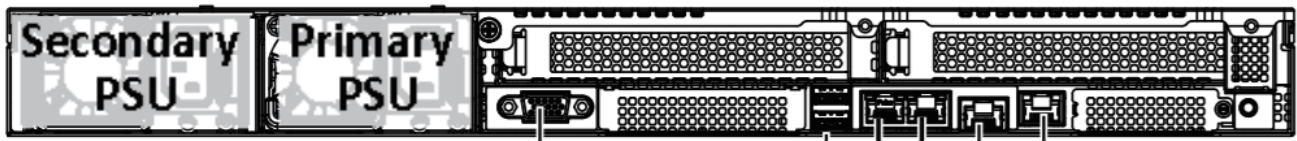
1. Loosen shipping screw to pull out chassis.
2. Press the disconnect tab forward to remove chassis.



## Power Supply

System is installed with dual 1200W power supply. By default the system is configured as follows

Total Power Consumption	Behavior	LED
< 400W	HOT-STANDBY Only Primary PSU is active, Secondary PSU in standby mode	Primary, Solid Green Secondary, Green Blink
> 400W	HOT-HOT Both Primary and Secondary PSU are in active mode	Both Solid Green

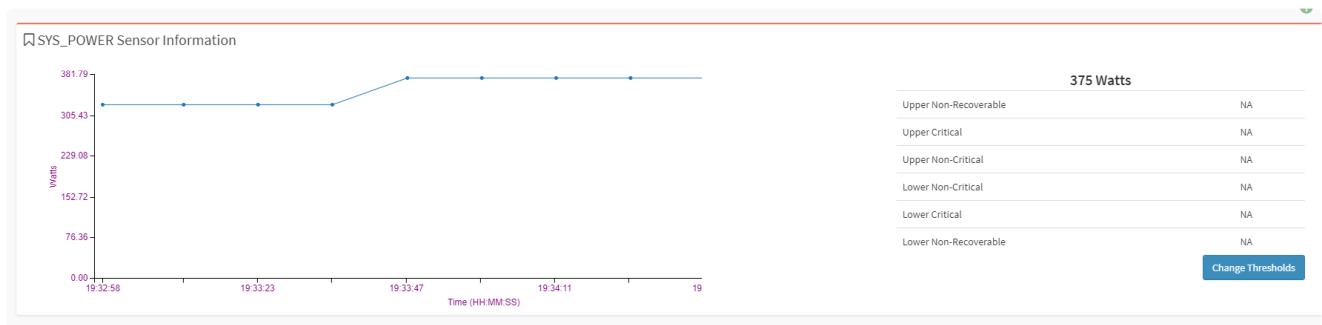


FMADIO 100G PSU Configuration

The above behavior can be overridden to always be in HOT-HOT load balancing mode, with a configuration setting. Please contact support@fmad.io for more information

## Power Consumption

### FMADIO 100G 1U Analytics



## Power Consumption Reading

Current Power Consumption (W)

340

Minimal Power Consumption (W)

192

Maxmal Power Consumption (W)

426

Average Power Consumption (W)

325

## Specification

### Physical Dimensions:

- 438 x 43.5 x 730mm
- 17.2" x 1.7" x 28.7"
- (Designed for EIA 19" rack)

### Weight:

- 26KG / 57 lbs

### Color:

- Silver metallic

### Power Supply:

- 2 x Redundant Hot swappable 1200W
- 110-240AC, 12-7A, 50-60Hz

### Fan:

- 8 x 40x40x56mm (23,000rpm) Chassis
- 2 x 1 x 40x40x56mm (Power Supply)

### Airflow:

- Front to back

**Operating Conditions:**

- Operating temperature: 10°C to 28°C
- Operating humidity: 8-80% (non-condensing)

**Storage:**

- 10 x U.2 NVMe Front panel drives

**Network:**

- 1 x IPMI BMC out of band management (RJ45)
- 2 x RJ45 Management ports
- 2 x SFP+(or QSFP+) High speed management ports
- 2 x QSFP28 Capture ports
- 1 x Serial port (RJ45 connector)

**Platform:**

- Dual socket Intel Xeon Platform
- DDR4 RDIMM RAM

# Performance

The performance of a capture system can be characterized in a number of different ways. We provide the following performance dimensions

## **Microburst Capture Speed ( < 500msec )**

This is the short time burst capture rate of the system. For the 100G Gen2 system this is burst capture rate that fills up the DDR buffer on the FMAD FPGA Capture card.

## **Burst Capture Speed ( < 10min)**

For the 100G Gen2 FMADIO Packet Capture system, all storage is on high speed NVMe SSDs, so the Burst Capture Speed is the same as the Sustained Capture Speed.

FMADIO 20G and 40G systems use a mixture of SSD and magnetic disk storage, so for those systems the Burst Capture Speed is higher than the Sustained Capture Speed.

## **Sustained Capture Speed (24/7)**

We indicate this as the sustained capture rate, i.e. the capture rate that a system can sustain 24/7 without any packet loss. For systems with Magnetic storage, mixing capture with downloads effects the writeback to magnetic storage speed, this performance metric is *Capture Only* with no simultaneous/concurrent downloads.

## **Capture and Download Speed**

Performance metric is assuming no bottlenecks on the egress (download client) what is the capture performance while simultaneously downloading.

## **Download Only speed**

The other metric is Download only speed. This metric is used to calculate the maximum rate data can be moved off the device over 10G or 40G ethernet.

# Getting Started

## Connecting

There are 4 ways to connect to the appliance for initial configuration:

- KVM – The appliance has a VGA port for a monitor and USB for keyboard.
- Serial – The serial port can be used with a configuration of 115200, 8, None, 1.
- SSH to the OS – There are 2 management ports preconfigured with IP addresses:

```
man0 (1GbE copper): 192.168.0.95 / 24  
man10 (10GbE SFP+): 192.168.15.10 / 24
```

- IPMITOOL to the IPMI/BMC/iLO/iDARC – The IPMI port is preconfigured with IP address:

```
bmc (1GbE copper): 192.168.0.93 / 24
```

## Show Network Status

To show the current network status use the `fmadiocli` utility, by typing it on the shell after logging in.

```
fmadiocli
```

Example



```
[Fri Jun 9 19:12:19 2023] > show interface status
[Fri Jun 9 19:12:19 2023] Port          Description Status          Speed          Transceiver
[Fri Jun 9 19:12:19 2023] -----
[Fri Jun 9 19:12:19 2023] man0                connected      1G
[Fri Jun 9 19:12:19 2023] man10               connected      10G            10G Base-SR 0
[Fri Jun 9 19:12:19 2023] cap0                notconnected   10G
[Fri Jun 9 19:12:19 2023] cap1                connected      10G            10G SR
[Fri Jun 9 19:12:19 2023] >
```

NOTE: system has no way to check the BMC link up/down status.

## Configure New Network IP

Once logged in, the first task is to set new IP addresses. Type `fmadiocli` to enter the configuration CLI, then use the following commands

To set the IPv4 Address

```
config interface ip <interface name> <IPv4 address>
Example: config interface ip man0 192.168.187.10
```

To set the IPv4 netmask

```
config interface netmask <interface name> <subnet mask>
Example: config interface netmask man0 255.255.255.0
```

To set the IPv4 Gateway

```
config interface gateway <interface> <IPv4 gateway address>
Example: config interface gateway man0 192.168.187.30
```

To set the IPv4 DNS (optional)

```
config interface <interface> <IPv4 DNS address>
Example: config interface dns man0 1.1.1.1
```

Full documentation is at



fmadiocli  
fmadio documentation

Example setup process is shown below



## Using IPMITOOL serial port to login

```
# ipmitool -U admin -P ***** -H 192.168.0.93 -I lanplus sol activate
[SOL Session operational. Use ~? for help]

fmadio20v3
fmadio20v3-287 login: fmadio
Password:
          .----.---      20Gv3
_/_  _/_\  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_
\  _/_\  /  \_/_  \  /  _/_  | |  | |  /  _/_  \
|  | |  Y Y  \ /  _/_  \_/_ /_/_ / | |  | ( <_> )
|_/_| |_/_| | /(_/_/_  /\_/_/_  | |_/_| \_/_/_/
          \/_  \/_  \/_

=====
  +- all your packet are belong to us +-
fmadio@fmadio20v3-287:~$ fmadiocli
fmad fmadlua Jun  7 2023 (/opt/fmadio/bin/fmadiolua --nocal /opt/fmadio/bin/fmadiocli )
Disable cycle calibration
[Fri Jun  9 19:17:50 2023]  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_
[Fri Jun  9 19:17:50 2023] _/_  _/_\  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_  _/_/_
[Fri Jun  9 19:17:50 2023] \  _/_\  /  \_/_  \  /  _/_  | |  | |  /  _/_  \
[Fri Jun  9 19:17:50 2023] |  | |  Y Y  \ /  _/_  \_/_ /_/_ / | |  | ( <_> )
[Fri Jun  9 19:17:50 2023] |_/_| |_/_| | /(_/_/_  /\_/_/_  | |_/_| \_/_/_/
[Fri Jun  9 19:17:50 2023]          \/_  \/_  \/_
[Fri Jun  9 19:17:50 2023] =====
[Fri Jun  9 19:17:50 2023]   +- Packets confiscated by Customs +-
[Fri Jun  9 19:17:50 2023]
[Fri Jun  9 19:17:50 2023] type '?' for command information
[Fri Jun  9 19:17:50 2023] type '???' for verbose information
[Fri Jun  9 19:17:50 2023]
[Fri Jun  9 19:17:50 2023] History: 32
[Fri Jun  9 19:17:53 2023] >
```

## Configure the IPv4 man0 interface

```

[Fri Jun 9 19:18:13 2023] > config interface ip man0 192.168.187.10
[Fri Jun 9 19:18:15 2023] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocal
[Fri Jun 9 19:18:15 2023] UPDATING: fmad fmadlua Jun 7 2023 (/usr/local/bin/fmadiolua --no
[Fri Jun 9 19:18:15 2023] UPDATING: Disable cycle calibration
[Fri Jun 9 19:18:15 2023] UPDATING: network config [Fri Jun 9 19:18:15 2023] fmadio20v3
.
.
.
[Fri Jun 9 19:18:18 2023] UPDATING: Network Setup... Done
[Fri Jun 9 19:18:18 2023] UPDATING: done 3.184404Sec 0.053073Min
[Fri Jun 9 19:18:18 2023] UPDATING: cat /opt/fmadio/etc/hosts >> /etc/hosts
[Fri Jun 9 19:18:18 2023] set interface [man0] ip (192.168.187.10) -> (192.168.187.10)
[Fri Jun 9 19:18:18 2023]
[Fri Jun 9 19:18:18 2023] Port          Mode          IP             Netmask        Gateway
[Fri Jun 9 19:18:18 2023] -----
[Fri Jun 9 19:18:20 2023] bmc          static        192.168.187.2  255.255.255.0  192.168.187
[Fri Jun 9 19:18:20 2023] man0         static        192.168.187.10 255.255.255.0  192.168.187
[Fri Jun 9 19:18:20 2023] man1         disabled      192.168.1.2    255.255.255.0  192.168.1.1
[Fri Jun 9 19:18:20 2023] man10        static        192.168.91.50  255.255.255.0
[Fri Jun 9 19:18:20 2023] cap0
[Fri Jun 9 19:18:20 2023] cap1
[Fri Jun 9 19:18:20 2023] -----
[Fri Jun 9 19:18:39 2023] >

```

Configure the IPv4 man0 netmask

```

[Fri Jun 9 19:18:39 2023] > config interface netmask man0 255.255.255.0
[Fri Jun 9 19:18:40 2023] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocall
[Fri Jun 9 19:18:40 2023] UPDATING: fmad fmadlua Jun 7 2023 (/usr/local/bin/fmadiolua --no
[Fri Jun 9 19:18:40 2023] UPDATING: Disable cycle calibration
[Fri Jun 9 19:18:40 2023] UPDATING: network config [Fri Jun 9 19:18:40 2023] fmadio20v3
[Fri Jun 9 19:18:40 2023] UPDATING: load intel 10G driver
[Fri Jun 9 19:18:41 2023] UPDATING: load intel 1G driver
[Fri Jun 9 19:18:43 2023] UPDATING: [0/10] phy0:up phy1:up man10:up true
.
.
.
[Fri Jun 9 19:18:43 2023] UPDATING: Using Custom hosts
[Fri Jun 9 19:18:43 2023] UPDATING: Network Setup... Done
[Fri Jun 9 19:18:43 2023] UPDATING: done 3.177373Sec 0.052956Min
[Fri Jun 9 19:18:43 2023] UPDATING: cat /opt/fmadio/etc/hosts >> /etc/hosts
[Fri Jun 9 19:18:43 2023] set interface [man0] netmask (255.255.255.0) -> (255.255.255.0)
[Fri Jun 9 19:18:43 2023]
[Fri Jun 9 19:18:43 2023] Port          Mode          IP             Netmask       Gateway
[Fri Jun 9 19:18:44 2023] -----
[Fri Jun 9 19:18:46 2023] bmc          static        192.168.187.2  255.255.255.0 192.168.187
[Fri Jun 9 19:18:46 2023] man0         static        192.168.187.10 255.255.255.0 192.168.187
[Fri Jun 9 19:18:46 2023] man1         disabled      192.168.1.2    255.255.255.0 192.168.1.1
[Fri Jun 9 19:18:46 2023] man10        static        192.168.91.50  255.255.255.0
[Fri Jun 9 19:18:46 2023] cap0
[Fri Jun 9 19:18:46 2023] cap1
[Fri Jun 9 19:18:46 2023] -----
[Fri Jun 9 19:19:19 2023] >

```

Configure IPv4 man0 gateway

```

[Fri Jun 9 19:19:32 2023] > config interface gateway man0 192.168.187.30
[Fri Jun 9 19:19:32 2023] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocal
[Fri Jun 9 19:19:32 2023] UPDATING: fmad fmadlua Jun 7 2023 (/usr/local/bin/fmadiolua --no
[Fri Jun 9 19:19:32 2023] UPDATING: Disable cycle calibration
[Fri Jun 9 19:19:32 2023] UPDATING: network config [Fri Jun 9 19:19:32 2023] fmadio20v3
[Fri Jun 9 19:19:32 2023] UPDATING: load intel 10G driver
[Fri Jun 9 19:19:34 2023] UPDATING: load intel 1G driver
[Fri Jun 9 19:19:35 2023] UPDATING: [0/10] phy0:up phy1:up man10:up true
.
.
.
[Fri Jun 9 19:19:36 2023] UPDATING: Using Custom hosts
[Fri Jun 9 19:19:36 2023] UPDATING: Network Setup... Done
[Fri Jun 9 19:19:36 2023] UPDATING: done 3.457898Sec 0.057632Min
[Fri Jun 9 19:19:36 2023] UPDATING: cat /opt/fmadio/etc/hosts >> /etc/hosts
[Fri Jun 9 19:19:36 2023] set interface [man0] Gateway (192.168.187.30) -> (192.168.187.30)
[Fri Jun 9 19:19:36 2023]
[Fri Jun 9 19:19:36 2023] Port          Mode          IP             Netmask       Gateway
[Fri Jun 9 19:19:36 2023] -----
[Fri Jun 9 19:19:38 2023] bmc          static        192.168.187.2  255.255.255.0 192.168.187
[Fri Jun 9 19:19:38 2023] man0         static        192.168.187.10 255.255.255.0 192.168.187
[Fri Jun 9 19:19:38 2023] man1         disabled      192.168.1.2    255.255.255.0 192.168.1.1
[Fri Jun 9 19:19:38 2023] man10        static        192.168.91.50  255.255.255.0
[Fri Jun 9 19:19:38 2023] cap0
[Fri Jun 9 19:19:38 2023] cap1
[Fri Jun 9 19:19:38 2023] -----
[Fri Jun 9 19:19:53 2023] >

```

Configure the IPv4 man0 DNS (optional)

```

[Fri Jun 9 19:19:53 2023] > config interface dns man0 1.1.1.1
[Fri Jun 9 19:19:53 2023] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocal
[Fri Jun 9 19:19:53 2023] UPDATING: fmad fmadlua Jun 7 2023 (/usr/local/bin/fmadiolua --no
[Fri Jun 9 19:19:53 2023] UPDATING: Disable cycle calibration
[Fri Jun 9 19:19:53 2023] UPDATING: network config [Fri Jun 9 19:19:53 2023] fmadio20v3
[Fri Jun 9 19:19:53 2023] UPDATING: load intel 10G driver
[Fri Jun 9 19:19:55 2023] UPDATING: load intel 1G driver
[Fri Jun 9 19:19:56 2023] UPDATING: [0/10] phy0:up phy1:up man10:up true
[Fri Jun 9 19:19:56 2023] UPDATING: device man0 already exists; can't create bridge with th
[Fri Jun 9 19:19:56 2023] UPDATING: device man1 already exists; can't create bridge with th
.
.
.
[Fri Jun 9 19:19:57 2023] UPDATING: echo "nameserver 1.1.1.1" >> /etc/resolv.conf
[Fri Jun 9 19:19:57 2023] UPDATING: Using Custom hosts
[Fri Jun 9 19:19:57 2023] UPDATING: Network Setup... Done
[Fri Jun 9 19:19:57 2023] UPDATING: done 3.181773Sec 0.053030Min
[Fri Jun 9 19:19:57 2023] UPDATING: cat /opt/fmadio/etc/hosts >> /etc/hosts
[Fri Jun 9 19:19:57 2023] set interface [man0] DNS (nil) -> (1.1.1.1)
[Fri Jun 9 19:19:57 2023]
[Fri Jun 9 19:19:57 2023] Port           Mode           IP              Netmask         Gateway
[Fri Jun 9 19:19:57 2023] -----
[Fri Jun 9 19:19:58 2023] bmc           static         192.168.187.2   255.255.255.0   192.168.187
[Fri Jun 9 19:19:58 2023] man0          static         192.168.187.10  255.255.255.0   192.168.187
[Fri Jun 9 19:19:58 2023] man1          disabled      192.168.1.2     255.255.255.0   192.168.1.1
[Fri Jun 9 19:19:58 2023] man10         static         192.168.91.50   255.255.255.0
[Fri Jun 9 19:19:58 2023] cap0
[Fri Jun 9 19:19:58 2023] cap1
[Fri Jun 9 19:19:58 2023] -----
[Fri Jun 9 19:19:58 2023] >

```

# Errata

Errata related to FMADIO100v2 Capture System

## Capture Port 0

Due to design decision of the FPGA clocking QSFP0 Capture port must be connected first or put in shutdown mode using the `fmadiocli`.

Recommended configuration is always connect QSFP0 port first

### Root cause

The reason is if the frontend of the capture FPGA uses the clock 322mhz clock that runs off the QSFP0 port FPGA GTY transceiver. If the port is left un-connected and not shutdown the system will try automatically link up in different modes. The constant attempted link up, transitioning between modes (eg FEC and non-FEC) requires resetting of the clock resulting in unstable behavior on Capture Port 1

# Configuration

# Network Configuration (GUI)

Network port configuration can be achieved using a) the web interface, b) SSH command line interface (CLI). Using the Web interface is the easiest route, however in highly constrained network environments a pure CLI based configuration can be easier

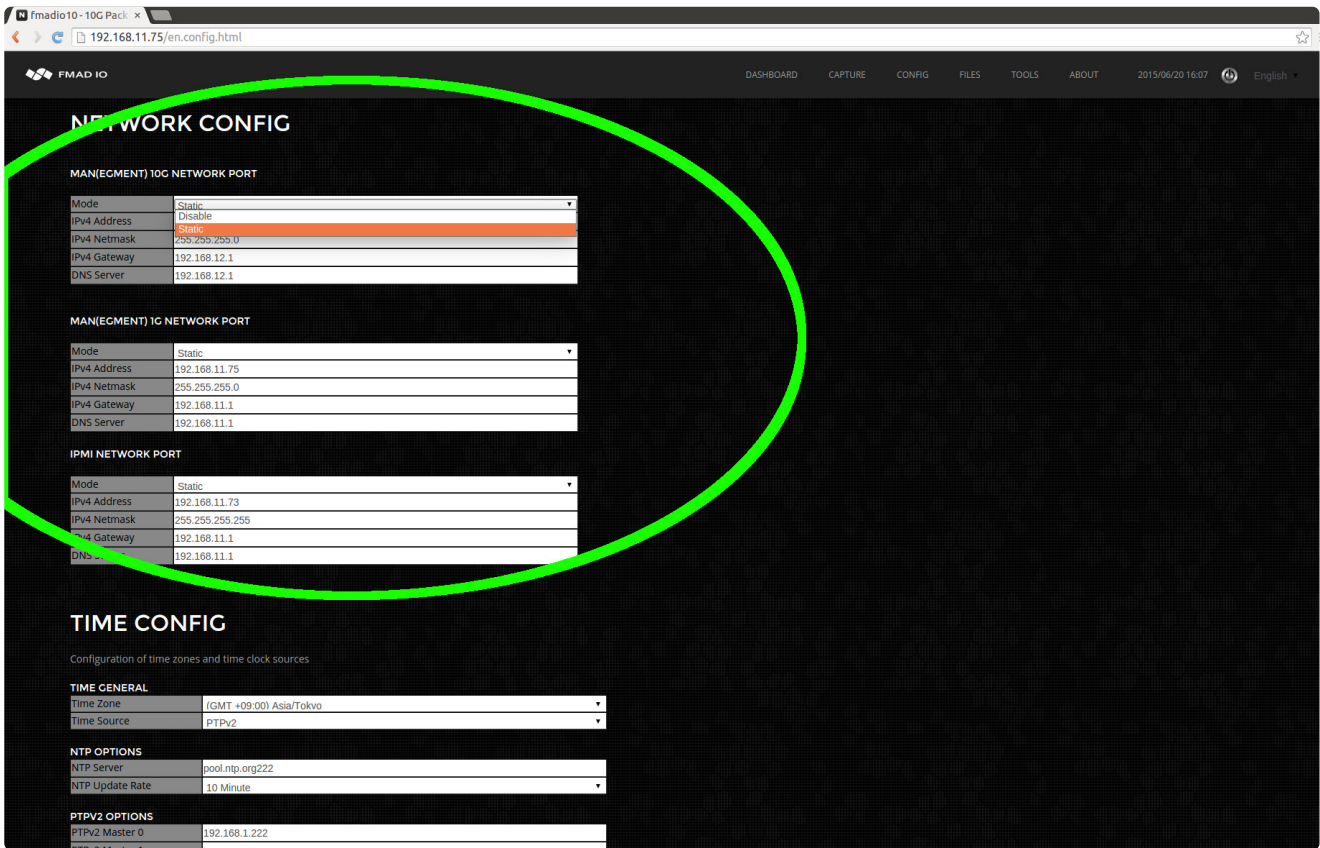
## WEB INTERFACE: NETWORK CONFIG

From the dashboard page, Start by selecting the configuration menu option from as shown below (highlighted in green).

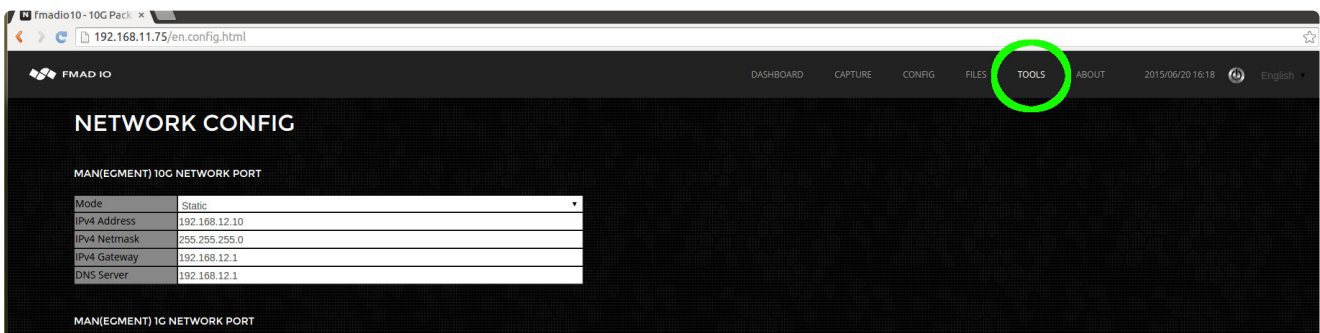


Then edit the network configuration's IP/Netmask/Gateway/DNS setting as shown in the image below. After each field has been edited the system automatically saves and updates the system setting (save button is not required). After completing the update, refresh the web page to confirm the new settings.





Select the tools menu from the top toolbar, as shown in the image below.



And finally select the Power Cycle / Reboot button to restart the system

## POWER CYCLE SYSTEM

Initiate an orderly system reboot, takes approximately 1 minute.



## CHANGE PASSWORD

Change system password

Enter New Password	<input type="password"/>	 Change
Enter New Password (again)	<input type="password"/>	

## FIRMWARE UPDATE



Select or upload new system firmware

File Name	MD5	Date Uploaded	Version	Activate	Select
fmadio10_20150619_2126.bin	9e1642aff11079087922e3b690377724	21:28:22 - Jun 19 2015	fmadio10 svn:1150 Build Fri Jun 19 21:26:28 2015	true	<input checked="" type="checkbox"/>
fmadio10_20150619_2344.bin	02a72ef10bc9241d0080f7acc78e6872f	23:47:28 - Jun 19 2015	fmadio10 svn:1166 Build Thu Jun 18 23:44:45 2015		<input type="checkbox"/>

# Network Configuration (CLI)

Modifying the network configuration setting in a restricted Colocation environment can be far easier to achieve via the command line. The first step is SSH into the system, change to the specified directory and view the current network settings, as shown below.

Specifically:

interface name	Hardware Port	Description
man0	L1 1G RJ45	1G management port
bmc	IPMI 1G RJ45	1G Out of bands managemer port
man10	SFP/QSFP 10G/40G	High speed management poi

## Default Settings

Interface Name	IP Address	Netmask	Gateway	DNS
man0	192.168.0.95	255.255.255.0	192.168.0.1	192.168.0.1
bmc	192.168.0.93	255.255.255.0	192.168.0.1	
man10	192.168.15.95	255.255.255.0		

Example configuration is shown below

```

fmadio@fmadio10-049:/mnt/store0/etc$ cat network.lua
local Config =
{
  ["man0"] =
  {
    ["Mode"]      = "static",
    ["Address"]   = "192.168.0.95",
    ["Netmask"]   = "255.255.255.0",
    ["Gateway"]   = "192.168.0.1",
    ["DNS"]       = "192.168.0.1",
  },
  ["man10"] =
  {
    ["Mode"]      = "static",
    ["Address"]   = "192.168.15.95",
    ["Netmask"]   = "255.255.255.0",

    ["Gateway"]   = "",
    ["DNS"]       = "",
  },
  ["bmc"] =
  {
    ["Mode"]      = "static",
    ["Address"]   = "192.168.0.93",
    ["Netmask"]   = "255.255.255.0",
    ["Gateway"]   = "",
    ["DNS"]       = "",
  },
}
return Config

```

After editing the file, save it and confirm no syntax errors, running the following command

```
fmadiolua /opt/fmadio/etc/network.lua
```

A successful output looks like the following

```

fmadio@fmadio20n40v3-363:/opt/fmadio/bin$ fmadiolua /opt/fmadio/etc/network.lua
loading filename [/opt/fmadio/etc/network.lua]
done 0.000052Sec 0.000001Min
fmadio@fmadio20n40v3-363:/opt/fmadio/bin$

```

Reboot the system and confirm the IP settings are correct

## Management VLAN

There may be a requirement for management VLAN interfaces on the FMADIO packet capture system. These are simple to add using the standard linux infrastructure

Example is to create a VLAN 123 on the 10G man10 interface. Add the following configuration to

```
/opt/fmadio/etc/network.lua
```

Create a new entry names "man10.123" as follows

```
["man10.123"] =  
{  
  ["Mode"]      = "static",  
  ["vlan"]      = "123",  
  ["Address"]   = "192.168.123.2",  
  ["Netmask"]   = "255.255.255.0",  
  ["Gateway"]   = "",  
  ["DNS0"]      = "",  
  ["DNS1"]      = "",  
  ["Speed"]     = "10g",  
  ["TSMode"]    = "nic",  
},
```

The only difference is the addition of the "vlan" setting and an entry named "man10.123" all other fields are configured as a normal static interface.

After configuring be sure to run the following to confirm the configuration file is formatted correctly. The following is the correct output

```
fmadio@fmadio20n40v3-364:~$ fmadlua /opt/fmadio/etc/network.lua  
fmad fmadlua Aug 23 2021  
failed to open self? [fmadlua]  
loading filename [/opt/fmadio/etc/network.lua]  
done 0.000047Sec 0.000001Min  
fmadio@fmadio20n40v3-364:~$
```

## Changing BMC/IPMI Network Settings

In addition to the above, if the BMC information has been changed, please run the following command to configure BMC IP address on the system.

```
fmadio@fmadio20n40v3-363:/opt/fmadio/bin$ sudo ./setup_network.lua --updatebmc
```

Example output of BMC IP configuration is shown below

```

fmadio@fmadio20n40v3-363:/opt/fmadio/bin$ sudo ./setup_network.lua --updatebmc
loading filename [./setup_network.lua]
network config [Mon Jul  5 02:19:44 2021] fmadio40v3
load intel 10G driver

rmmod: can't unload 'ixgbe': Device or resource busy
insmod: can't insert '/opt/fmadio/drivers/current/ixgbe.ko': File exists
load intel 1G driver
rmmod: can't unload 'igb': Device or resource busy
insmod: can't insert '/opt/fmadio/drivers/current/igb.ko': File exists
[0/10] phy0:up phy1:up man10:up  true
/bin/brctl addbr man0
device man0 already exists; can't create bridge with the same name
/bin/brctl addif man0 phy0
device phy0 is already a member of a bridge; can't enslave it to bridge man0.
/bin/brctl addbr man1
device man1 already exists; can't create bridge with the same name
/bin/brctl addif man1 phy1
device phy1 is already a member of a bridge; can't enslave it to bridge man1.
/bin/brctl addbr man10
device man10 already exists; can't create bridge with the same name
/bin/brctl addif man10 phy10
interface phy10 does not exist!
/sbin/ifconfig phy0 up
/sbin/ifconfig phy1 up
/sbin/ifconfig phy10 up
.
.
.
.
BMC Setting: Addr:192.168.2.223 Netmask:255.255.255.0 GW:192.168.2.254
/sbin/ifconfig man0 192.168.2.225 netmask 255.255.255.0
/sbin/ifconfig man10 192.168.15.225 netmask 255.255.255.0
/sbin/ifconfig man10 mtu 9200
/sbin/route add -net default gw 192.168.2.254
route: SIOCADDRT: File exists
echo "" > /etc/resolv.conf
echo "nameserver 192.168.2.254" >> /etc/resolv.conf
/usr/sbin/ipmitool lan set 1 ipaddr 192.168.2.223
Setting LAN IP Address to 192.168.2.223
/usr/sbin/ipmitool lan set 1 netmask 255.255.255.0
Setting LAN Subnet Mask to 255.255.255.0
/usr/sbin/ipmitool lan set 1 defgw ipaddr 192.168.2.254
Setting LAN Default Gateway IP to 192.168.2.254
Network Setup... Done
done 4.552995Sec 0.075883Min
fmadio@fmadio20n40v3-363:/opt/fmadio/bin$

```

Once Complete confirming the BMC IP network settings using the following command

```
sudo ipmitool lan print
```

Example output looks like the following

```
fmadio@fmadio20n40v3-363:/opt/fmadio/bin$ sudo ipmitool lan print
Set in Progress           : Set Complete
Auth Type Support        : NONE MD2 MD5 PASSWORD OEM
Auth Type Enable         : Callback : MD5
                          : User      : MD5
                          : Operator : MD5
                          : Admin   : MD5
                          : OEM     : MD5
IP Address Source        : Static Address
IP Address                : 192.168.0.93
Subnet Mask              : 255.255.255.0
MAC Address              : 18:c0:4d:b4:0e:72
SNMP Community String    : AMI
IP Header                 : TTL=0x40 Flags=0x40 Precedence=0x00 TOS=0x10
BMC ARP Control          : ARP Responses Enabled, Gratuitous ARP Disabled
Gratuitous ARP Intrvl    : 1.0 seconds
Default Gateway IP       : 192.168.0.1
Default Gateway MAC      : 10:da:43:c8:03:ac
Backup Gateway IP        : 0.0.0.0
Backup Gateway MAC       : 00:00:00:00:00:00
802.1q VLAN ID          : Disabled
802.1q VLAN Priority     : 0
RMCP+ Cipher Suites     : 0,1,2,3,6,7,8,11,12,15,16,17
Cipher Suite Priv Max    : caaaaaaaaaaXXX
                          : X=Cipher Suite Unused
                          : c=CALLBACK
                          : u=USER
                          : o=OPERATOR
                          : a=ADMIN
                          : O=OEM
Bad Password Threshold   : 0
Invalid password disable: no
Attempt Count Reset Int.: 0
User Lockout Interval   : 0
fmadio@fmadio20n40v3-363:/opt/fmadio/bin$
```

After confirming the IP address, BMC IP address is pingable also reachable with IPMITOOL, HTTP and HTTPS.

# Capture Port Link Speed

FMADIO Capture systems capture at multiple different link speeds based on the Device Model number selected, we offer all port speeds at no additional charge. The following port configurations are supported

## FMADIO100v2:

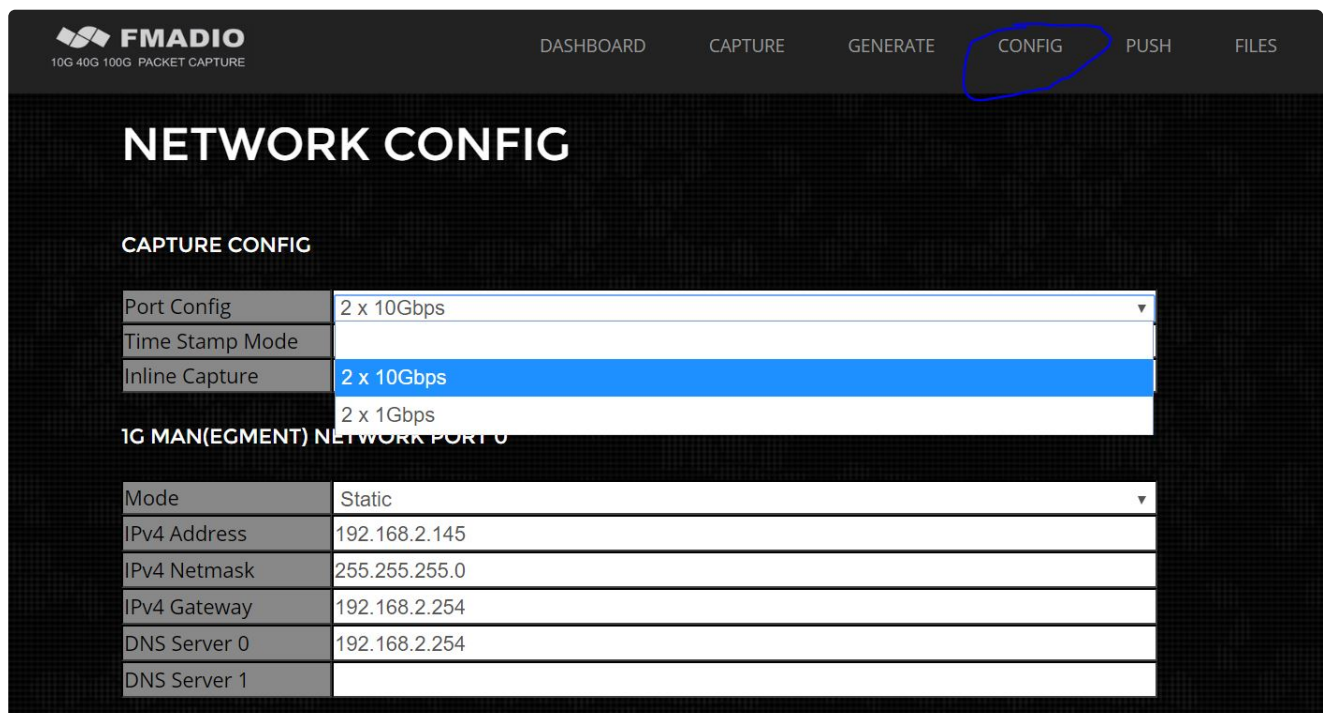
- 2x100G
- 2x40G
- 4x25G (in progress)
- 8x10G

## CONFIG

Configuring the different port speeds requires updating the FPGA NIC, which requires setting the Capture Port mode and then re-updating the devices firmware. The steps are shown below:

### Step 1)

Select the port configuration "Config Page - > Port Config" as shown below. In this example 2x10G mode is selected.



The screenshot shows the FMADIO NETWORK CONFIG interface. The top navigation bar includes DASHBOARD, CAPTURE, GENERATE, CONFIG (highlighted with a blue circle), PUSH, and FILES. The main content area is titled NETWORK CONFIG and contains a CAPTURE CONFIG section. This section has a table with the following fields:

Port Config	2 x 10Gbps
Time Stamp Mode	
Inline Capture	2 x 10Gbps
	2 x 1Gbps

Below this is the 1G MAN(EGMENT) NETWORK PORT 0 section, which has a table with the following fields:

Mode	Static
IPv4 Address	192.168.2.145
IPv4 Netmask	255.255.255.0
IPv4 Gateway	192.168.2.254
DNS Server 0	192.168.2.254
DNS Server 1	

### Step 2)

After the port configuration has been chosen, click the "CHANGE" button to change the port speed. This will reboot the system twice as its reconfiguring the FPGA device. It will take 3-5 minutes to complete the operation



# NETWORK CONFIG

## CAPTURE CONFIG

Port Config	2 x 100Gbps	Change
Time Stamp Mode	Capture Card	
Inline Capture	Disable	

### Step 3)

Once the update has completed, please verify the capture port configuration on the GUI dashboard, as shown below in blue.

# NETWORK PORT STATUS

Name	Port	Speed	Link Status	Link Uptime	Link type	Wavelength
1G Management	eth0	1Gb	UP	0D 23H 48M		
10G Management	man0	10Gb	UP	0D 23H 48M		
Capture 0	cap0	10Gb	UP	0D 23H 48M	10G SR	850nm
Capture 1	cap1	10Gb	UP	0D 23H 48M	10G SR	850nm

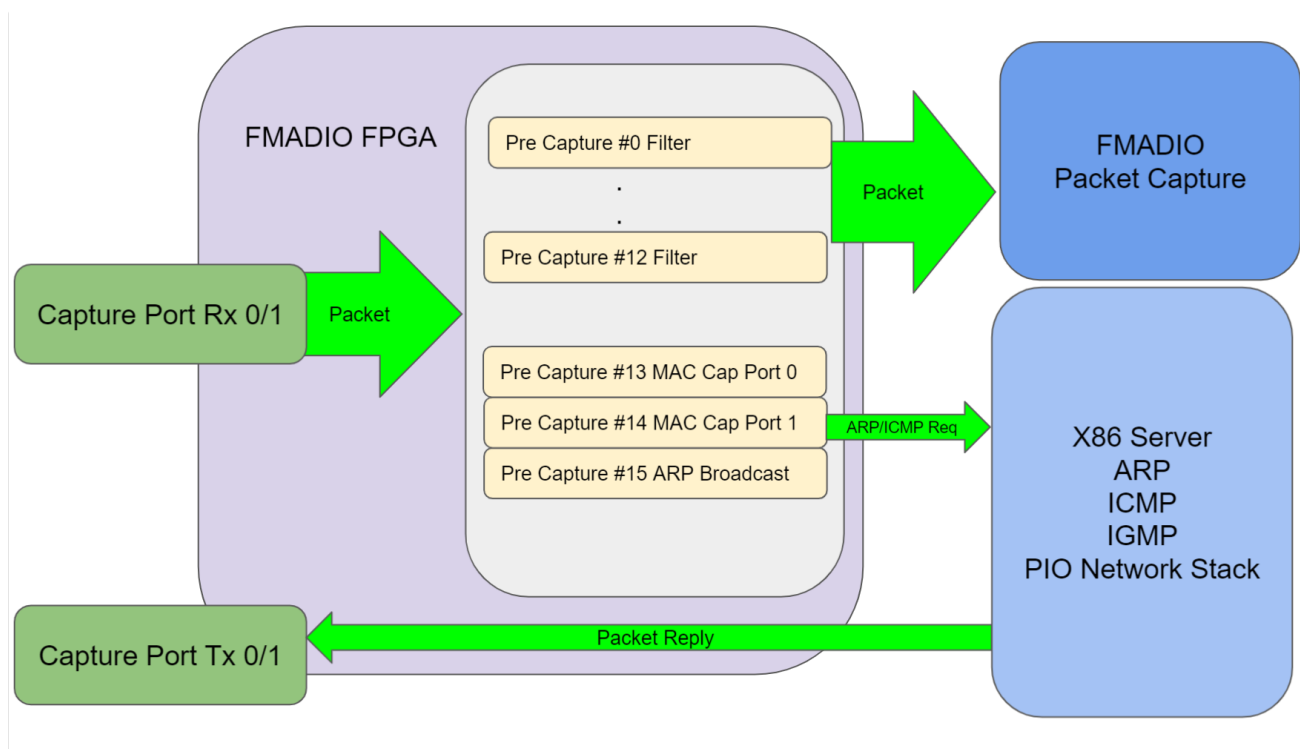
# Capture Port IP/MAC

FW Version: 7130+

By default FMADIO devices capture ports operate without any MAC or IP information. It receives and records any and all ethernet traffic on the wire. Its essentially a black hole high speed data recorder.

However there are some situations where the Capture interfaces need an IP MAC address, this is for ERSPAN IP targets, and also having the capture ports directly join Multicast groups. The follow demonstrates how to setup IP MAC Address,

Using FMAIO DPI Engine we can filter out low bandwidth traffic such as ARP/ICMP requests without any effect on the 100Gbps / 149Mpps packet capture performance. As seen below using a few of the PreCapture filter rules and forwarding a few packets to our ARP/ICMP/IGMP software network stack running on the x86 Server. This allows full ARP and ICMP protocol support on the capture interfaces.



FMADIO Packet Capture DPI ARP ICMP Network Architecture

NOTE: Enabling this feature reduces the total number of Pre Capture filter rules, It requires

Rule Number	Description
15	MAC Broadcast ARP Requests
14	Capture Port 0 MAC filter
13	Capture Port 1 MAC Filter

Depending on the capture port port configuration (2 ports or 8 ports), a maximum of 9 pre-capture filter rules may be used.

## CONFIG

To Configure IP/MAC information for the capture ports. edit the files below, by adding 2 new sections ["cap0"] and ["cap1"] i the configuration file

```
/opt/fmadio/etc/network.lua
```

Edit the file such that it look like the following, adding in your own MAC and IP addresses. There is no Netmask or Gateway associated with the capture ports.

```
fmadio@fmadio20v2-149:~$ cat /opt/fmadio/etc/network.lua
-- auto generated on Wed Dec 26 20:13:20 2018
local Config =
{
.
.
.
.
.

["cap0"] =
{
    ["MAC"]      = "00:11:22:33:44:55",
    ["Address"]  = "192.168.15.170",
},
["cap1"] =
{
    ["MAC"]      = "00:66:77:88:99:aa",
    ["Address"]  = "192.168.15.171",
},
}
return Config
```

Save the file and reboot the system.

The capture interfaces MAC/IP/IGMP is only enabled when there is an active capture. Please start a capture then try arping the interface first

```
fmadio@fmadio100v2-228U:~$ sudo arping -I man10 192.168.15.170
ARPING to 192.168.15.170 from 192.168.15.175 via man10
Unicast reply from 192.168.15.170 [00:11:22:33:44:55] 0.097ms
Unicast reply from 192.168.15.170 [00:11:22:33:44:55] 0.044ms
Unicast reply from 192.168.15.170 [00:11:22:33:44:55] 0.041ms
Unicast reply from 192.168.15.170 [00:11:22:33:44:55] 0.040ms
Unicast reply from 192.168.15.170 [00:11:22:33:44:55] 0.040ms
^CSent 5 probe(s) (1 broadcast(s))

Received 5 reply (0 request(s), 0 broadcast(s))
fmadio@fmadio100v2-228U:~$
```

If that is successful then check ICMP ping is functioning correctly also

```
fmadio@fmadio100v2-228U:~$ ping 192.168.15.170
PING 192.168.15.170 (192.168.15.170): 56 data bytes
64 bytes from 192.168.15.170: seq=1 ttl=64 time=40.726 ms
64 bytes from 192.168.15.170: seq=2 ttl=64 time=0.069 ms
64 bytes from 192.168.15.170: seq=3 ttl=64 time=0.075 ms
64 bytes from 192.168.15.170: seq=4 ttl=64 time=0.097 ms
64 bytes from 192.168.15.170: seq=5 ttl=64 time=0.089 ms
^C
--- 192.168.15.170 ping statistics ---
6 packets transmitted, 5 packets received, 16% packet loss
round-trip min/avg/max = 0.069/8.211/40.726 ms
fmadio@fmadio100v2-228U:~$
```

Any problems please check the log file

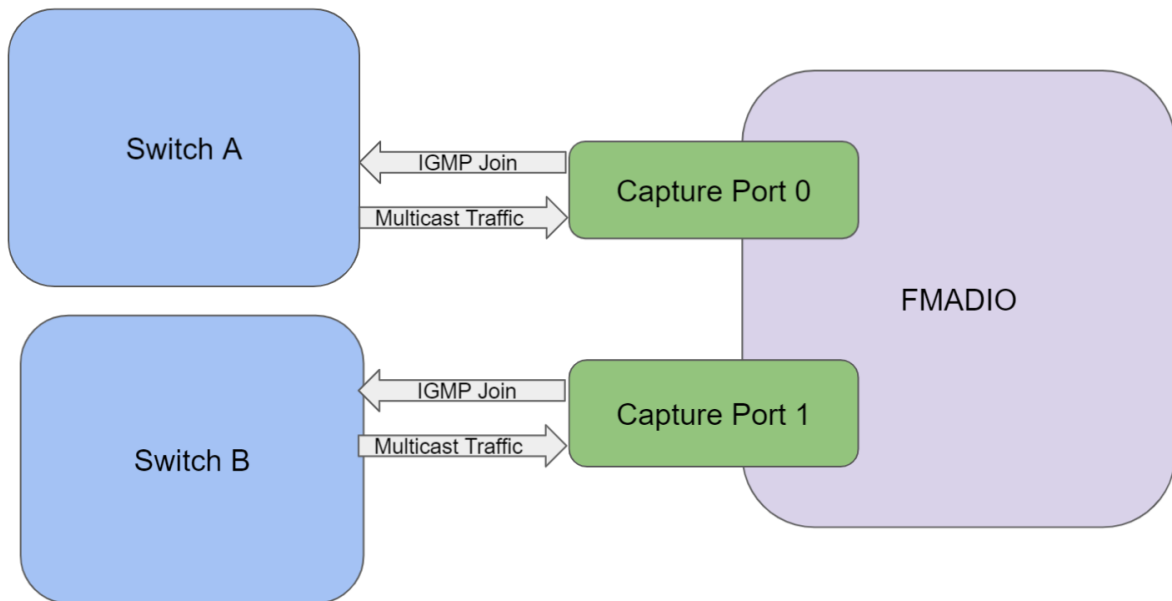
```
/mnt/store0/log/fnic_ping.cur
```

It may provide more information on why ARP or ICMP is not responding correctly

# Multicast IGMP Subscribe

FW Version: 7130+

There are many cases where the FMADIO Packet Capture system should subscribe to Multicast traffic via the capture Port.



In the above example each capture port is connected to 2 separate switches (e.g. Market Data Feed A and B) such that each capture port receives the a direct Market data feed, instead of relying on SPAN/Mirror sessions in a passive setup.

To configure start with enabling the capture ports IP/MAC address as per



Capture Port IP/MAC

Next is configuring the IGMP groups and port configuration. Example configuration for the OPRA Options market data feed is as follows. Edit or create the file

```
/opt/fmadio/etc/igmp.lua
```

Example igmp.lua file is as follows.







Note each Capture port (Port0/Port1) is subscribing to a different MC Group. Also the interval for broadcasting the joins by default is 60sec. In the above example this is reduced to 30sec (Interval = 30e9 is in nanoseconds)

After editing the file confirm the syntax is correct by running the command as follows

```
fmadio@fmadio100v2-228U:~$ fmadiolua /mnt/store0/etc/igmp.lua
fmad fmadlua May 22 2021
calibrating...
0 : 2095078768          2.0951 cycles/nsec offset:4.921 Mhz
Cycles/Sec 2095078768.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
argv fmadiolua
failed to open self? [fmadiolua]
loading filename [/mnt/store0/etc/igmp.lua]
done 0.000352Sec 0.000006Min
fmadio@fmadio100v2-228U:~$
```

There should not be any syntax errors or error statements.

Changes to this file requires the capture to be restarted. Please stop the current capture, then restart it. After the restart the system will issue the IGMP joins based on the above.

Troubleshooting logfile can be seen in /mnt/store0/log/fnic\_ping.cur this logfile should list all the IGMP joins that get issued, example as follows.

```
Port cap0 Change 00:af:a0:02:00:00 192.168.15.170
Port cap1 Change 00:af:a0:02:01:00 192.168.15.171
Port cap0 Change 00:af:a0:02:00:00 192.168.15.170
Port cap1 Change 00:af:a0:02:01:00 192.168.15.171
Port cap0 Change 00:af:a0:02:00:00 192.168.15.170
Port cap1 Change 00:af:a0:02:01:00 192.168.15.171
IGMP cap0: 0 : VLAN: 0 IGMP Group:233. 43.202. 1 Port: 11101
IGMP cap0: 1 : VLAN: 0 IGMP Group:233. 43.202. 2 Port: 11102
IGMP cap0: 2 : VLAN: 0 IGMP Group:233. 43.202. 3 Port: 11103
IGMP cap0: 3 : VLAN: 0 IGMP Group:233. 43.202. 4 Port: 11104
IGMP cap0: 4 : VLAN: 0 IGMP Group:233. 43.202. 5 Port: 11105
IGMP cap0: 5 : VLAN: 0 IGMP Group:233. 43.202. 6 Port: 11106
IGMP cap0: 6 : VLAN: 0 IGMP Group:233. 43.202. 7 Port: 11107
IGMP cap0: 7 : VLAN: 0 IGMP Group:233. 43.202. 8 Port: 11108
IGMP cap0: 8 : VLAN: 0 IGMP Group:233. 43.202. 9 Port: 11109
.
.
.
.
```



# Forward Error Correction Support (FEC)

FW Version: 7130+

Support for FEC is built into all 100G capable Packet Capture systems. By default its turned off and requires manual setting to enable the port to link up.

## Configuration

Enabling FEC support requires editing the file

```
/opt/fmadio/etc/network.lua
```

Add in cap0 and cap1 interface settings into the file if they do not exist, such as the following

```
,
["cap0"] =
{
    ["Address"] = "192.168.15.170",
    ["MAC"]     = "00:af:a0:02:00:00",
    ["FEC"]     = true,
},
["cap1"] =
{
    ["Address"] = "192.168.15.171",
    ["MAC"]     = "00:af:a0:02:01:00",
    ["FEC"]     = true,
},
```

Address and MAC settings are not required, only if you need the capture port to have an IP/MAC address

After updating the file check the syntax is correct by running the following command

```
fmadiolua /opt/fmadio/etc/network.lua
```

The output should look like the following, without any errors or warnings about the syntax.

```

fmadio@fmadio100v2-228U:~$ fmadiolua /opt/fmadio/etc/network.lua
fmad fmadlua May 22 2021
calibrating...
0 : 2095072184          2.0951 cycles/nsec offset:4.928 Mhz
Cycles/Sec 2095072184.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
argv fmadiolua
failed to open self? [fmadiolua]

loading filename [/opt/fmadio/etc/network.lua]
done 0.000143Sec 0.000002Min
fmadio@fmadio100v2-228U:~$

```

Once complete, please reboot the system and FEC should be enabled on boot.

### Manual FEC settings

FEC settings can be overridden and set manually per the following commands

Force FEC on all ports

```

fmadio@fmadio100v2-228U:~$ sudo fnic_test --fec-force
FEC Force
calibrating...
0 : 2095071400          2.0951 cycles/nsec offset:4.929 Mhz
Cycles/Sec 2095071400.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
PCIVersion: 50434930 50434930
FECENable: 1 PortMask:0003
[0] FECENable: 1 FECForce:1
[1] FECENable: 1 FECForce:1

```

Disable FEC on all ports

```

fmadio@fmadio100v2-228U:~$ sudo fnic_test --no-fec-force
no FEC Force
calibrating...
0 : 2095073970          2.0951 cycles/nsec offset:4.926 Mhz
Cycles/Sec 2095073970.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
PCIVersion: 50434930 50434930
FECENable: 0 PortMask:0003
[0] FECENable: 0 FECForce:0
[1] FECENable: 0 FECForce:0
fmadio@fmadio100v2-228U:~$

```

# Syslog

Requires FW:6761+

In large server deployments using remote syslogd where syslog entries are written over UDP is quite helpful. This allows a central server to monitor a fleet of servers by receiving all log entries over the network. This is a standard linux feature set. FMADIO Packet Capture devices support this feature, as follows:

Copy the default syslogd.conf to /opt/fmadio/etc/

```
sudo cp /etc/syslogd.conf /opt/fmadio/etc/
```

Then edit the file as follows, replacing the destination IP with configuration specific to your environment

```

# rsyslog configuration file (fmadio default)

##### MODULES #####

module(load="imuxsock") # provides support for local system logging (e.g. via logger command)
module(load="imklog")   # provides kernel logging support (previously done by rklogd)
module(load="immark")  # provides --MARK-- message capability

# Provides UDP syslog reception
# for parameters see https://www.rsyslog.com/doc/imudp.html
#module(load="imudp") # needs to be done just once
#input(type="imudp" port="514")

# Provides TCP syslog reception
# for parameters see https://www.rsyslog.com/doc/imtcp.html
#module(load="imtcp") # needs to be done just once
#input(type="imtcp" port="514")

##### GLOBAL DIRECTIVES #####

template(name="facility_priority" type="list") {
    property(name="syslogfacility-text")
    constant(value=".")
    property(name="syslogpriority-text")
}
set $!facility_priority = exec_template("facility_priority");

template(name="syslog_fmadio" type="list") {
    property(name="timereported" dateFormat="year")
    constant(value=".")
    property(name="timereported" dateFormat="month")
    constant(value=".")
    property(name="timereported" dateFormat="day")
    constant(value="-")
    property(name="timereported" dateFormat="hour")
    constant(value=":")
    property(name="timereported" dateFormat="minute")
    constant(value=":")
    property(name="timereported" dateFormat="second")
    constant(value=".")
    property(name="timereported" dateFormat="subseconds")
    constant(value=" ")
    constant(value="(")
    property(name="timereported" dateFormat="tzoffsetdirection")
    property(name="timereported" dateFormat="tzoffsethour")
    constant(value=":")
    property(name="timereported" dateFormat="tzoffsetmin")
    constant(value=") | ")
    property(name="hostname")
    constant(value=" | ")
}

```

```

property(name="!facility_priority" position.to="16" fixedwidth="on")
constant(value="| ")

property(name="programname" position.to="10" fixedwidth="on")
constant(value="|")
property(name="msg" spifno1stsp="on")
property(name="msg" droplastlf="on")
constant(value="\n")
}

# Use default timestamp format
#$ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat
$ActionFileDefaultTemplate syslog_fmadio

# File syncing capability is disabled by default. This feature is usually not required,
# not useful and an extreme performance hit
#$ActionFileEnableSync on

# Include all config files in /etc/rsyslog.d/
#$IncludeConfig /etc/rsyslog.d/*.conf

##### RULES #####

# Log all kernel messages to the console.
# Logging much else clutters up the screen.
kern.err                                /dev/console

# log everything to disk
*.*                                     /mnt/store0/log/messages

# remote host is TCP: name/ip:port, e.g. 192.168.0.1:514, port optional
*.* @192.168.1.100:514

```

In the above example all syslog log entries are also written to a server at 192.168.1.100 over TCP on port 514.

For UDP on port 514 use the following setting

```

// remote host over UDP is name/ip;port
*.* @192.168.1.100:514

```

Its the standard syslogd from inted package additional customization can be done if required. Example syslog output as follows

```
Aug 12 21:06:36 box local7.info fmadio: Capture (Enb: 0 Pkt: 0 Drop: 0 FCSError: 0 Capture
Aug 12 21:06:36 box local7.info fmadio: Mem (0.00GB ECC 0) Writeback (0.00GB) Dropped
Aug 12 21:06:46 box local7.info fmadio: Temp (CPU0:33.00 CPU1:33.00 PCH:41.00 SYS:36.00
Aug 12 21:06:46 box local7.info fmadio: Fan (SYS0:13650 SYS1:13800 SYS2:13800 SYS3:135
Aug 12 21:06:46 box local7.info fmadio: Disk OS (Temp:27 ERR: 0 ) SSD (Valid:1 1 1 0 Te
Aug 12 21:06:46 box local7.info fmadio: Link Capture (1 1 0 0 0 0 0 0 ) Man (1G 1 10G 0
Aug 12 21:06:46 box local7.info fmadio: DiskIO (Rd: 0.00Gbps Wr: 0.00Gbps)
```

# Automatic Push PCAP

## Requires FW:6979+

FMADIO Packet Capture systems provide a built in Push mode to transfer capture PCAP data on a regular schedule to a remote system. An example is pushing 1minute PCAPs to a remote NFS share or an S3 storage bucket.

Configuration is via configuration scripts located:

```
/opt/fmadio/etc/push_pcap.lua
```

If there is no such file above, please copy the basic example from the following location

```
/opt/fmadio/etc_ro/push_pcap.lua.basic
```

An example is shown as follows:

```

local Config = {}

Config.TimeoutRing      = 5*60e9
Config.Target           = {}

table.insert(Config.Target,
{
    Desc      = "Full",
    Mode      = "File",
    Path      = os.date("/mnt/remote0/pcap/%Y%m%d/all-"),
    Split     = "--split-time" .. (60*60*1e9),
    SplitCmd  = "-Z fmadio",
    FileName  = "--filename-tstr-HHMMSS",
    FilterBPF = "",
    PipeCmd   = "zstd -c -T8",
    FileSuffix= ".pcap.zstd",
})

table.insert(Config.Target,
{
    Desc      = "tcp_192_168_1_0",
    Mode      = "File",
    Path      = os.date("/mnt/remote0/pcap/%Y%m%d/tcp_host-"),
    Split     = "--split-time" .. (60*60*1e9),
    SplitCmd  = "-Z fmadio",
    FileName  = "--filename-tstr-HHMMSS",
    FilterBPF = "net 192.168.1.0/24",
    PipeCmd   = "zstd -c -T8",
    FileSuffix= ".pcap.zstd",
})

return Config

```

Multiple push targets can be specified, there is no real limit however throughput does get effected.

In the above example there are 2 push rules

A) Push all packet data (no filter)

This push target sends all PCAP data the remote NFS share mounted on

/mnt/remote0

See NFS mount configuration section for details on setting up /mnt/remote0 mounting points.

The sepcified is "FilterBPF=nil" meaning there is no filter, thus all traffic is pushed

B) Push all TCP data from network 192.168.1.0/24



The second example shows pushing all TCP data on the network 192.168.1.0/24 to the specified /mnt/remote0/push/ directory with a PCAP file prefix of "tcp\_\*

Note `FilterBPF=net 192.168.1.0/24 and tcp` This applies a full BPF (Berkeley Packet Filter [https://en.wikipedia.org/wiki/Berkeley\\_Packet\\_Filter](https://en.wikipedia.org/wiki/Berkeley_Packet_Filter)) with the filter "tcp" on the packets before writing it to the location. This results in only TCP data written to the /mnt/remote0/push/tcp\_\*.pcap output files

## Supported Endpoints

Mode	Description
linux file	linux file on FMADIO capture system
NFS	remote NFS mountpoint on FMADIO capture system
SFTP	remote SSH file system via rclone ( <a href="https://rclone.org/sftp/">https://rclone.org/sftp/</a> )
FTP	FTP push via rclone ( <a href="https://rclone.org/ftp/">https://rclone.org/ftp/</a> )
S3	S3 protocol via rclone ( <a href="https://rclone.org/s3/">https://rclone.org/s3/</a> )
Google Drive	Google drive via rclone ( <a href="https://rclone.org/drive/">https://rclone.org/drive/</a> )
Digital Ocean	Digital Ocean Spaces via rclone ( <a href="https://rclone.org/s3/#digitalocean-spaces">https://rclone.org/s3/#digitalocean-spaces</a> )
Azure Blob	Microsoft Azure Blob via rclone ( <a href="https://rclone.org/azureblob/">https://rclone.org/azureblob/</a> )
Dropbox	Dropbox via rclone ( <a href="https://rclone.org/dropbox/">https://rclone.org/dropbox/</a> )
Hadoop HDFS	Hadoop file system via rclone ( <a href="https://rclone.org/hdfs/">https://rclone.org/hdfs/</a> )
Ceph	Ceph S3 interface via rclone ( <a href="https://rclone.org/s3/">https://rclone.org/s3/</a> )

and many more, see the rclone documentation for full list of endpoints supported

<https://rclone.org/docs/>

## Command Reference

Following is a description of each option for per push target.

### Desc

### Required

Provides a text human readable description for each push target. It is also used for for log file description

```
Desc = "pcap-all",
```

For example the above push logfiles will go to /mnt/store0/log/push\_pcap-all\_\* this can be helpful for troubleshooting any problems

## Mode

### Default (FILE)

Specifies how the output files are written. Currently there are 2 modes, standard linux file "File" and rclone which provides multiple end points such as FTP, S3, Google Drive, Azure Cloud and many more.

```
Mode = "FILE",
```

## Options

Command	Description
FILE	output a regular linux file. this can be In the local file system or over a remote NFS mount
RCLONE	<p>use rclone as the end point file. Note rclone needs to be setup and configured before remote push is started</p> <p>For RCLONE Config please see their documentation <a href="https://rclone.org/commands/rclone_config/">https://rclone.org/commands/rclone_config/</a></p> <p>FMADIO by default stores config file into <code>/opt/fmadio/etc/rclone.conf</code></p> <p><b>Requires FW:7157+</b></p>
LXC	<p>Writes output to the LXC ring buffer. Location of ring buffer is the Path variable e.g</p> <p><code>/opt/fmadio/queue/lxc_ring0</code></p> <p><b>Requires FW:7738+</b></p>
CURL	Pipe down a CURL pipe, e.g. to FTP or HTTP POST

## Path

### Required

Full remote path of the target PCAPs + the leading prefix of the remote output.

```
Path = "/mnt/remote0/push/all",
```

The above example uses the "FILE" mode, which specifies a full linux system file path.

Command	Description
<code>/mnt/remote0/push/all</code>	FILE mode output PCAP files will be written for example as <code>/mnt/remote0/push/all_20210101_010101.cap</code>
<code>gdrive://pcap/all</code>	RCLONE mode output PCAP files written be written to the rclone configured google drive endpoint into the google drive directory <code>/pcap</code>

## Split

### Required

This specifies how to split the incoming PCAP data, either by Bytes or by Time. Following example is splitting by Time

```
Split = "--split-time 60e9",
```

Command	Description
<code>--split-time &lt;nano seconds&gt;</code>	Splits PCAP data by time, argument is in nanoseconds Scientific notation can be used
<code>--split-byte &lt;bytes&gt;</code>	Splits PCAP data by Size. argument is in bytes, Scientific notation can be used

## FileName

### Required

Specifies how to split filename is encoded. Different downstream applications require specific encodings. If your downstream applications need an encoding not listed, please contact us for support.

```
FileName = "--filename-epoch-sec-startend",
```

Command	Description
<code>--filename-epoch-sec-startend</code>	writes the sec epoch start/end time as the file name. e.g March 21, 2021 1:50:55 <code>1616334655-1616334755.pcap</code>
<code>--filename-epoch-sec</code>	writes the sec epoch start time as the file name. e.g March 21, 2021 1:50:55 <code>1616334655.pcap</code>
<code>--filename-epoch-msec</code>	Writes the epoch start time in milliseconds as the file name e.g for April 22 02:48 GMT <code>fmadio_1650595712592.pcap</code>
<code>--filename-epoch-usec</code>	Writes the epoch start time in microseconds e.g For April 22 02:48 GMT <code>fmadio_1650585598007301.pcap</code>
<code>--filename-epoch-nsec</code>	Writes the epoch start time in nanoseconds e.g For April 22 02:48 GMT <code>fmadio_1650585598007301462.pcap</code>
<code>--filename-tstr-HHMM</code>	writes the YYYYMMDD_HHMM style file name. e.g. 2021 Dec 1st 23:50 <code>20211201_2350.pcap</code>
<code>--filename-tstr-HHMMSS</code>	writes the YYYYMMDD_HHMMSS style file name. e.g. 2021 Dec 1st 23:50:59 <code>20211201_235059.pcap</code>
<code>--filename-tstr-HHMMSS_NS</code>	writes the YYYYMMDD_HHMMSS.MSEC.USEC.NSEC style file name. e.g. 2021 Dec 1st 23:50:59 <code>123456789nsec 20211201_235059.123.45789.pcap</code>

<pre>--filename-tstr-HHMMSS_TZ</pre>	<p>Writes the filename in Hour Min Sec with a local timezone suffix</p> <p>e.g 2022 April 22 19:59 CST</p> <pre>fradio__2022-04-21_19:59:58-04:00.pcap</pre>
<pre>--filename-strftime &lt;time string&gt;</pre>	<p>Generic strftime print</p> <p>e.g command line</p> <pre>--filename-strftime "%Y%m%d%H%M%S"</pre> <p>Output is as follows</p> <pre>fradio__20220421224832.pcap</pre>

## FilterBPF

### Default (nil)

Full libpcap BPF filter can be applied to reduce the total PCAP size or segment specific list of PCAPs . The system uses the native libpcap library, everything that tcpdump supports FilterBPF also supports.

```
FilterBPF = "net 192.168.1.0/24 and tcp"
```

The above is an example BPF filter "net 192.168.1.0/24 and tcp" its a slightly more complicated BPF and shows the flexibility and wide range of options available. Technically there is no limit on the complexity of the BPF filter, we recommend to keep it as simple as possible to reduce the CPU load

Command	Description
FilterBPF	<p>Enter a full tcpdump equivalent BPF filter express</p> <p>example host filter</p> <pre>FilterBPF="host 192.168.1.1"</pre>

## Decap

### Default (true)

In addition to FilterBPF full packet de-encapsulation is performed by default before the BPF filter is applied. This for example can decode VLAN, ERSPAN, GRE tunnels and many more. It enables the BPF filter is applied on the inner payload instead of the encapsulated output,

Example to disable automatic De-encapsulation

```
Decap = false,
```

Configuration is a simple boolean type only

Command	Description
Decap	boolean value of "true" enables Packet De-encapsulation (Default true)

## PipeCmd

**Requires FW:7157+**

**Default (nil)**

Pipe commands are processed on a per PCAP split basis before the end transport is applied. Examples to use this are to GZIP or compress files before sending to the endpoint.

This is a generic stdin/stdout linux application, gzip, lz4 are current examples, Other options are possible, please contact us for more details

```
PipeCmd="gzip -1 -c"
```

The above runs gzip with compression level 1 on the split PCAP before sending to the output location. Some examples are shown below

Command	Description
gzip -c -1	Run GZIP on split PCAPs with fastest compress mode
gzip -c -9	Run GZIP on split PCAPs in maximum compression mode
lz4 -c	Run LZ4 compression on split PCAPs for fast compression

## FileSuffix

**Requires FW:7157+**

**Default (nil)**

By default the split PCAP filename suffix is `.pcap`. For most operations that is sufficient, however for more complicated operations such as GZIP compressing with PipeCmd a `.pcap.gz` file suffix is more appropriate. The Following is an example config target that compresses and outputs splits in `.pcap.gz` file format

```
table.insert(Config.Target,
{
  Desc      = "pcap-all-gz",
  Mode      = "rclone",
  Path      = "s3-fmad://pcap/all",
  Split     = "--split-time 60e9",
  FileName  = "--filename-tstr-HHMMSS",
  FilterBPF = nil,
  PipeCmd   = "gzip -c",
  FileSuffix = ".pcap.gz"
})
```

The above example pushes gzip 1minute PCAP splits to an S3 protocol storage device

Command	Description
<code>.pcap</code>	Default suffix.
<code>.pcap.gz</code>	GZIP Compressed PCAP
<code>.pcap.lz4</code>	LZ4 compressed PCAP

## Chunked

**Requires FW: 7355+**

**Default: (nil)**

Chunked mode is a more optimized processing mode. It increases the aggregate throughput of the Push operation specifically for network traffic profiles skewed towards small packets.

By default its disabled.

Example as follows

```
Chunked = true,
```

Text	Description
<code>true</code>	Enables chunk mode



## FollowStart

### Requires FW: 7355+

FollowStart forces the push to start from the beginning of the capture. If its disabled it will push from the current capture position.

Default is "false" push from the current capture position

Example as follows

```
FollowStart = true,
```

Text	Description
false	Push from the current capture position
true	Push from the start of the currently active capture

## CPU

### Requires FW: 7750+

Sets a specific CPU for stream\_cat to run on by overriding the default CPU setting. This is helpful when multiple pushes are running in parallel.

Default: the system assigned CPU number for push (typically CPU 23)

Example as follows

```
CPU = 30,
```

Setting	Description
nil	Uses the default system CPU for push operation
<numeric value>	Literal Numeric value indicating which CPU to run stream_cat on

## Consumer Application Restart

The default behavior of the system is to constantly re-try sending data downstream without loss. In some cases its better to restart the push process and reset the start sending position, when the consumer application restarts.

An example is, if an application shuts down between 1AM and 6AM but the capture process runs 24/7. The application wants to only receive data starting at 6AM when the application starts up.

Another use case is, if the application has an error for an unspecified amount of time. The application requires real-time processing, and requires FMADIO Capture system to send data from the current time now without sending old historical data.

Adding the following configuration to the push\_realtime.lua config will cause stream\_cat to exit if the downstream consumer is unable to process data. FMADIO system scheduler will constantly restart the push process from the current time, until the consumer process starts processing data.

```
.  
.   
StreamCat = "--ring-timeout-exit ",  
FollowStart = false,  
.   
.
```

NOTE: Please keep the additional white space at the end of the command.

## Analytics Scheduler

In addition to configuration of

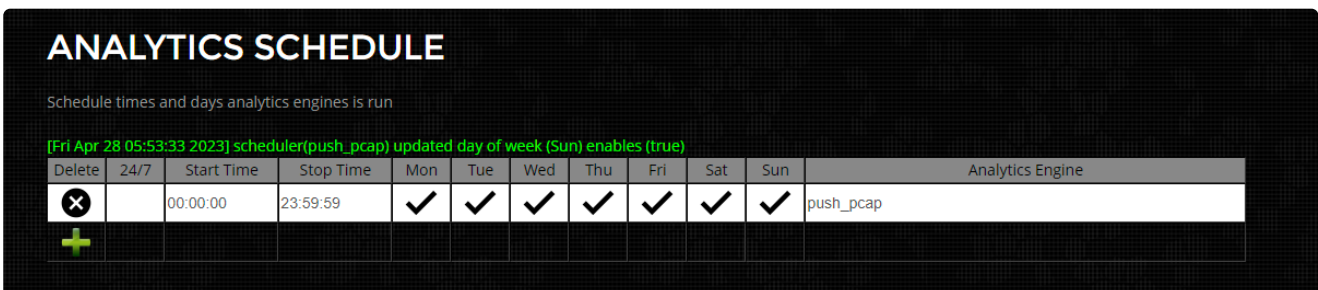
```
/opt/fmadio/etc/push_pcap.lua
```

To specify when the Push operation occurs the Analytics scheduler must be configured. This is on the "CONFIG" tab of the FMADIO GUI. An Example configuration to push files 24/7,

The "Analytics Engine" field must be exactly the following text.

```
push_pcap
```

Screenshot of 24/7 schedule is shown below



Troubleshooting

Logfiles

Configuration problems often occur when setting up the system. The following log files can be used to debug

```
/mnt/store0/log/analytics_push_pcap.cur
```

Monitoring the output can be as follows

```
fmadio@fmadio20v3-287:/mnt/store0/log$ tail -F analytics_push_pcap.cur
[Sat May 29 15:36:46 2021] push [pcap-all] : stream_cat:true
[Sat May 29 15:37:01 2021] push [pcap-all] : stream_cat:true
[Sat May 29 15:37:16 2021] push [pcap-all] : stream_cat:true
[Sat May 29 15:37:31 2021] push [pcap-all] : stream_cat:true
[Sat May 29 15:37:46 2021] push [pcap-all] : stream_cat:true
[Sat May 29 15:38:01 2021] push [pcap-all] : stream_cat:true
[Sat May 29 15:38:16 2021] push [pcap-all] : stream_cat:true
```

In addition each Push entry has a log file with the following format. The Desc value is described

<https://docs.fmad.io/fmadio-documentation/configuration/automatic-push-pcap#desc>

```
/mnt/store0/log/push_<Desc value>_YYYYMMDD_HHMM
```

Example output of correct functionality is as follows

```

fmadio@fmadio20v3-287:/mnt/store0/log$ head -n 100 push_pcap-all_20210524_1851
args
  --uid
  push_1621849863557502976
  -o
  /mnt/remote0/push/
  --split-time
  60e9
  --filename-epoch-sec-startend
--uid
UID [push_1621849863557502976]
-o
--split-time
Split Every 4768169126130614272 Sec
--filename-epoch-sec-startend
Filename EPOCH Sec Start/End
PCAP Nano
[0.001 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.
[0.001 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.
[0.001 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.
[0.001 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.
[0.001 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.
[0.001 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.
[0.002 H][2021-05-24 18:51:04] /mnt/remote0/push/1621849860-1621849920.pcap : Total Bytes 0.

```

## Manual Offline mode

In addition to log files its sometimes easier to debug via the CLI interface, by manually starting the push on specific capture files. This can also be helpful to push historical PCAP files.

This is done as the following CLI command

```

sudo /opt/fmadio/analytics/push_pcap.lua --force --offline <capture name>

```

Replace capture name with the complete name of the capture. Also ensure push scheduler has been disabled

Example output of successful offline mode run is shown below.

```
fmadio@fmadio20n40v3-363:~$ sudo /opt/fmadio/analytics/push_pcap.lua --force --offline capt
loading filename [/opt/fmadio/analytics/push_realtime.lua]
sudo /opt/fmadio/bin/stream_cat --uid push_1622272417078768896 capture1_20210521_0101 | /op
stream_cat UID [push_1622272417078768896]
stream_cat ioqueue: 4
[Sat May 29 16:13:37 2021] push [pcap-all] : stream_cat:true
StartChunkID: 28872404
StartChunk: 28872404 Offset: 0 Stride: 1
StartChunk: 28872404
[Sat May 29 16:13:52 2021] push [pcap-all] : stream_cat:true
[Sat May 29 16:14:07 2021] push [pcap-all] : stream_cat:true
.
.
.
```

Note the following repeated status line indicates the push is operating successfully

```
stream_cat:true split:true
```

For problems per push target, the logfile shown in the above command line here

```
/mnt/store0/log/push_pcap-all.cur
```

A good way to debug that is running `tail -F /mnt/store0/log/push_pcap-all.cur` to monitor it such as the following

```

$ tail -F /mnt/store0/log/push_pcap-all.cur
args
  --uid
  push_1622275884783852032
  -o
  ssh://mnt/store0/tmp2/rc1one/everything
  --split-time
  60e9
  --filename-tstr-HHMMSS
  --pipe-cmd
  gzip -c -1
  --rc1one
  --filename-suffix
  .pcap.gz
--uid
UID [push_1622275884783852032]
-o
--split-time
Split Every 4768169126130614272 Sec
--filename-tstr-HHMMSS
Filename TimeString HHMMSS
--pipe-cmd
pipe cmd [gzip -c -1]
--rc1one
Output Mode RClone
--filename-suffix
Filename Suffix [.pcap.gz]
PCAP Nano
[gzip -c -1 | rc1one --config=/opt/fmadio/etc/rc1one.conf rcat ssh://mnt/store0/tmp2/rc1one/
[0.000 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.000 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.001 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.001 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.001 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.002 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.002 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.002 H][2021-05-21 01:05:40] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
[0.003 H][2021-05-21 01:05:41] ssh://mnt/store0/tmp2/rc1one/everything_20210521_010600.pcap.
.
.
.

```

## Multiple Push Schedules

Multiple push\_pcap schedules can be added to the system, for example

Push A) Realtime 1min push

- used for realtime monitoring throughout the day

Push B) End of Day, recon push.

- Used for End of Day Recon pushing data to back office systems

This can be achieved with the following steps

### 1) create a directory for custom analytics schedules

```
mkdir /opt/fmadio/etc/analytics/
```

All files in this directory are sym linked to the /opt/fmadio/analytics directory used by the scheduler.

### 2) Copy the current push\_pcap loader and rename it

Create a new push\_pcap\_eod loader as follows

```
cp /opt/fmadio/analytics/push_pcap /opt/fmadio/etc/analytics/push_pcap_eod
```

Then Edit the file to load a different configuration

```
fmadio@fmadio100v2-228U:~$ cat /opt/fmadio/etc/analytics/push_pcap_eod
#!/bin/sh
ulimit -n 10000000
/opt/fmadio/analytics/push_pcap.lua --config /opt/fmadio/etc/push_pcap_eod.lua
fmadio@fmadio100v2-228U:~$
```

In this case the config file is called `push_pcap_eod.lua`

### 3) Configure the new push\_pcap\_eod.lua

Configure the new push\_pcap\_eod.lua file, it will require hand editing of the file, as fmadiocli only operates on the default configuration

### 4) Enable in the scheduler

Going to the GUI -> Config page add the new loader file into the schedule with the new loader file `push_pcap_eod`

Example is shown below

# ANALYTICS SCHEDULE

Schedule times and days analytics engines is run

Delete	24/7	Start Time	Stop Time	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Analytics Engine
✘		00:00:00	23:59:59	✓	✓	✓	✓	✓	✓	✓	push_pcap
✘		00:00:00	23:59:59	✓	✓	✓	✓	✓	✓	✓	push_lxc
✘		17:00:00	23:59:59	✓	✓	✓	✓	✓	✘	✘	push_pcap_eod
+											

## 5) Confirm operation

Log files for (push\_pcap\_eod) are named

```
/mnt/store0/log/analytics_push_pcap_eod.cur
```

## Performance testing

Push performance is critical and subject to multiple factors. The following provides a baseline test of different variables.

### Remote Write Performance

A first initial step is to confirm the writing to the remote file system has sufficient bandwidth. This is simply achieved running the commands

```
time sudo dd if=/dev/zero bs=1G count=20 > /mnt/remote0/test1.bin
```

Example run

```
fmadio@fmadio20v3-287:~$ time sudo dd if=/dev/zero bs=1G count=20 > /mnt/remote0/push/test1.
20+0 records in
20+0 records out
21474836480 bytes (20.0GB) copied, 30.170007 seconds, 678.8MB/s
real    0m 30.17s
user    0m 0.00s
sys     0m 22.03s
fmadio@fmadio20v3-287:~$
```



The above command writes a 20GB file to the remote file location. In this case its writing @ 678MB/sec ( 5.4Gbps throughput)

## Maintenance

### Force PCAP Link Layer Setting

To force the PCAP link layer type to Ethernet use the following CLI command on the PCAP

```
printf "\x1" | dd of=<path to pcap>.pcap bs=1 seek=20 count=1 conv=notrunc
```

The symptoms of this is unusual TCPDUMP output such as the following

```
fmadio@fmadio20v2-149:/mnt/remote0/push$ tcpdump -r blah_20211103_175955.pcap -nn | head
reading from file blah_20211103_175955.pcap, link-type NULL (BSD loopback)
17:59:55.587798 AF Unknown (3772079709) length 134:
0x0000: 2151 18c0 4db4 0e6c 0800 4500 0074 635f  !Q..M..l..E..tc_
0x0010: 4000 4011 5039 c0a8 02e1 c0a8 02af b380  @.@.P9.....
0x0020: 00a2 0060 cdeb 3056 0201 0104 0670 7562  ... ..0V.....pub
0x0030: 6c69 63a7 4902 0345 a965 0201 0002 0100  !ic.I..E.e.....
0x0040: 303c 300f 0608 2b06 0102 0101 0300 4303  0<0...+.....C.
0x0050: 070a d430 1806 0a2b 0601 0603 0101 0401  ...0...+.....
0x0060: 0006 0a2b 0601 0401 82ee 2b04 0130 0f06  ...+.....+.0..
0x0070: 0a2b 0601 0401 82ee 2b04 0102 0101 7a3a  .+.....+.....z:
0x0080: c461                                     .a
```

After setting the PCAP Link Layer setting using the above command the output is as follows

```
fmadio@fmadio20v2-149:/mnt/remote0/push$ tcpdump -r blah_20211103_175955.pcap -nn -XX | head
reading from file blah_20211103_175955.pcap, link-type EN10MB (Ethernet)
17:59:55.587798 IP 192.168.2.225.45952 > 192.168.2.175.162: V2Trap(73) .1.3.6.1.2.1.1.3.0=461524 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.1 .1.3.6.1.4.1.46891.4.1=1
0x0000: e085 585d 2151 18c0 4db4 0e6c 0800 4500  .!IQ..M..l..E.
0x0010: 0074 635f 4000 4011 5039 c0a8 02e1 c0a8  .tc.@.P9.....
0x0020: 02af b380 00a2 0060 cdeb 3056 0201 0104  .0V.....
0x0030: 0670 7562 6c69 63a7 4902 0345 a965 0201  .public.I..E.e..
0x0040: 0002 0100 303c 300f 0608 2b06 0102 0101  ...0<0...+.....
0x0050: 0300 4303 070a d430 1806 0a2b 0601 0603  .C....0...+....
0x0060: 0101 0401 0006 0a2b 0601 0401 82ee 2b04  .+.....+.....+
0x0070: 0130 0f06 0a2b 0601 0401 82ee 2b04 0102  .0...+.....+...
0x0080: 0101 7a3a c461                               .z:..a
```

Which contains the correctly decoded packets.

## Examples

Example Configuration files for refence

### Push to NFS Share with BPF Filter and 1 minute PCAPs

```

local Config = {}

Config.Target = {}

-- push all tcp data to /mnt/remote0/push/tcp_*.pcap
table.insert(Config.Target,
{
  Desc      = "nfs-tcp",
  Mode      = "File",
  Path      = "/mnt/remote0/push/tcp",
  Split     = "--split-time 60e9",
  FileName  = "--filename-epoch-sec-startend",

  FilterBPF = "net 192.168.1.0/24 and tcp"
})

return Config

```

## Push to NFS Share with BPF Filter and HHMMSS Timezone

### FW: 7936+

Example pushes a single UDP multicast group 1001 at 1 minute snapshots using an Hour Min Sec with Timezone filename.

```

local Config = {}

Config.Target = {}

-- push all multicast port 10001 data to /mnt/remote0/push/udp-10001_*.pcap
table.insert(Config.Target,
{
  Desc      = "udp-multicast-1001",
  Mode      = "File",
  Path      = "/mnt/remote0/push/udp-10001",
  Split     = "--split-time 60e9",
  FileName  = "--filename-tstr-HHMMSS_TZ",
  FilterBPF = "multicast and port 10001"
})

```

## Push to NFS Share 1min Splits with BPF Filter and LZ4 compression

Example pushes 1min PCAPs with a BPF filter (port 80) and applying LZ4 compression. LZ4 compression is fast and reasonably good compression rates.

```

local Config = {}
Config.Target = {}
table.insert(Config.Target,
{
    Desc      = "port80-lz4",
    Mode      = "File",
    Path      = "/mnt/store0/tmp2/push/udp-10001",
    Split     = "--split-time 60e9",
    FileName  = "--filename-tstr-HHMMSS_TZ",
    FilterBPF = "port 80",
    PipeCmd   = "lz4 -c",
    FileSuffix = ".pcap.lz4",
})

return Config

```

### Push to NFS share 1min Splits with BPF Filter and ZSTD compression

Example pushes 1min PCAPs with a BPF filter (port 80) and applying ZSTD compression. ZSTD is a new compression format with performance close to LZ4 but compression rates close to GZIP.

```

local Config = {}

Config.Target = {}

table.insert(Config.Target,
{
    Desc      = "port80 zstd",
    Mode      = "File",
    Path      = "/mnt/store0/tmp2/push/udp-10001",
    Split     = "--split-time 60e9",
    FileName  = "--filename-tstr-HHMMSS_TZ",
    FilterBPF = "port 80",
    PipeCmd   = "zstd -c",
    FileSuffix = ".pcap.zstd",
})

return Config

```

### Push to NFS/CIFS Share 1GB splits

#### FW: 7936+

Example pushes the raw data to a remote NFS/CIFS (Windows Share) splitting by 1GB file size writing a gzip compressed PCAP file to the remote location.

```

local Config = {}

Config.Target = {}

-- push everything to /mnt/remote0/push/capture*.pcap.gz at 1GB splits compressed gz
table.insert(Config.Target,
{
  Desc      = "capture",
  Mode      = "File",
  Path      = "/mnt/remote0/push/capture",
  Split     = "--split-size 1e9",
  FileName  = "--filename-tstr-HHMMSS",
  FileSuffix = ".pcap.gz",
  FilterBPF = ""
})

return Config

```

## Push to MAGPACK over FTP

```

local Config = {}

Config.Target = {}

table.insert(Config.Target,
{
  Desc      = "magpack-all",
  Mode      = "CURL",
  Path      = "ftp://192.168.1.100/device-prefix",
  Split     = "--split-byte 1e9",
  FileName  = "--filename-tstr-HHMMSS_SUB",
  FilterBPF = "not (host 192.168.1.105 or host 192.168.1.102)",
  Chunked   = true,
  FollowStart = true,
  ScriptNew = "/opt/fmadio/analytics/push_realtime_checkremote.lua 10"
})

return Config

```

## Push to an LXC 24/7

```

local Config = {}

Config.Target = {}

-- push all pcap data to lxc ring buffer 0
table.insert(Config.Target,
{
  Desc      = "lxc-all",
  Mode      = "LXC",
  Path      = "/opt/fmadio/queue/lxc_ring0",
  FilterBPF = nil
})

return Config

```

## Push to Multiple lxc\_ring 24/7

Example pushes different VLAN traffic to separate lxc\_rings

```

local Config = {}

Config.Target = {}

-- push vlan1 data to lxc ring buffer 0
table.insert(Config.Target,
{
  Desc      = "lxc-vlan0",
  Mode      = "LXC",
  Path      = "/opt/fmadio/queue/lxc_ring0",
  FilterBPF = "vlan 1"
})

-- push vlan2 data to lxc ring buffer 1
table.insert(Config.Target,
{
  Desc      = "lxc-vlan1",
  Mode      = "LXC",
  Path      = "/opt/fmadio/queue/lxc_ring1",
  FilterBPF = "vlan 1"
})

return Config

```

## Push to AWS S3 Bucket with Compression

Pushing captured PCAP data from the local device to AWS S3 Bucket can be done using the RCLONE support.

Below is an example push\_pcap.lua config file for that

```
-- autogenerated Thu Jun  9 20:07:14 2022 from fmadio_config
local Config = {}
Config.FollowStart = true
Config.Decap       = true
Config.Target      = {}
table.insert(Config.Target,
{
  ["Desc"]        = "S3Cloud",
  ["Mode"]        = "RCLONE",
  ["Path"]        = "fmadio-s3://fmadio-pcap/pcap/fmadio20p3-coffee/full",
  ["Split"]       = "--split-time 60e9",
  ["SplitCmd"]    = "",
  ["PipeCmd"]     = " gzip -c ",
  ["FileSuffix"]  = ".gz",
  ["FileName"]    = "--filename-tstr-HHMM",
  ["FilterBPF"]   = "",
  ["FilterFrame"] = "",
})
return Config
```

This uses gzip to compress the data. Also note we added a PreCapture filter to 64B Slice all traffic to AWS S3 IP address. This prevents the capture size for a run-away explosion.

Below is the resulting output in AWS S3

Amazon S3 > Buckets > fmadio-pcap > pcap/ > fmadio20p3-coffee/

Copy S3 URI

Objects Properties

Objects (40)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

Find objects by prefix

Name	Type	Last modified	Size	Storage class
full_S3Cloud_20220618_1848.gz.pending	pending	June 18, 2022, 17:47:13 (UTC+08:00)	5.1 MB	Standard
full_S3Cloud_20220618_1847.gz	gz	June 18, 2022, 17:47:11 (UTC+08:00)	6.0 MB	Standard
full_S3Cloud_20220618_1846.gz	gz	June 18, 2022, 17:46:11 (UTC+08:00)	13.9 MB	Standard
full_S3Cloud_20220618_1845.gz	gz	June 18, 2022, 17:45:10 (UTC+08:00)	54.5 MB	Standard
full_S3Cloud_20220618_1844.gz	gz	June 18, 2022, 17:44:48 (UTC+08:00)	10.9 MB	Standard
full_S3Cloud_20220618_1843.gz	gz	June 18, 2022, 17:44:43 (UTC+08:00)	8.9 MB	Standard
full_S3Cloud_20220618_1842.gz	gz	June 18, 2022, 17:44:38 (UTC+08:00)	7.3 MB	Standard
full_S3Cloud_20220618_1841.gz	gz	June 18, 2022, 17:44:33 (UTC+08:00)	6.9 MB	Standard
full_S3Cloud_20220618_1840.gz	gz	June 18, 2022, 17:44:28 (UTC+08:00)	12.2 MB	Standard
full_S3Cloud_20220618_1839.gz	gz	June 18, 2022, 17:44:23 (UTC+08:00)	31.8 MB	Standard
full_S3Cloud_20220618_1838.gz	gz	June 18, 2022, 17:44:14 (UTC+08:00)	33.2 MB	Standard
full_S3Cloud_20220618_1835.gz	gz	June 18, 2022, 17:36:21 (UTC+08:00)	799.2 MB	Standard
full_S3Cloud_20220618_1834.gz	gz	June 18, 2022, 17:35:23 (UTC+08:00)	1.1 GB	Standard
full_S3Cloud_20220618_1833.gz	gz	June 18, 2022, 17:34:19 (UTC+08:00)	1.1 GB	Standard
full_S3Cloud_20220618_1832.gz	gz	June 18, 2022, 17:33:18 (UTC+08:00)	817.7 MB	Standard
full_S3Cloud_20220618_1831.gz	gz	June 18, 2022, 17:32:32 (UTC+08:00)	532.3 MB	Standard

PCAP Push to S3

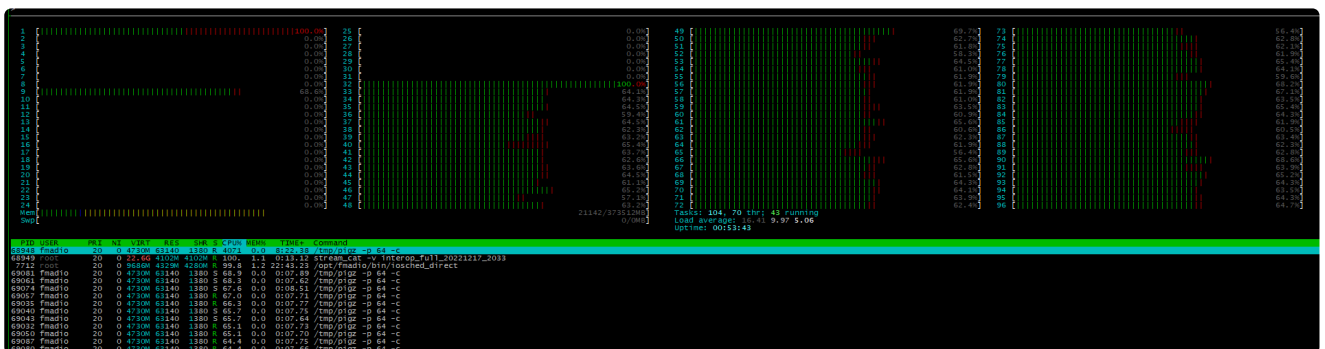
This does require RCLONE S3 Config to be configured before using.

## Performance

Performance of push varies by protocol filter and compression mode. Typically the remote push locations IO bandwidth is typically the bottleneck.

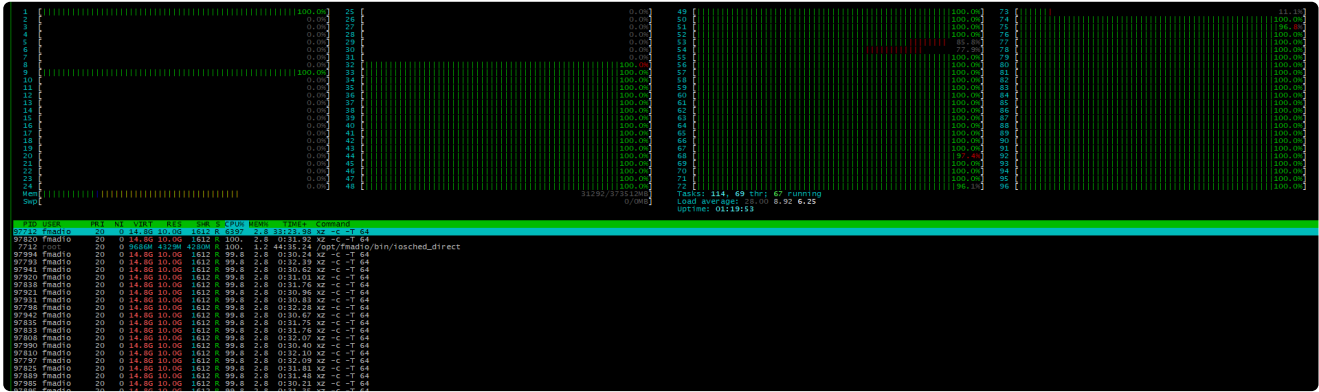
Here we are testing against a RAM disk on the local system. This removes the network and remote IO performance from the benchmark. Focusing entirely on the FMADIO fetch filter and compress performance.

Parallel GZIP (pigz) running on an FMADIO 100Gv2 Analytics System (96 CPUs total). Making good use of all the CPUs



Parallel GZIP 64 CPUs

XZ compression using 64 CPUs, more utilization than Parallel GZ



xz compression 64 CPUs

## Traffic Profile ISP

The dataset we are testing with is a WAN connection that has an ISP like L2-L7 packet distribution. As its ISP like traffic the data compression rate is not high.

Description	Gbps	Size	Ratio
raw (not compressed)	6.9 Gbps	45.5GB	1.0
lz4	2.5Gbps	41.2GB	x 1.104
gzip -1 (fast)	0.16Gbps	40.5GB	x 1.123
gzip default	0.15Gbps	40.2GB	x 1.131
pigz 64 CPU (Parallel GZIP with 64 CPUs)	4.8Gbps	40.3GB	x 1.129
pigz 32 CPU (Parallel GZIP with 32 CPUs)	4.4Gbps	40.3GB	x 1.129
pigz 8 CPU (Parallel GZIP with 8 CPUs)	1.04Gbps	40.3GB	x 1.129
zstd default	0.77Gbps	39.5GB	x 1.15
zstd --fast	2.4Gbps	40.3GB	x 1.129
xz default	0.019Gbps	39.2GB	x 1.160
xz 64 CPU (-T 64)	0.909Gbps	38.9GB	x 1.17
xz 32 CPU (-T 32)	0.502Gbps	39.3GB	x 1.157

## Traffic Profile Finance

The data set tested is a full days worth of OPRA A+B Feed dataset, raw uncompressed data size is just under 1TB. Financial data typically gets x2 to x3 compression ratio with xz maxing out at x5.



Description	Gbps	Time	Size	Ratio
raw (not compressed)	4.29Gbps	0.6H	979GB	x 1.0
lz4	1.65Gbps	1.3H	372GB	x 2.629
gzip -1 (fast)	0.341Gbps	6.3H	311GB	x 3.14
gzip (default)	0.129Gbps	16.6H	266GB	x 3.67
pigz 64 CPU (Parallel GZIP)	3.92Gbps	0.55H	264GB	x 3.70
pigz 32 CPU (Parallel GZIP)	3.92Gbps	0.55H	264GB	x 3.70
pigz 16 CPU (Parallel GZIP)	1.98Gbps	1.09H	264GB	x 3.70
pigz 8 CPU (Parallel GZIP)	0.997	2.1 H	264GB	x 3.70
zstd default	1.118Gbps	1.9H	255GB	x 3.83
zstd --fast	1.90Gbps	1.13H	294GB	x 3.32
xz default	0.012Gbps	181H	184GB	x 5.32
xz 64 CPU	0.604Gbps	3.6H	184GB	x 5.32
xz 32 CPU	0.372Gbps	5.8H	184GB	x 5.32
xz 16 CPU	0.192Gbps	11.3H	184GB	x 5.32
xz 8 CPU	0.096Gbps	22.6H	184GB	x 5.32

# Management Port Layout

## Configure 40G Management Port

### ANALYTICS SKU only

40G management ports can run in 2 x 40G or 4 x 10G mode or 2 x 2 x 10G The utility is located in

```
/opt/fmadio/drivers/current/qcu64e
```

### Get the current setting as follows

```
fmadio@fmadio100v2-228U:/opt/fmadio/drivers/4.9.247-tinycore64$ sudo ./qcu64e
Intel(R) QSFP+ Configuration Utility
```

```
QCU version: v2.34.17.03
Copyright(C) 2014 - 2019 by Intel Corporation.
Software released under Intel Proprietary License.
```

```
NIC Seg:Bus Ven-Dev   Mode   Adapter Name
=== =====
1) 000:026 8086-1583 2x40   Intel(R) Ethernet Server Adapter XL710-Q20CP
```

```
Warning: No adapter selected.
fmadio@fmadio100v2-228U:/opt/fmadio/drivers/4.9.247-tinycore64$
```

### To check the available port modes

To query what port configuration modes are available use

```
sudo ./qcu64e /nic=1 /info
```

As shown below

```
Intel(R) QSFP+ Configuration Utility
```

```
QCU version: v2.34.17.03
```

```
Copyright(C) 2014 - 2019 by Intel Corporation.
```

```
Software released under Intel Proprietary License.
```

```
Adapter supports QSFP+ Configuration modification.
```

```
Current Configuration: 2x40
```

```
Supported Configurations:
```

```
1x40
```

```
2x40
```

```
4x10
```

```
2x2x10A
```

```
2x2x10B
```

```
fmadio@fmadio100v2-228U:/opt/fmadio/drivers/4.9.247-tinycore64$
```

## Set a specific Port mode

Changing the port is simply

```
qcu64e /nic=1 /set=4x10
```

As follows

```
$ sudo ./qcu64e /nic=1 /set=4x10
```

```
Intel(R) QSFP+ Configuration Utility
```

```
QCU version: v2.34.17.03
```

```
Copyright(C) 2014 - 2019 by Intel Corporation.
```

```
Software released under Intel Proprietary License.
```

```
Changing the configuration...
```

```
Done.
```

```
Restart the system to apply the changes.
```

```
fmadio@fmadio100v2-228U:/opt/fmadio/drivers/4.9.247-tinycore64$
```

Then restart the system

# Mount Remote NFS (Linux) Drive

Mounting a remote NFS file system on the FMADIO capture device can be extremely useful in many cases.

It provides a simple way to process PCAPs on the Capture Device then writing the results out to a remote storage system.

Provides a simple and robust way to push PCAP data automatically from the capture device to an NFS share

## NFS Remote Mount Config

Please edit the configuration file in

```
/opt/fmadio/etc/disk.lua
```

Create an entry named "NFSDisk" as shown below, be careful all punctuation is correct. The mount point should be "remote0" or "remote1" or "remote2" which maps to /mnt/remote0 /mnt/remote1 on the capture device.

```
["NFSDisk"] =  
{  
    ["192.168.2.131:/home"] = "remote0",  
},
```

Mount options can be specified after a double colon : e.g. to mount using NFSv3 or specify locking behavior

```
["192.168.1.200:/home"] = "remote0:-o vers=3 -o nolock",
```

After configuration file has been updated, run the following command to confirm the config file has no syntax errors.

```
fmadio@fmadio20v3-287:/mnt/store0/etc$ fmadiolua /opt/fmadio/etc/disk.lua  
failed to open self? [fmadiolua]  
loading filename [/opt/fmadio/etc/disk.lua]  
done 0.000065Sec 0.000001Min  
fmadio@fmadio20v3-287:/mnt/store0/etc$
```

Above is the correct operation. Once complete please reboot the system to confirm the new mount points.

**NOTE:** if the NFS mount fails it may take 5 min to boot the system. This is the time it takes for the NFS client timeout and give up on mounting the remote partition. If long boot delays are seen please check the log messages in /mnt/store0/log/messages for reasons the NFS mount failed.

## NFS Remote Mount Example

Below is a a full disk.lua example configuration file for reference

```
return
{
  CacheDisk =
  {
    ["S3EWNX0K116564W"] = "ssd0",
    ["S3EWNX0K116582F"] = "ssd1",
    ["S3EWNX0K116574D"] = "ssd2",
    ["S3EWNX0K116592K"] = "ssd3",
  }
,
  RaidDisk =
  {
    ["6522KJS3FSAA"] = "hdd0",
    ["25Q8K2SXFSAA"] = "hdd1",
    ["357FK7NVFSAA"] = "hdd2",
  }
,
  ParDisk =
  {
    ["16PFKD4QFSAA"] = "par0",
  },
  OSDisk =
  {
    ["D8E107781F8400012443"] = "os0",
  }
,
  ["NFSDisk"] =
  {
    ["192.168.1.100:/home"] = "remote0",
  }
,
  IndexDisk = "ssd",
  CacheLevel = "full",
  RaidLevel = "raid0",
}
```

# Mount Remote CIFS (Windows) Drive

## Requires FW:7256+

Mounting a remote CIFS/Windows file system on the FMADIO capture device can be extremely useful in many cases. It allows end users to use familiar Windows file browser to access and interface with PCAPs and FMADIO systems.

The main usage for remote CIFS mounts is to provide a simple and robust way to push PCAP data automatically from the capture device to a Windows Share drive. Once on a Windows Share drive it is easily accessible from multiple devices.

## CIFS Remote Mount Config

Please edit the configuration file in

```
/opt/fmadio/etc/disk.lua
```

Create an entry named "CIFSDisk" as shown below, be careful all punctuation is correct. The mount point should be "remote0" or "remote1" or "remote2" which maps to /mnt/remote0 /mnt/remote1 on the capture device.

```
["CIFSDisk"] =  
{  
    ["//192.168.1.100/share"] = "remote0:username=user,password=secret",  
},
```

Mount options for username and password are specified after a double colon :

After the configuration file has been updated, run the following command to confirm the config file has no syntax errors.

```
fmadio@fmadio20v3-287:/mnt/store0/etc$ fmadiolua /opt/fmadio/etc/disk.lua  
failed to open self? [fmadiolua]  
loading filename [/opt/fmadio/etc/disk.lua]  
done 0.000065Sec 0.000001Min  
fmadio@fmadio20v3-287:/mnt/store0/etc$
```

Above is the correct operation. Once complete please reboot the system to confirm the new mount points.

To confirm the mount point please run `ls -al /mnt/remote*` and check the file/directory contents

```
fmadio@fmadio20v3-287:/mnt/store0/etc$ ls -al /mnt/remote0
total 1024
drwxr-xr-x   3 nobody  nogroup          0 Jul  4 01:52 ./
drwxr-xr-x  11 root     root            240 Jul  4 02:43 ../

drwxr-xr-x   2 nobody  nogroup          0 Jul  4 01:52 test1/
-rw-r--r--   1 root     root            27 Jul  4 01:52 example
fmadio@fmadio20v3-287:/mnt/store0/etc$
```

# Capture Pipeline Flush

FMADIO Packet Capture systems like all capture systems have multiple internal buffers. These internal buffers can sometimes cause problems for low bandwidth connections which requires Packets to be available on disk immediately for downstream processing.

One such example is Financial Order and Entry data, which can sometimes be extremely low bandwidth however downstream systems require packets to be available ASAP for further processing.

FMADIO Gen2 systems buffer between 2MB-4MB of data internally. To support multiple use cases the flushing mechanics can be tuned based on the customers requirements. By default the flushing occurs when there is no new packets in the last 1 second.

## Configuration

Please edit the configuration file in

```
/opt/fmadio/etc/time.lua
```

The relevant sections are (there may be more or less entries in the ["Capture"] config block)

```
["Capture"] =  
{  
  
    ["FlushPktCnt"] = 2000,  
    ["FlushPeriod"] = 0,  
    ["FlushIdle"]   = 1e9,  
  
}
```

If these options are not visible in the config file, please go to the GUI Config page, change the PCAP Time Resolution to Micro Second, then back to Nano Second. This will write the default values into the config file. Alternatively you can paste the missing lines from the above example.

NOTE: After changing the settings capture must be stopped, and restarted for the new settings to take effect

## Settings

Description of what each field setting does.

### FLUSHPKTCNT



Flushing works by injecting specially marked NOP packets into the system right at the capture port. Its as if the packets arrived on the ingress port, but are never visible or downloadable. This parameter sets the number of packets for each flush per port to be injected. The packets are 256B in length.

Default value is: 2000 pkts \* 256B = 512,000 bytes per port.

For usage models where quick Flush to disk is critical, its recommended to use 5,000 or 10,000 packets for a complete flush. Note this will directly effect how much storage is consumed by the flushing behavior

## **FLUSHPERIOD**

Flushing based on a pre-defined time interval. For example flush the entire pipeline every 1 minute regardless of how much data has been seen. For a 1 minute flush, the value here should be 60e9, scientific notation is accepted and the unit of time is nano seconds.

Default value is: 0 - this disables the periodic flushing

Lowest recommended setting is 1 minute, otherwise excessive flushing will consume disk space.

## **FLUSHIDLE**

Flushing based on an in-activity idle timeout. This will flush the pipeline if no new packets are received within X amount of time. For example the default setting is 1 second, if no new packets are received after 1 second a SINGLE pipeline flush is issued. The next pipeline flush will only occur if new packets are received.

Default value is: 1e9 - flush after 1 second of inactivity, value in nanoseconds To disable set to 0

This mode is the default configuration

## **Recommended (Aggressive Flushing)**

For Financial and other customers who require a constant flush to disk, the following setting is recommended

```
["FlushPktCnt"] = 5000,  
["FlushPeriod"] = 60e9,  
["FlushIdle"]   = 0,
```

This will flush both ports every 1 minute continuously.

1 Hour / 1 Min = 60 flushes

1 Flush 2 x 5000 packets \* 256 Bytes = 2,560,000 Bytes per flush

Total of extra 153MB per hour for the continuous flushing. or 1.2GB for 8 Hours is fairly reasonable.

## **Default (Idle Flushing)**

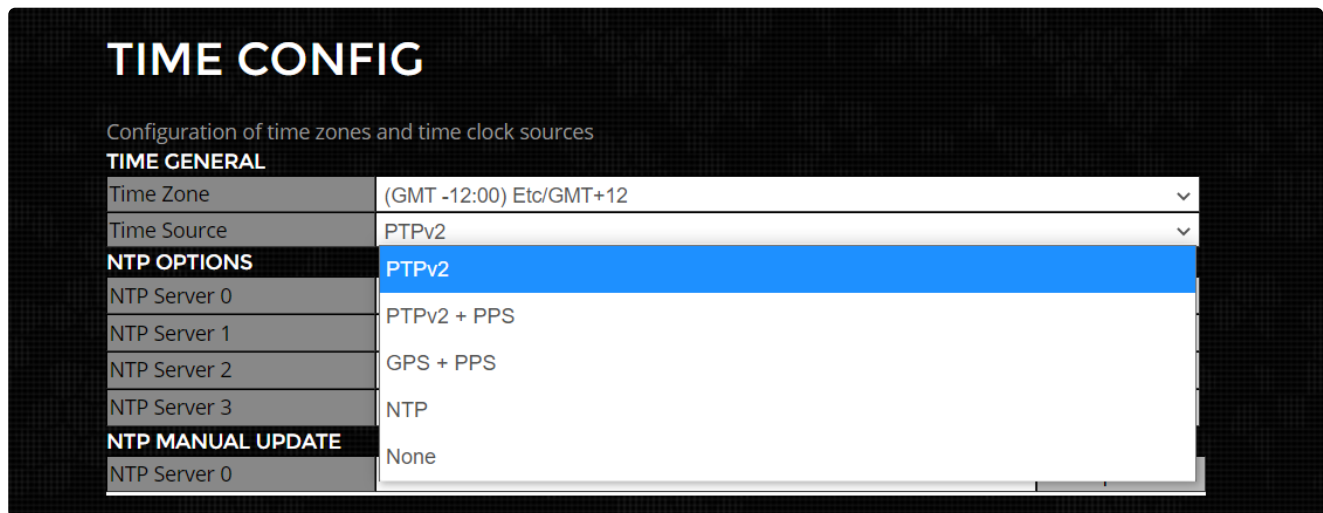
Default system behavior is flushing when capture is idle for second (1e9 nanoseconds).

```
["FlushPktCnt"] = 2000,  
["FlushPeriod"] = 0,  
["FlushIdle"]   = 1e9,
```

# Global Time PTPv2

FMADIO capture systems support PTPv2 using the linux PTP open source project. (<http://linuxptp.sourceforge.net/>) running only on the 10G management interface. 1G links are supported using 1G RJ45 SFP transceiver.

Enabling PTPv2 via GUI is as follows



PTPv2 Time Selection

In addition the 10G Bridged networking config must be disabled. As LinuxPTP must run directly on the bare metal NIC. Please edit the file

```
/opt/fmadio/etc/60-persistent-ethernet.rules
```

Change the "phy10" values as shown below to "man10"

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:b4:0e:6c", KERNEL=="eth*", NAME="phy0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:b4:0e:6d", KERNEL=="eth*", NAME="phy1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:17:7a:4e", KERNEL=="eth*", NAME="phy10"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:17:7a:4f", KERNEL=="eth*", NAME="phy11"
```

Before Bare Metal NIC

Changed to "man10" as follows

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:b4:0e:6c", KERNEL=="eth*", NAME="phy0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:b4:0e:6d", KERNEL=="eth*", NAME="phy1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:17:7a:4e", KERNEL=="eth*", NAME="man10"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*"; ATTR{address}=="18:c0:4d:17:7a:4f", KERNEL=="eth*", NAME="phy11"
```

Enable Bare metal 10G Interface

System must be rebooted for the above to take effect.

## PTPv2 VLAN Interface

In some environments PTPv2 is run on its own dedicated VLAN interface. To enable this update the configuration file

```
/opt/fmadio/etc/time.lua
```

Specifically setting the VLANID as follows, in the below example its set to VLANID "123"

```
["PTP"] =  
{  
  ["Master0"] = "",  
  ["Master1"] = "",  
  ["Master2"] = "",  
  ["Master3"] = "",  
  ["UpdateRate"] = "",  
  ["VLANID"] = "123",  
  ["Interface"] = "man10",  
},
```

After editing save and confirm the file syntax is correct, ensure there are no syntax or parse errors. The correct output is shown below.

```
fmadio@fmadio20n40v3-364:~$ fmadlua /opt/fmadio/etc/time.lua  
fmad fmadlua Aug 10 2021  
failed to open self? [fmadlua]  
loading filename [/opt/fmadio/etc/time.lua]  
done 0.000051Sec 0.000001Min  
fmadio@fmadio20n40v3-364:~$
```

Either reboot the system, or kill all ptp processes (e.g. `sudo killall ptp4l; sudo killall phc2sys`)

On reboot a man10.123 VLAN interface that matches the set VLANID should be populated as follows

```
fmadio@fmadio20n40v3-364:~$ ifconfig man10.123  
man10.123 Link encap:Ethernet HWaddr 18:C0:4D:17:7A:4E  
BROADCAST MULTICAST MTU:9200 Metric:1  
RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)  
  
fmadio@fmadio20n40v3-364:~$
```

And the linux ptp daemo should have binded to the newly created instance as follows

```
fmadio@fmadio20n40v3-364:~$ ps aux |grep ptp4l
64727 root      /opt/fmadio/bin/ptp4l -4 -H -i man10.123 -m -s
fmadio@fmadio20n40v3-364:~$
```

And the PHC interface also

```
fmadio@fmadio20n40v3-364:~$ ps aux |grep phc2sys
64729 root      /opt/fmadio/bin/phc2sys -s man10.123 -c CLOCK_REALTIME -w -m -P 0.7 -I 0.3
fmadio@fmadio20n40v3-364:~$
```

## PTPv2 on Second 10G Port

The above config assumes PTPv2 is running on the primary 10G management port. There is not requirement for using the primary port, instead the 2nd 10G management port can also be used. Below are the differences required

### 1) in the udev rules phy10 -> phy11 -> man11

Use phy11 interface

```
fmadio@fmadio100v2-228U:/mnt/store0/development_20230914-1-C33-V1-system/fmadio100v2-228U:/mnt/store0/etc$ cat 60-persistent-ethernet.rules
# dev 228
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="e0:d5:5e:5d:21:51", KERNEL=="eth*", NAME="phy1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="e0:d5:5e:5d:21:50", KERNEL=="eth*", NAME="phy0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="f8:f2:1e:69:bb:e0", KERNEL=="eth*", NAME="phy10"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="f8:f2:1e:69:bb:e1", KERNEL=="eth*", NAME="phy11"
```

Rename to man11

```
fmadio@fmadio100v2-228U:/mnt/store0/etc$ sudo vi 60-persistent-ethernet.rules
fmadio@fmadio100v2-228U:/mnt/store0/etc$ cat 60-persistent-ethernet.rules
# dev 228
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="e0:d5:5e:5d:21:51", KERNEL=="eth*", NAME="phy1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="e0:d5:5e:5d:21:50", KERNEL=="eth*", NAME="phy0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="f8:f2:1e:69:bb:e0", KERNEL=="eth*", NAME="phy10"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="f8:f2:1e:69:bb:e1", KERNEL=="eth*", NAME="man11"
```

### 2) Set the IP address of man11

man11 needs to be configured with an IP address, either editing network config file directly

```
/opt/fmadio/etc/network.lua
```

or using the fmadiocli command line utility

### 3) Specify man11 in the ptp config per below

```
["PTP"] =  
  
{  
  ["Master0"] = "",  
  ["Master1"] = "",  
  ["Master2"] = "",  
  ["Master3"] = "",  
  ["UpdateRate"] = "",  
  ["VLANID"] = nil,  
  ["Interface"] = "man11",  
},
```

## PTP on the Second 1G RJ45 Port

PTPv2 can be run on the 2nd 1G management interface, as it supports both a PHY clock and IEEE1588 hardware timestamping.

**NOTE: PTPv2 can not be run on the primary/first 1G interface as this is a software bridged interface and does not support hardware timestamping.**

### 1) Configure the udev rules as follows

```
/opt/fmadio/etc/60-persistent-ethernet.rules
```

Set the phy1 interface renamed to man1 as shown below

```
# dev 228  
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="e0:d5:5e:5d:21:51", KERNEL=="eth*", NAME="phy0"  
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="e0:d5:5e:5d:21:50", KERNEL=="eth*", NAME="man1"  
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="f8:f2:1e:69:bb:e0", KERNEL=="eth*", NAME="man10"  
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="f8:f2:1e:69:bb:e1", KERNEL=="eth*", NAME="phy11"
```

### 2) Set the IP address of man1

man1 needs to be configured with an IP address that the PTP master can reach. Set the network information either editing network config file directly

```
/opt/fmadio/etc/network.lua
```

or using the fmadiocli command line utility

### 3) Specify man1 in the ptp config per below

```
["PTP"] =
{
  ["Master0"] = "",
  ["Master1"] = "",
  ["Master2"] = "",
  ["Master3"] = "",
  ["UpdateRate"] = "",
  ["VLANID"] = nil,
  ["Interface"] = "man1",
},
```

## Custom LinuxPTP Config

FMADIO system use the standard LinuxPTP project for time synchronization

The Linux PTP Project

Linux PTP Configuration

As such the PTP configuration is highly customizable. For example Broadcast environments use a non standard domain and update rate. FMADIO support this by using a custom LinuxPTP configuration file.

Custom config file can be located below

```
/opt/fmadio/etc/ptp41.conf
```

If this file is present the system will use it when starting the application ptp4l.

After creating or editing the file, need to stop ptp4l as follows

```
// Some code
sudo killall ptp4l
sudo killall phc2sys
```

The system will then automatically start it on a 60sec timer.

Logfiles for debugging can be found in

```
// Some code
/mnt/store0/log/ptp_ptp41.log
/mnt/store0/log/ptp_phc2sys.log
```

## PPS Time Synchronization

FMADIO Devices have a PPS input connect that expect 1 1PPS 50ohm 5V signal. The start of second boundary occurs on the rising edge of the PPS clock. Required hold time is 100usec or more.

Enabling of PPS is automatic, when the system detects PPS input, it automatically selects PPS to discipline the timestamp clock.



# Timestamp Accuracy

FMADIO devices timestamp all incoming packets on the **first byte of payload data** received at the PCS layer. The granularity of the timestamp depends on the port speed which selects the PCS clock frequency. The table below shows the granularity of the timestamp based on Port Speed

Port Speed	Timestamp Granularity (Nanoseconds)
2x100G	3.103ns
2x40G	3.2ns
8x10G	6.4ns

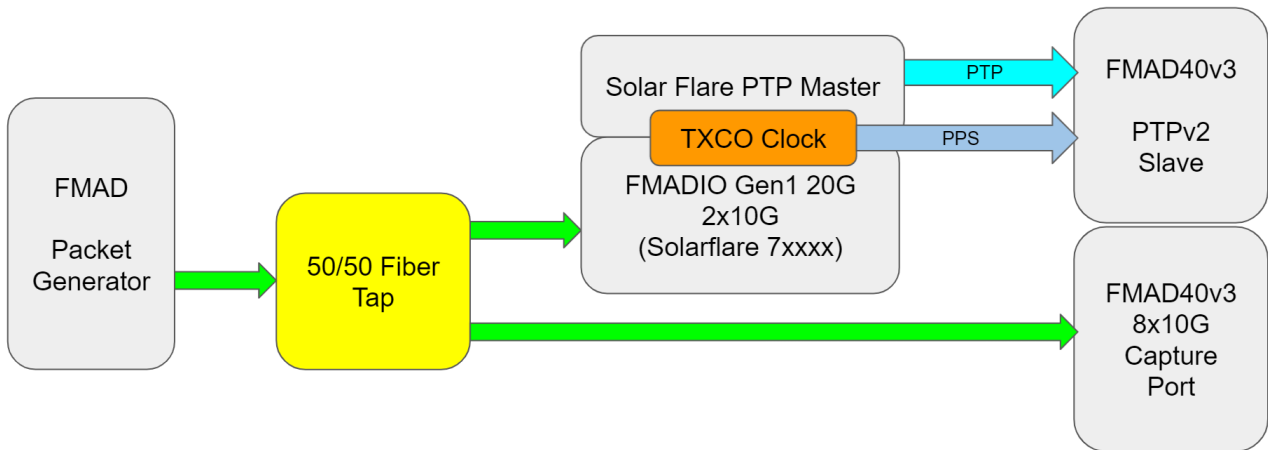
Timestamps are always written in nanoseconds, which requires the system to constantly discipline the FPGA internal clock. This provides highly accurate timestamps against global world time down to the nanosecond level.

Below is a summary on the accuracy that can be achieved using FMADIO packet capture systems

Time Synchroniation	Global Time Accuracy
NTP	100 milliseconds
PTPv2	100 nanoseconds
Pulse Per Second	10 nanoseconds

## Verification Environment

The following is FMADIO internal test setup on how to measure the accuracy of our hardware timestamps.

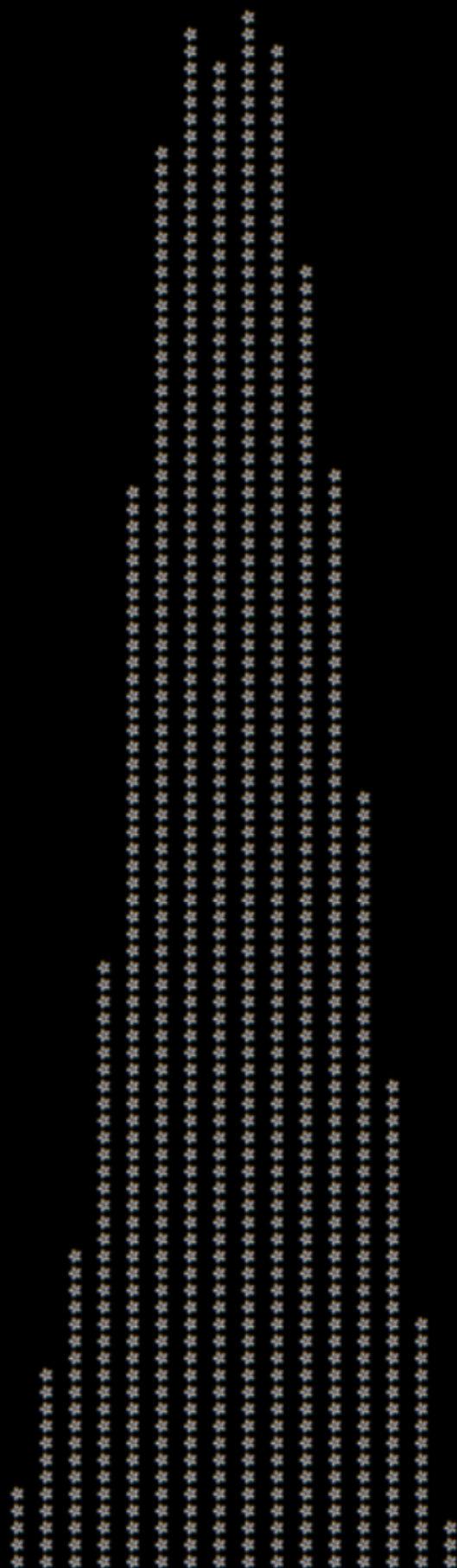


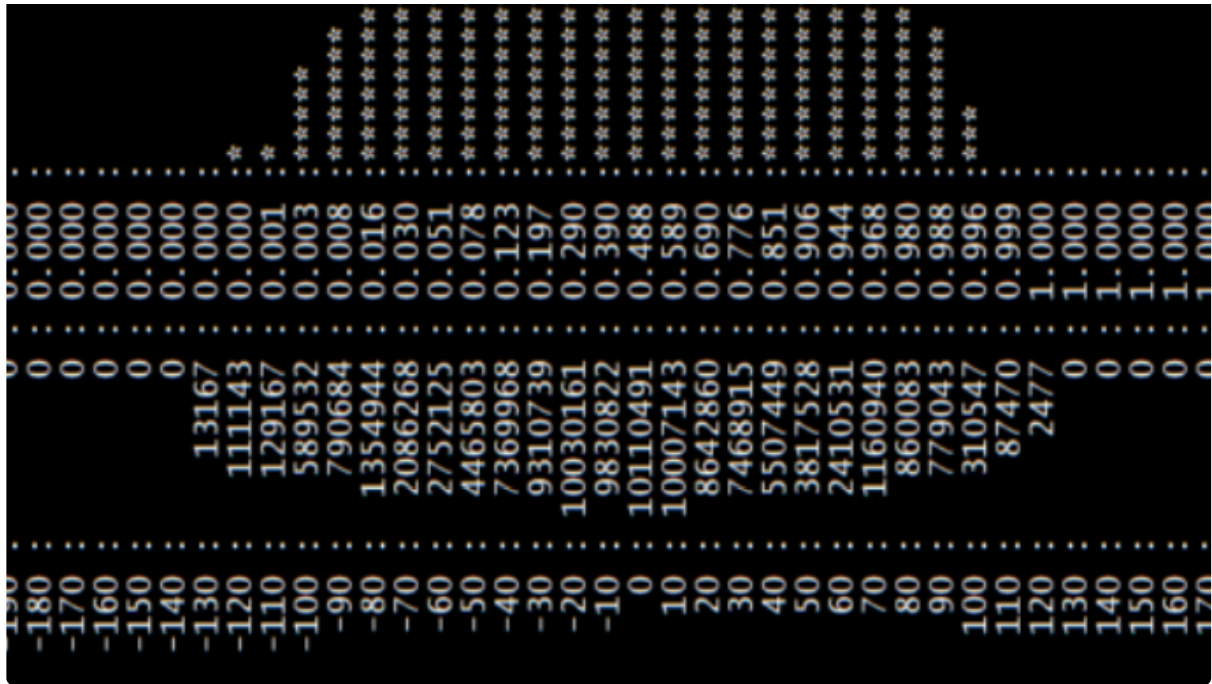
Time accuracy Testing setup

The picture above shows we are verifying the time accuracy of the system by comparing it against FMADIO Gen1 system that uses Solarflare NIC for its capture and timestamping. As the Solarflare also has a TXCO clock and running the full PTPv2 and PPS output master the Solarflare is timestamping packets directly with the TXCO master clock. This creates an excellent test.

### PTPv2 Global Time Accuracy

PTPv2 time accuracy of PTPv2 only is about +/- 100 nanoseconds. This is the measured results of the above testing setup, the resulting histogram below shows the time delta in nanoseconds between the FMADIO Gen1 (Solarflare master clock timestamp) and the FMADIO Gen3 (8x10G) timestamp as the slave clock.





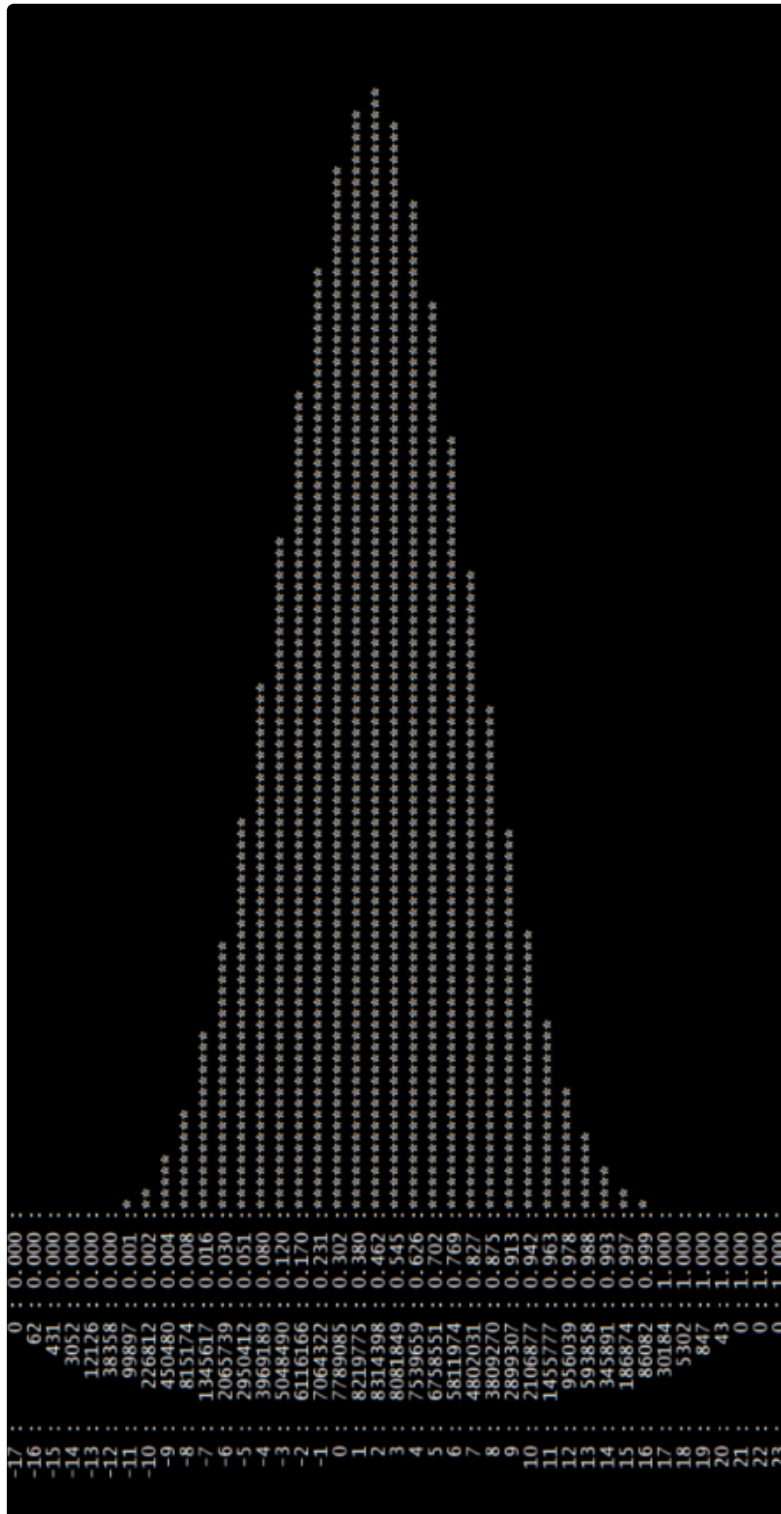
FMADIO PTPv2 Only Global Time Accuracy

Per the above time histogram, the support width is about +/- 100nsec this is the expectation of PTPv2 synchronized clocks. Its pretty good considering it requires no dedicated clock hardware, just time synchronization over 10G ethernet

### PPS Global Time Accuracy

For the ultimate global time accuracy, Pulse Per Second timing gives the most accurate clock synchronization results. This requires dedicated Clock hardware typically using Coax cable and SMA/BNC connectors. Typical setup is a PTPv2 Grandmaster synchronized using a GPS antenna, this Grandmaster has PTPv2 master to sync against but it also has PPS Coax output for slave clocks to achieve the absolute best global time accuracy.

Below is a time histogram of our FMADIO Gen1 (Solarflare NIC as the master clock) compared against FMADIO 40G Gen3 (8x10G) system as the slave clock. The clock are synchronized using PTPv2 and PPS coax cable. The result is impressive



PPS Global Time Accuracy (Nanoseconds)

In the above histogram the time bins are 1 nanosecond. As you can see the support width of the histogram is +/- 10ns. This means the global time accuracy is within 10 nanoseconds.

This the extreme level of clock synchronization as required by extreme customers. This is particle physics level accuracy as required by the ultimate most high end applications.

# Username Passwords

## WEBGUI

FMADIO Packet Capture Systems use the default login and password when the system is shipped. Additional WebGUI users can be added manually using the `htpasswd` utility. To set a new password "password" for the `fmadio` account use the following command line:

```
fmadio@fmadio20v2-149:$ sudo htpasswd /opt/fmadio/etc/htpasswd fmadio password
```

By default this utility overwrites the existing user account, so only 1 user account is possible.

## WEBGUI Multiple Accounts

Additional users are added by appending to the `/opt/fmadio/etc/htpasswd` file. The following shows creating a user account "test" with the password "newpassword".

```
fmadio@fmadio20v2-149:$ sudo htpasswd /tmp/ptmp test newpassword
fmadio@fmadio20v2-149:$ cat /tmp/ptmp >> /opt/fmadio/etc/htpasswd
```

Please be careful duplicate usernames are not in the `/opt/fmadio/etc/htpasswd` file. Use a text editor to adjust the file if needed.

The new users and passwords can now access the GUI. In addition for logging the `nginx` access logs will show the username for all URL requests.

## SSH

Unfortunately adding additional SSH usernames is not possible, as the permissions may be incorrectly set causing undefined system behavior. However multiple people can login to the system using different SSH keys via the `.authorized_keys` config file.

The authorized ssh keys file is located in

```
/opt/fmadio/etc/fmadio_authorized_keys
```

Please note, the `authorized_keys` file in the users `.ssh` account directory does not persist across reboots. Keys must be added to the above location.

## Change default SSH password

Changing the default SSH password uses the standard linux utilit "passwd".

```
sudo passwd fmadio
```

Example below

```
fmadio@fmadio20v3-287:/etc$ sudo passwd fmadio
Changing password for fmadio
Enter the new password (minimum of 5, maximum of 8 characters)
Please use a combination of upper and lower case letters and numbers.
New password:
Re-enter new password:
passwd: password changed.
fmadio@fmadio20v3-287:/etc$
```

After setting the new password, it needs to be copied to persistent storage

```
sudo cp /etc/shadow /opt/fmadio/etc/
```

# SSH Customization

FMADIO devices run exclusively from pseudo-ROM where any changes on the file system between reboots is lost. This ROM approach provides consistency and system predictability making maintenance simpler.

## Shell Environment

One problem with this approach is shell customization becomes quite difficult. To allow small modifications in the shell environment when a user logs into the system it can run the shell script for each SSH session. Configuration file is:

```
/mnt/store0/etc/fmadio.rc
```

Please do not use this excessively, typically its used for setting ENV variables.

Example:

```
$ cat /mnt/store0/etc/fmadio.rc
# local shell prompt configuration (ash) ran on at boot time

export TEST="random test variable"
```

## Persistent authorized\_keys

This file is usually located in `~/.ssh/` directory. As that is part of the volatile file system, the persistent version of this is placed into

```
/opt/fmadio/etc/authorized_keys
```

This allows SSH keys to be used in a persistent way across reboots and power cycles. Note the file in `/opt/fmadio/etc/authorized_keys` is only copied during bootup. Updates made after reboot are not copied to the user `.ssh` directory.

## Custom SSHD config

A customized sshd configuration file can be used by placing the customized configuration into

```
/opt/fmadio/etc/sshd_config
```



This is helpful for example to force exclusive RSA based login / disable password login. Which is a good practice if the device is on a public network.

## Custom SSH RSA ID

In many cases using the default fmadio SSH RSA ID is not a good security practice. As such custom SSH RSA keys both public and private can be copied into

```
/opt/fmadio/etc/fmadio_id_rsa  
/opt/fmadio/etc/fmadio_id_rsa.pub
```

These will copied into ~/.ssh/idrsa and idrsa\_pub on boot

## Persistent SSH Tunnel

FW: 7974+

The FMADIO Packet Capture device may not be conveniently accessible on a network. Ability to form persistent SSH tunnels both to and from the FMADIO Packet Capture Device is important.

We use autossh for this feature.

One typical example is pushing rsyslog traffic to a centralized location for ingest and processing. In this example we show how to configure such a tunnel that remains persistent across reboots.

1) Create a new RSA keys on the FMADIO device

Storing the key in /mnt/store0/etc/sshtunnel.id without any password

```

fmadio@fmadio40v3SM-455:/mnt/store0/etc$ ssh-keygen -t rsa -f /mnt/store0/etc/sshtunnel.id
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /mnt/store0/etc/sshtunnel.id.
Your public key has been saved in /mnt/store0/etc/sshtunnel.id.pub.
The key fingerprint is:
0b:9f:0e:eb:2f:00:1b:ca:01:b2:ff:17:2e:9f:97:2e fmadio@fmadio40v3SM-455
The key's randomart image is:
+--[ RSA 2048]-----+
|           |
|o          |
|o.         |
|..o        |
|.o.+ . S   |
|..o . .o o |
| . o..+.   |
|  o E=o    |
|   ==*+    |
+-----+
fmadio@fmadio40v3SM-455:/mnt/store0/etc$ ls -al sshtunnel.id*
-rw-----  1 fmadio  staff      1679 Jun 11 19:12 sshtunnel.id
-rw-r--r--  1 fmadio  staff       405 Jun 11 19:12 sshtunnel.id.pub
fmadio@fmadio40v3SM-455:/mnt/store0/etc$

```

2) Add the public key to the remote servers

Add the `sshtunnel.id.pub` key to the remote servers `authorized_keys`. This allows password-less ssh access to the remote server.

3) Test the connection by ssh to the remote server

To ensure it logs in correctly manually.

```
fmadio@fmadio40v3SM-455:~$ ssh -i /mnt/store0/etc/sshtunnel.id ubuntu@192.168.1.100
```

```
Are you sure you want to continue connecting (yes/no)? yes
```

```
Warning: Permanently added '192.168.1.100' (ECDSA) to the list of known hosts.
```

```
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-1008-aws aarch64)
```

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage
```

```
System information as of Sat Jun 11 10:16:26 UTC 2022
```

```
System load: 0.11376953125    Processes:           176
Usage of /:   98.1% of 7.59GB  Users logged in:    1
Memory usage: 89%
Swap usage:   23%
```

```
=> / is using 98.1% of 7.59GB
```

```
* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
```

```
https://ubuntu.com/aws/pro
```

```
5 updates can be applied immediately.
```

```
To see these additional updates run: apt list --upgradable
```

```
$ exit
```

```
logout
```

```
Connection to closed.
```

```
fmadio@fmadio40v3SM-455:~$
```

#### 4) Create an on boot file that uses autossh

This gets run automatically on system boot. Because autossh handles reconnects, no cronjob or monitoring is required.

Create the file

```
/opt/fmadio/etc/boot.lua
```

Paste the contents as follows. This is a standard lua script, in this case its running autossh with some command line arguments. Anything can be run in the script for boot time setup

```
-- setup autossh tunnel to a remote syslog server
local Cmd = "/usr/bin/autossh -f -M 0 -N -o StrictHostKeyChecking=false -i /mnt/store0/etc/
print(Cmd)
os.execute(Cmd)
```

## 5) Test the boot script

Can test the functionality of the boot script as follows. Correct output is similar to this

```
fmadio@fmadio40v3SM-455:~$ fmadiolua /mnt/store0/etc/boot.lua

fmad fmadlua Jun 11 2022 (fmadiolua /mnt/store0/etc/boot.lua )
calibrating...
0 : 2399987938          2.4000 cycles/nsec offset:0.012 Mhz
Cycles/Sec 2399987938.0000 Std:          0 cycle std( 0.00000000) Target:2.40 Ghz
failed to open self? [fmadiolua]
/usr/bin/autossh -f -M 0 -N -o StrictHostKeyChecking=false -i /mnt/store0/etc/sshtunnel.id
done 0.001740Sec 0.000029Min
fmadio@fmadio40v3SM-455:~$
```

## 6) Reboot the system

On reboot the boot.lua script is executed and a persistent ssh tunnel to the remote system is formed.

## Custom sudoers File

A customized persistent /etc/sudoers file can be created at the following location

```
/opt/fmadio/etc/sudoers
```

On boot the system will override the default /etc/sudoers file with the file created per above

```
/opt/fmadio/etc/sudoers
```

# WWW Customization

We use nginx as our WWW front end interface, enabling all the standard features and stability this provides.

## Custom TLS Certificates

Custom TLS/SSL WWW x509 certificates are supported. Steps as follows

### STEP 1)

Copy <you cert>.pem to /opt/fmadio/etc/fmadio\_cert.pem

Copy <you cert>.key to /opt/fmadio/etc/fmadio\_cert.key

### STEP 2)

Stop the current nginx process

```
sudo killall nginx
```

### STEP 3)

Wait for nginx to re-spawn (takes ~ 60 sec) and your certs should be used for all HTTPS/TLS communication.

Any problems check the log files in /mnt/store0/log/nginx\_error.log

# Authentication

In many environments different Authentication is required. By default FMADIO capture systems using built in BASIC authentication over HTTP. As this makes configuration and setup simple but is very weak security setting.

## Supported Authentication

- BAISC (insecure)
- HTTPS Only + BASIC
- RADIUS
- Active Directory (SSO via OAUTH 2.0)
- Google Cloud (SSO via OAUTH 2.0)
- Ping Identity Cloud (SSO via OAUTH 2.0)

## HTTPS Only

By default HTTP and HTTPS are enabled on the GUI. In any security setting HTTP needs to be disabled, as its an unsecure protocol. To disable HTTP edit the config file

## General Config

```
/opt/fmadio/etc/time.lua
```

Find the "Security" section as follows

```
["Security"] =  
{  
  ["HTTPAccess"] = "enable",  
  ["Auth"] = "BASIC",  
  ["ConfigAccess"] = "full",  
  ["GUIMode"] = "full",  
},
```

Change the "HTTPAccess" section from "enable" to false as follows

```
["HTTPAccess"] = false,
```

Save the file

## Restart Nginx

Then restart nginx as follows

```
sudo killall nginx
```

NGINX will restart automatically within 60 seconds with the updated configuration. Only HTTPS access is possible.

SSO configuration is more complicated, please contact [support@fmad.io](mailto:support@fmad.io) and we can walk you thru the setup personally

## RADIUS

### FW: 7563+

We support RADIUS authentication using the freeradius client. Configuration is as follow

### General Config

Edit the configuration file

```
/opt/fmadio/etc/time.lua
```

Find the "Security" section, example shown below

```
["Security"] =  
{  
  ["HTTPAccess"] = false,  
  ["Auth"] = "BASIC",  
  ["ConfigAccess"] = "full",  
  ["GUIMode"] = "full",  
  ["RADIUS_Secret"] = "testing123",  
  ["RADIUS_Host"] = "192.168.1.1",  
  ["RADIUS_Protocol"] = "PPP",  
  ["RADIUS_Timeout"] = 864000000000000,  
},
```

### Disable HTTP Access

Change the following, this disabled the HTTP protocol

```
["HTTPAccess"] = false,
```

Changes the following, this enables RADIUS as the authentication method

```
["Auth"] = "RADIUS",
```

Configure your RADIUS login information

```
["RADIUS_Secret"] = "testing123",  
["RADIUS_Host"] = "192.168.1.1",  
["RADIUS_Protocol"] = "PPP",
```

Finally the Timeout, this is how long the system waits until it will automatically logout the user and requirement them to re-authenticate. Value is in nanoseconds, scientific notation and formula is no problem. Per below, 24 hours \* 60 min \* 60 sec \* 1e9 (nanos)

```
["RADIUS_Timeout"] = 24*60*60*1e9,
```

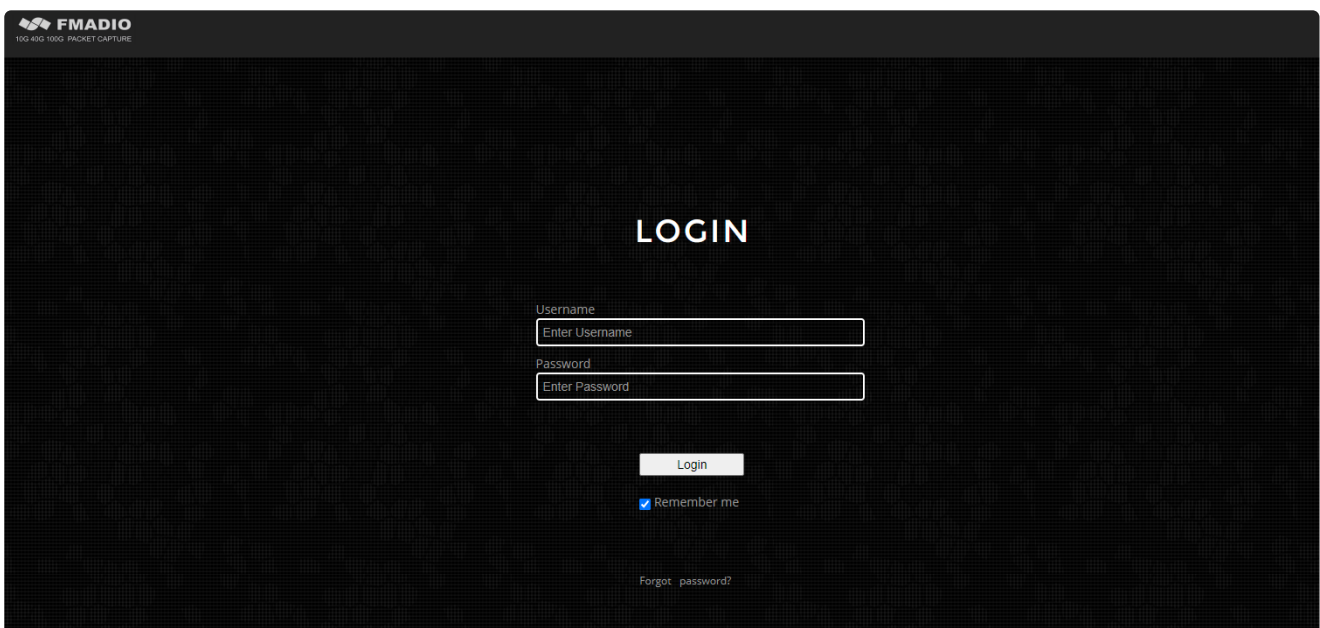
## Restart Nginx

Restart nginx as follows, it will re-spawn within 60sec automatically

```
sudo killall nginx
```

## Login

You should see a login page when accessing FMADIO as follows



FMADIO  
100 400 1000 PACKET CAPTURE

# LOGIN

Username  
Enter Username

Password  
Enter Password

Login

Remember me

[Forgot password?](#)

## TROUBLESHOOTING

If there is some problems, please confirm on CLI using radclient, example as follows.



```
fmadio@fmadio100v2-228U:$ echo "User-Name = steve" | radclient <radius server ip>:1812 auth
Sent Access-Request Id 95 from 0.0.0.0:56527 to 192.168.2.132:1812 length 27
Received Access-Reject Id 95 from 192.168.2.132:1812 to 192.168.2.175:56527 length 20
(0) -: Expected Access-Accept got Access-Reject
fmadio@fmadio100v2-228U:$
```

## Active Directory (SSO via OAUTH 2.0)

### FW:7608+

FMADIO Capture devices can authenticate the users using Active Directory via the OAUTH 2.0 protocol. This enable Single Sign On with ADFS.

### Public IP Testing

In the follow example we have used a reverse SSH tunnel to temporarily put FMADIO system on a public IP, as Azure Active Directory services require internet accessible devices for the redirect\_uri.

For an On Premise Active Directory server this is not required.

Example Reverse SSH Tunnel

```
ssh -R 8888:192.168.1.100:443 ec2-user@aws-instance.compute.amazonaws.com
```

NOTE: SSH tunnel should not use localhost, as all localhost sourced requests bypass authentication. Instead use the IP address of the management interface

### General Config

Start by editing the general FMADIO configuration file

```
/opt/fmadio/etc/time.lua
```

Then setting HTTP (un-encrypted) access to "disable", and Auth method to "OAUTH", example shown below. The other security fields can be left as is.

```
["Security"] =
{
    ["HTTPAccess"]    = "disable",
    ["Auth"]          = "OAUTH",
    .
    .
}
```

Save the file and ensure there are no parse errors by running `fmadiolua /opt/fmadio/etc/time.lua`

## OAUTH Config

Next create a file name

```
/opt/fmadio/etc/oauth_opts.lua
```

This file contains the ADFS OAUTH End points as follows

```
local config =
{
  redirect_uri      = "https://fmadio100v2-ip-address:8888/secure/",
  discovery         = "https://login.microsoftonline.com/571b0fe2-75cb-48de-9144-d41c59e7-6906-4569-9cc0-c6762541d2cd",
  client_id        = "d41c59e7-6906-4569-9cc0-c6762541d2cd",
  client_secret    = "fSY7Q~dkbG~mHLJYipKiC0XCmhnXQbOkOP5iE",
  ssl_verify       = "no",
  scope            = "openid email profile",
  redirect_uri_scheme = "https",
}

return config
```

These fields are from the ADFS Endpoint URI information, for example as follows. We created a fmadio sign in entry, this has the following client\_id entered above.

The "discovery" config in the above needs to be the OpenID Connect Metadata document, as seen below.

The screenshot shows the Azure AD portal interface for the 'fmadio signin' application. The 'Endpoints' section is open, displaying a list of endpoints. The 'OpenID Connect metadata document' endpoint is highlighted with a red box. The endpoint URL is <https://login.microsoftonline.com/571b0fe2-75cb-48de-9144-0cb928e90751/v2.0/well-known/openid-configuration>. Other endpoints listed include OAuth 2.0 authorization endpoints, OAuth 2.0 token endpoints, Microsoft Graph API endpoint, Federation metadata document, WS-Federation sign-on endpoint, and SAML-P sign-on and sign-out endpoints.

the "client\_id" is the shown below

Home > Default Directory > fmadio signin

Search (Ctrl+/) Delete Endpoints Preview features

Overview Quickstart Integration assistant Manage Branding Authentication Certificates & secrets Token configuration API permissions Expose an API

Essentials

Display name : fmadio signin

Application (client) ID : d41c59e7-6906-4569-9cc0-c6762541d2cd

Object ID : b5ef284f-c866-4d7d-816a-9976b0cd036f

Directory (tenant) ID : 571b0fe2-75cb-48de-9144-0cb928e90751

Supported account types : My organization only

Client credentials : 0 certificate, 1 secret

Redirect URIs : 1 web, 0 spa, 1 public client

Application ID URI : Add an Application ID URI

Managed application in L... : fmadio signin

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide feature Authentication Library (MSAL) and Microsoft Graph. Learn more

Get Started Documentation

The "client\_secret" in the above config needs to be the Value shown below, not the secretID

fmadio signin | Certificates & secrets

Search (Ctrl+/) Got feedback?

Overview Quickstart Integration assistant Manage Branding Authentication Certificates & secrets Token configuration API permissions Expose an API App roles Owners Roles and administrators | Preview Manifest

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) Client secrets (1) Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value	Secret ID
client secret string	12/17/2023	fSY*****	65aba152-80fc-46af-85b1-22413daba09c

Finally the "redirect\_uri" needs to be registered as follows.

Microsoft Azure Upgrade Search resources, services, and docs (G+)

Home > fmadio signin

fmadio signin | Authentication

Search (Ctrl+/) Save Discard Got feedback?

Overview Quickstart Integration assistant Manage Branding Authentication Certificates & secrets Token configuration API permissions Expose an API App roles Owners Roles and administrators | Preview Manifest Support + Troubleshooting Troubleshooting New support request

Platform configurations

Depending on the platform or device this application is targeting, additional configuration may be required such as redirect URIs, specific authentication settings, or fields specific to the platform.

+ Add a platform

Web

Redirect URIs

The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating or signing out users. The redirect URI you send in the request to the login server should match one listed here. Also referred to as reply URLs. Learn more about Redirect URIs and their restrictions

This app has implicit grant settings enabled. If you are using any of these URIs in a SPA with MSAL.js 2.0, you should migrate URIs.

https://[redacted]:8888/secure

Add URI

Mobile and desktop applications

Redirect URIs

The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating users. The redirect URI you send in the request to the login server should match one listed here. Also referred to as reply URLs. Learn more about Redirect URIs and their restrictions

Once config is complete, please confirm no syntax errors by running

```
fmadioLua /opt/fmadio/etc/oauth_opts.lua
```

Correct output is as follows, if there are any syntax errors please correct.

```
fmadio@fmadio100v2-228U:/mnt/store0/etc$ fmadiolua /opt/fmadio/etc/oauth_opts.lua
fmad fmadlua Dec 18 2021
calibrating...
0 : 2095077292          2.0951 cycles/nsec offset:4.923 Mhz
Cycles/Sec 2095077292.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
argv fmadiolua
failed to open self? [fmadiolua]
loading filename [/opt/fmadio/etc/oauth_opts.lua]
done 0.000090Sec 0.000002Min
fmadio@fmadio100v2-228U:/mnt/store0/etc$ █
```

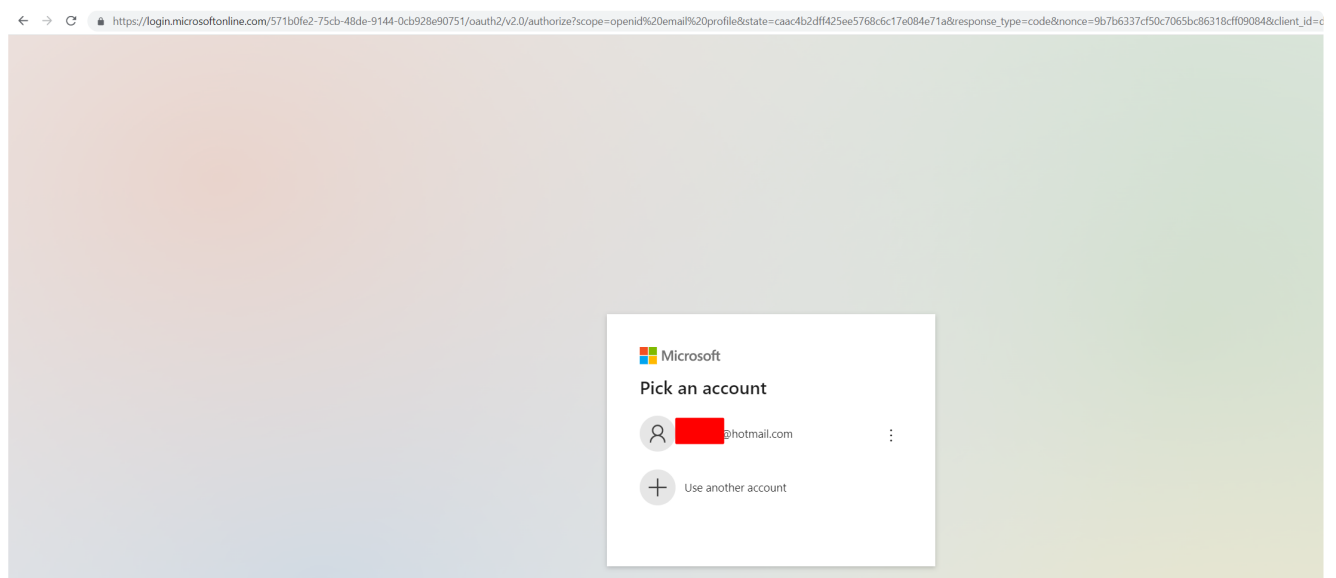
## Restart nginx

Restart nginx to load in the new configuration file, by killing the process as below. It will reswpan on a 1min cron job automatically

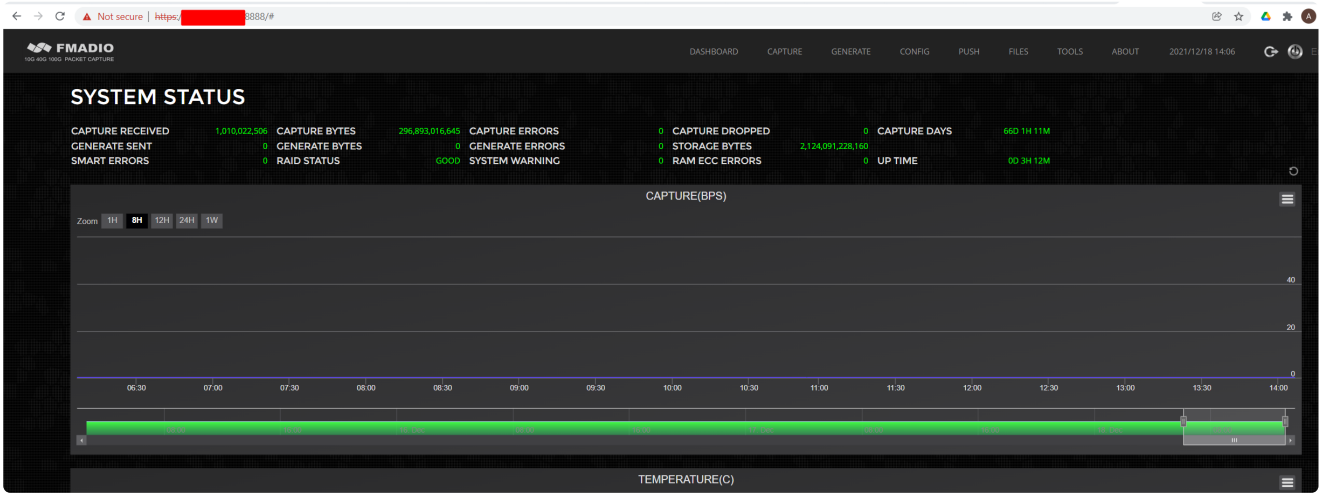
```
sudo killall nginx
```

## Logging in

Next point a browser to the FMADIO device, it should redirect you to the Active Directory login page as follows.

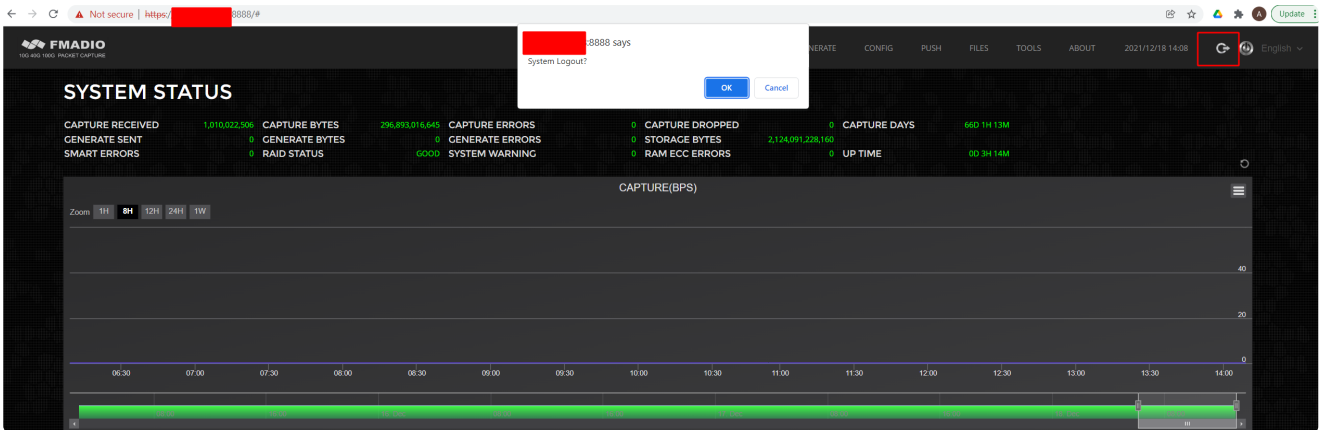


Login to the system using your Azure / Microsoft credentials. Then the FMADIO device dashboard will be shown as below

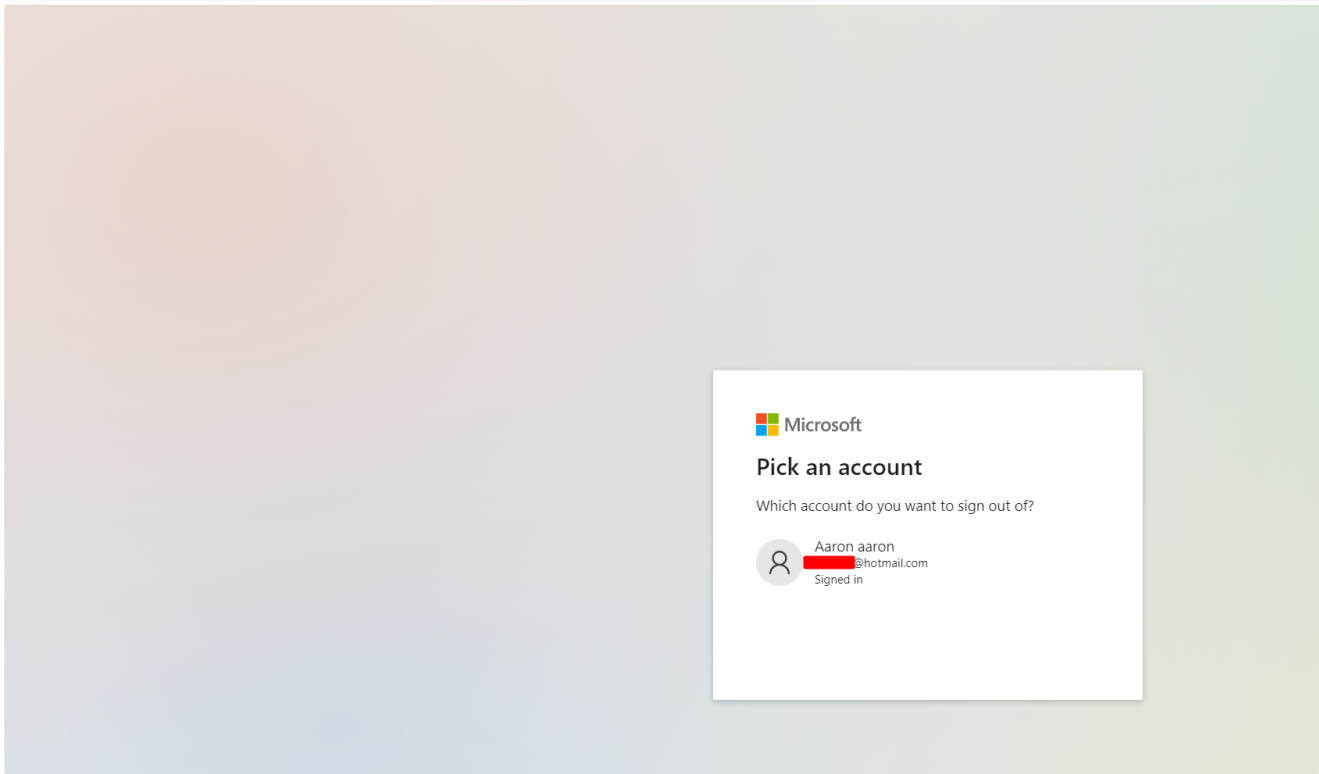


## Logout

Logout is the same, using the logout button shown below



Then choose an account to sign out of



## Google Cloud (SSO via OAUTH 2.0)

### FW:7608+

While less practical as its typically for publicly accessible sites, it can be used with a Google Cloud VPC to tunnel authentication requests from a private network to Google Cloud infrastructure.

In this example we just reverse ssh tunnel an FMADIO system onto the public internet (strongly discouraged) for demonstration purposes only.

### General Config

Start by editing the general FMADIO configuration file

```
/opt/fmadio/etc/time.lua
```

Then setting HTTP (un-encrypted) access to "disable", and Auth method to "OAUTH", example shown below. The other security fields can be left as is.

```
["Security"] =  
{  
  ["HTTPAccess"] = "disable",  
  ["Auth"] = "OAUTH",  
  .  
  .  
}
```

Save the file and ensure there are no parse errors by running `fmadiolua /opt/fmadio/etc/time.lua`

## OAUTH Config

Next create a file name

```
/opt/fmadio/etc/oauth_opts.lua
```

This file contains the Google Cloud OAUTH End points as follows.

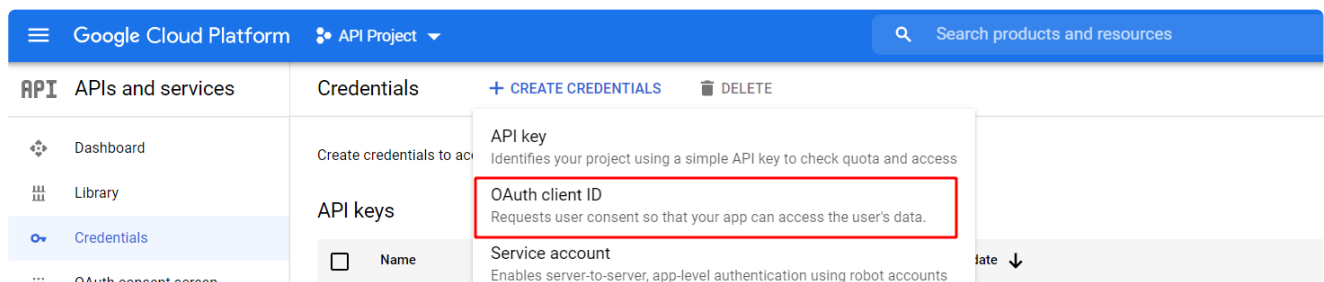
```
local config =
{
  redirect_uri          = "https://fmadio100v2-ip-address.com:8888/secure",
  discovery             = "https://accounts.google.com/.well-known/openid-configuration",
  client_id             = "431009152914-0jpm7fa0crh619gr48kv4c06h5ib19ou.apps.googleusercontent.com",
  client_secret        = "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",
  ssl_verify           = "no",
  scope                = "openid email profile",
  redirect_uri_scheme  = "https",
}

return config
```

The "clientid" and "client\_secret" need to be replaced with the generated authentication information from google per below. The above is a throw away example only

## Google Credentials

Next generate Google OAUTH credentials as follows.



The screenshot shows the Google Cloud Platform console interface. At the top, there is a blue header with 'Google Cloud Platform' and 'API Project' dropdown. Below the header, the 'APIs and services' section is active, and the 'Credentials' page is displayed. The page has a '+ CREATE CREDENTIALS' button and a 'DELETE' button. On the left, there is a navigation menu with 'Dashboard', 'Library', 'Credentials', and 'OAuth consent screen'. The main content area shows a list of credential types: 'API key' (Identifies your project using a simple API key to check quota and access), 'OAuth client ID' (Requests user consent so that your app can access the user's data.), and 'Service account' (Enables server-to-server, app-level authentication using robot accounts). The 'OAuth client ID' option is highlighted with a red rectangular box.

Then fill in the information, as follows. Google is a bit more strict and requires TLD endpoints not raw IPs

Google Cloud Platform API Project

API APIs and services

Dashboard  
Library  
**Credentials**  
OAuth consent screen  
Domain verification  
Page usage agreements

## ← Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

Application type \*  
Web application

Name \*  
fmadio sign in

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

**i** The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorised domains](#).

### Authorised JavaScript origins **?**

For use with requests from a browser

URIs \*

https://fmadio100v2-ip-address.com:8888

+ ADD URI

### Authorised redirect URIs **?**

For use with requests from a web server

URIs \*

https://fmadio100v2-ip-address.com:8888/secure

+ ADD URI

**CREATE** CANCEL

Which results in the following secret information

Google Cloud Platform API Project

API APIs and services

Dashboard  
Library  
**Credentials**  
OAuth consent screen  
Domain verification  
Page usage agreements

## ← Client ID for Web application

DOWNLOAD JSON RESET SECRET DELETE

Name \*  
fmadio sign in

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

**i** The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorised domains](#).

Client ID	431009152914-0jpm7fa0cm619gr48kv4c06h5ib9ou.apps.googleusercontent.com
Client secret	GOCSPX-p0-cd0knRW78cgHEDHcwxPVJa1
Creation date	19 December 2021 at 12:51:49 GMT+9

Update the oauth\_opts.lua file above with the information



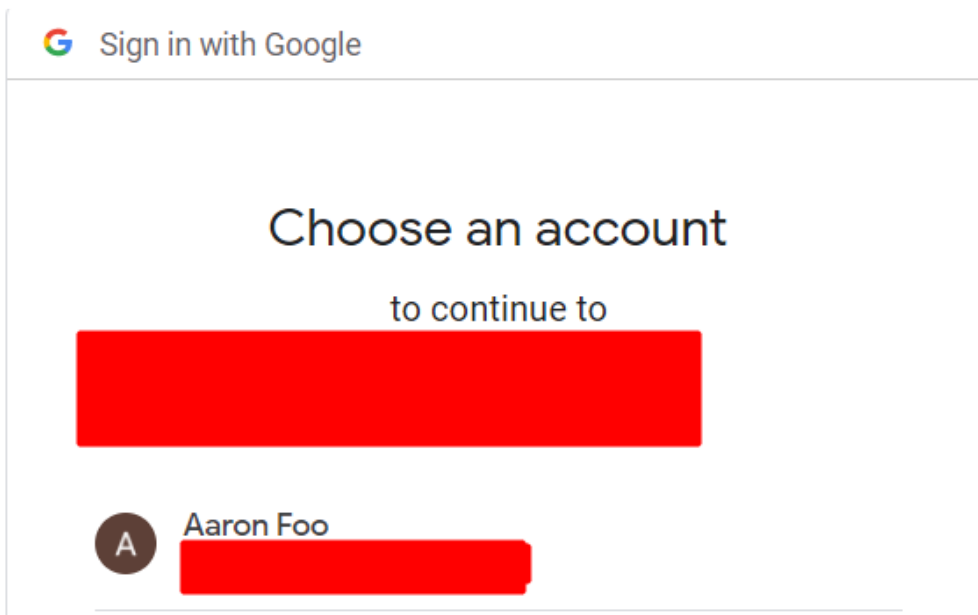
## Restart nginx

Restart nginx to load in the new configuration file, by killing the process as below. It will reswpan on a 1min cron job automatically

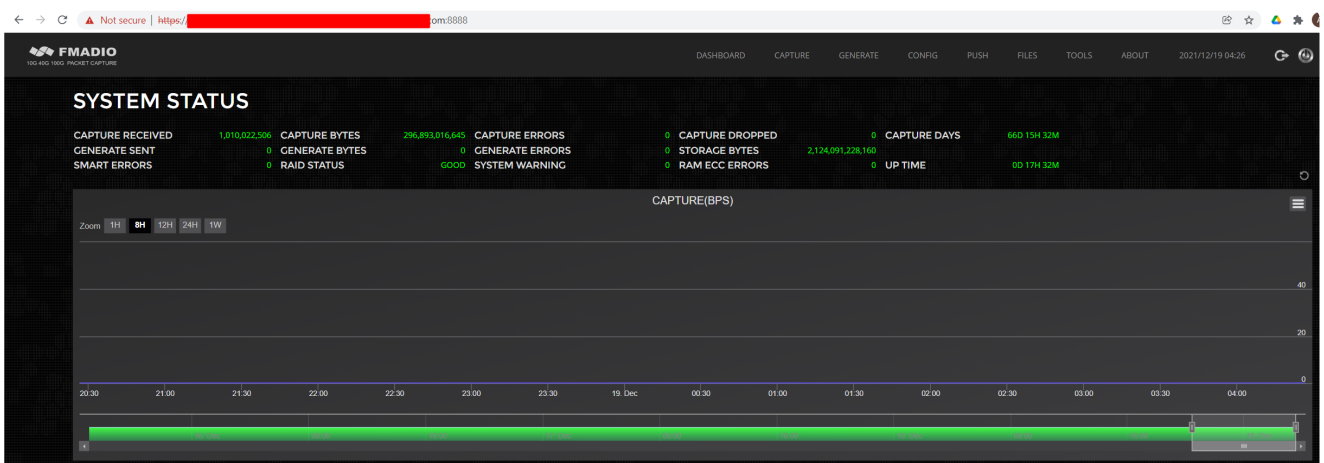
```
sudo killall nginx
```

## Logging In

Next point the browser to the FMADIO device and it will redirect to Google Sign in account



Login using your Google account information, and it will re-direct you to the FMADIO dashboard.



Any further questions please contact [support@fmad.io](mailto:support@fmad.io) for assistance.

## Ping Identity (SSO via OAUTH 2.0)

FW:7608+

Ping Identity is a popular onprem authentication system, typically used in large organizations. We support Single Sign On with their product suite, below is an example configuration example setup using the Cloud Services. This example uses a reverse SSH tunnel to put the FMADIO device on a publicly accessible IP (we strongly discourage) for demonstration purposes only, to replicate setting up an On Premise install.

## General Config

Start by editing the general FMADIO configuration file

```
/opt/fmadio/etc/time.lua
```

Then setting HTTP (un-encrypted) access to "disable", and Auth method to "OAUTH", example shown below. The other security fields can be left as is.

```
["Security"] =  
{  
  ["HTTPAccess"] = "disable",  
  ["Auth"] = "OAUTH",  
  .  
  .  
}
```

Save the file and ensure there are no parse errors by running `fmadiolua /opt/fmadio/etc/time.lua`

## OAUTH Config

Next create a file name

```
/opt/fmadio/etc/oauth_opts.lua
```

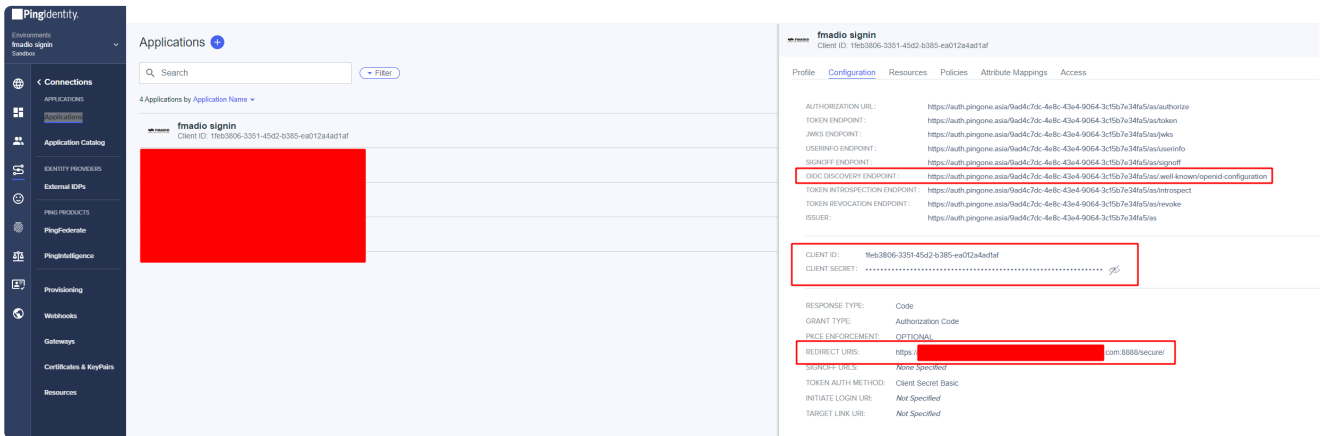
This file contains the Ping Identity OAUTH End points as follows.

```
local config =  
{  
  redirect_uri = "https://fmadio100v2-ip-address.com:8888/secure/",  
  discovery = "https://auth.pingone.asia/9ad4c7dc-4e8c-43e4-9064-3c15b7e",  
  client_id = "1feb3806-3351-45d2-b385-ea012a4ad1af",  
  client_secret = "3.j00-pU6TZ.g.MAReW53A2Dv11n~cdTbFINIiyMdkv_dUxH",  
  ssl_verify = "no",  
  scope = "openid email profile",  
  redirect_uri_scheme = "https",  
}  
  
return config
```

The "clientid" and "client\_secret" need to be replaced with the generated authentication information from Ping Identity interface per below. The above is a throw away example only

## Ping Identity Credentials

We setup a web application using Ping Identity interface as follows. The key fields are shown in red.



The screenshot shows the Ping Identity console interface. On the left is a navigation sidebar with categories like 'Connections', 'APPLICATIONS', 'IDENTITY PROVIDERS', 'PING PRODUCTS', 'PROVISIONING', 'WIDGETS', 'Gateways', 'Certificates & KeyPairs', and 'Resources'. The main area is titled 'Applications' and shows a list of applications. One application, 'fmadio signin' (Client ID: 1163806-3351-4562-b385-ea012a4ad1af), is selected. A large red rectangle obscures the application details in the list. To the right, the configuration page for this application is displayed. The 'Configuration' tab is active, showing various endpoints and settings. Several fields are highlighted with red boxes: 'SIGNOFF ENDPOINT', 'CLIENT ID', 'CLIENT SECRET', 'REDIRECT URL', and 'SECURITY URL'. The 'REDIRECT URL' is set to 'https://[redacted]com:8888/secure' and the 'SECURITY URL' is set to 'None Specified'.

Field	Value
AUTHORIZATION URL	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/authorize
TOKEN ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/token
AKMS ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/akms
USERINFO ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/userinfo
SIGNOFF ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/signoff
OPEN ID/VERIFY ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/well-known/openid-configuration
TOKEN INTROSPECTION ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/introspect
TOKEN REVOCATION ENDPOINT	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as/revoked
ISSUER	https://auth.pingone.asia/9ad4c7dc-4e8c-43a4-9064-3c15b7c34f5/as
CLIENT ID	1163806-3351-4562-b385-ea012a4ad1af
CLIENT SECRET	[Redacted]
RESPONSE TYPE	Code
GRANT TYPE	Authorization Code
PKCE ENFORCEMENT	OPTIONAL
REDIRECT URL	https://[Redacted]com:8888/secure
SECURITY URL	None Specified
TOKEN AUTH METHOD	Client Secret Basic
INITIATE LOGIN URL	Not Specified
TARGET LINK URL	Not Specified

These fields are mapped directly into the `oauth_opts.lua` configuration file above.

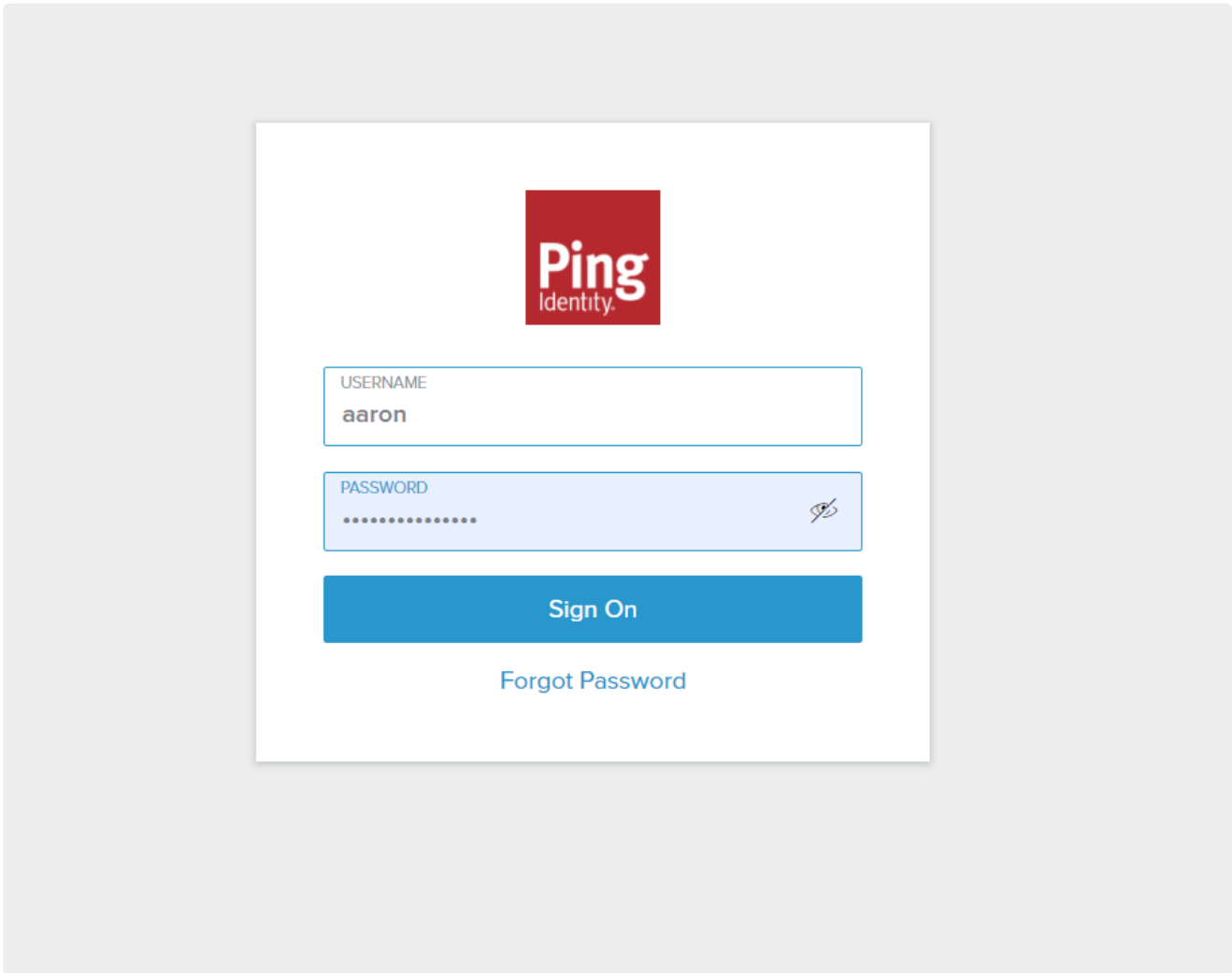
## Restart nginx

Restart nginx to load in the new configuration file, by killing the process as below. It will reswan on a 1min cron job automatically.

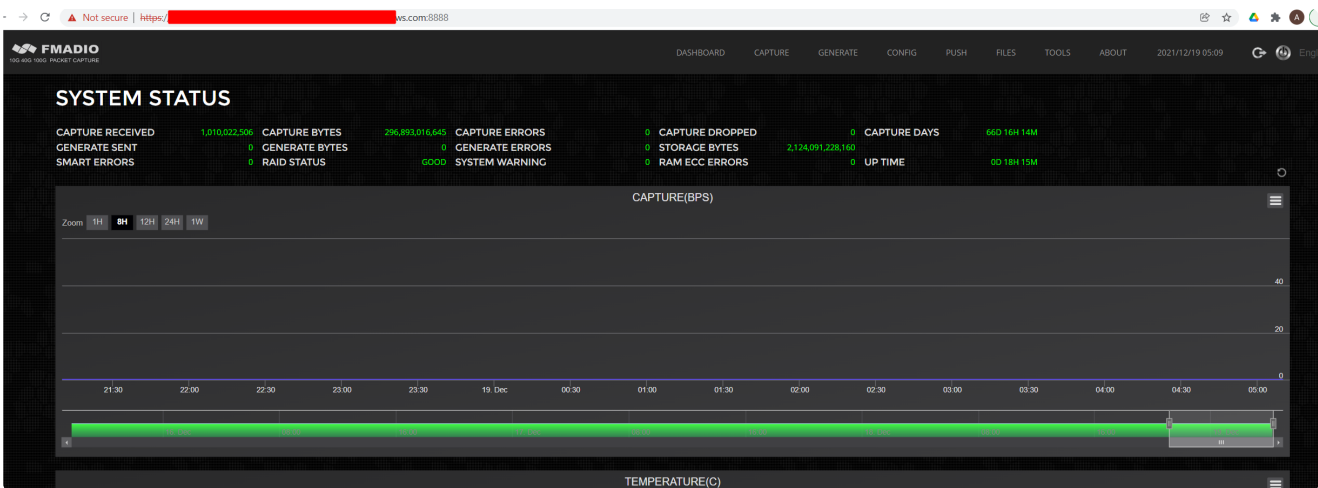
```
sudo killall nginx
```

## Logging In

Next point the browser to the FMADIO device and it will redirect to Ping Identity SSO account as follows



After a successful authentication the FMADIO dashboard is seen



Any further questions or problems, please contact us [support@fmad.io](mailto:support@fmad.io)

**PAM LDAP**

**FW: 8529+**

FMADIO systems support Linux PAM ( <https://github.com/linux-pam/linux-pam> ) as an authentication method. One option for centralized authentication is to use LDAP via PAM.

1) First run `fmadiocli` settings to set the authentication method

<https://docs.fmad.io/fmadio-documentation/cli-reference/fmadiocli#config-security-auth>

```
config security auth pam-ldap
```

2) We also strongly recommend to disable HTTP access as all username / passwords are sent over un-encrypted HTTP

<https://docs.fmad.io/fmadio-documentation/cli-reference/fmadiocli#config-security-http>

```
config security http false
```

3) Configure LDAP client `nslcd`. Copy the default config file as follows

```
cp /opt/fmadio/etc_ro/nslcd.conf /opt/fmadio/etc/nslcd.conf
```

The default config looks like the following

```
fmadio@fmadio100v2-228U:~$ cat /opt/fmadio/etc_ro/nslcd.conf
# /etc/nslcd.conf
# nslcd configuration file. See nslcd.conf(5)
# for details.
#log none debug

# The user and group nslcd should run as.
uid root
gid root

# The location at which the LDAP server(s) should be reachable.
uri ldap://192.168.1.100

# The search base that will be used for all queries.
base dc=fmad,dc=io

# The LDAP protocol version to use.
#ldap_version 3

# The DN to bind with for normal lookups.
#binddn cn=anonymous,dc=example,dc=net
#bindpw secret

# The DN used for password modifications by root.
#rootpwmoddn cn=admin,dc=example,dc=com

# SSL options
#ssl off

#tls_reqcert never
#tls_cacertfile /etc/ssl/certs/ca-certificates.crt

# The search scope.
#scope sub

#ssl start_tls
#tls_reqcert allow

fmadio@fmadio100v2-228U:~$
```

Modify the uri, base and any other LDAP specific configs to the enviroment and save it

4) reboot system

5) check LDAP connectivity

Changing the username/domain/ip address etc to match your environment

```
ldapwhoami -x -D cn=fmadio-user,dc=fmad,dc=io -H ldap://192.168.1.100/ -w "password"
```

Successful authentication looks like the following

```
fmadio@fmadio100v2-228U:~$ ldapwhoami -x -D cn=fmadio-user,dc=fmad,dc=io -H ldap://192.168
dn:cn=fmadio-user,dc=fmad,dc=io
```

Once this is working, both SSH, WWW-Admin and WWW-User LDAP posix group members can login to the system.

The LDAP posixGroups are

fmadio-ssh-admin - for SSH access

fmadio-www-admin - for WWW admin access (can change anything)

fmadio-www-user - for WWW user access (monitoring and pcap downloading)

6) Both SSH and WWW now fully configured using LDAP as centralized authentication

### **LDAP Optional**

Some environments require a notice when logging in, such as the following

# LOGIN

## AUTHORIZED SYSTEM ACCESS ONLY

This computer system is the property of The University of ABC. It is for authorized use only. By using this system, all users acknowledge notice of, and agree to comply with, the University's Acceptable Use of Information Technology Resources Policy. Users have no personal privacy rights in any materials they place, view, access, or transmit on this system. The University complies with state and federal law regarding certain legally protected confidential information, but makes no representation that any uses of this system will be private or confidential. Any or all uses of this system and all files on this system may be intercepted, monitored, recorded, copied, audited, inspected, and disclosed to authorized University and law enforcement personnel, as well as authorized individuals of other organizations. By using this system, the user consents to such interception, monitoring, recording, copying, auditing, inspection, and disclosure at the discretion of authorized University personnel. Unauthorized or improper use of this system may result in administrative disciplinary action, civil charges/criminal penalties, and/or other sanctions as set forth in the University's AUP. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. If you are physically located in the European Union, you may have additional rights per the GDPR. Visit the web site [dataprivacy.utk.edu](http://dataprivacy.utk.edu) for more information. ALL USERS SHALL LOG OFF A UNIVERSITY OF TENNESSEE OWNED SYSTEM IMMEDIATELY IF SAID USER DOES NOT AGREE TO THE CONDITIONS STATED ABOVE.

Username

Password



This can be customized as follows

1) copy the default template

```
cp /opt/fmadio/www/login/authorized_access.html.default /opt/fmadio/etc/authorized_access.ht
```

2) Edit the content of

```
/opt/fmadio/etc/authorized_access.html
```

3) restart nginx

kill nginx and wait 60sec for it to restart

```
sudo killall nginx
```



## Troubleshooting

Configuration usually does not go as planned, as such heres some tips to try

1) run nslcd in the foreground

```
sudo killall nslcd  
sudo /usr/local/sbin/nslcd -f
```

This will check the /etc/nslcd.conf configuration file is working correctly, either config typeo or LDAP server problems.

Once its running ensure local lookups work correctly as follows

```
ldapwhoami -x -D cn=fmadio-user,dc=fmad,dc=io -H ldap://192.168.1.100/ -w "password"
```

2) check nginx config files

The nginx logfiles are located in

```
tail -F -n 100 /mnt/store0/log/nginx_error.log
```

Any errors there might help understand the issues

3) check syslog file for PAM logs

```
tail -F -n 100 /mnt/store0/log/messages |grep -i pam
```

This will print out logs of all PAM messages and may help debugging

# Firewall IPTables

IPTables running on FMADIO Capture systems

## FW: 7650+

IPTables the linux statefull firewall software runs on the FMADIO devices. By default iptables is disabled / ACCEPT for everything. In some scenarios a tighter security policy may be needed

The iptables command works same on a standard Linux system. Please refer to the following link for documentation



iptables(8) - Linux man page

[IPTables documentation](#)

## Required Rules

FMADIO uses nginx and fcgi backends as well as proxy pass settings. As such IPTABLES requires an INPUT localhost ACCEPT rule such as the following.

```
3773 944K ACCEPT all -- * * 127.0.0.1 0.0.0.0/0
```

which can be added as follows

```
sudo iptables -I INPUT -s localhost -j ACCEPT
```

Without this INPUT ACCEPT rule the FMADIO GUI dashboards and status settings can not be retrieved.

## Persistent Configuration

After configuring the IPTABLES setup on the FMADIO Packet Capture device, the settings will be lost each time the system is rebooted.

We use iptables-save and iptables-restore command with the configuration file located

```
/opt/fmadio/etc/iptables.conf
```

To save the current state run

```
fmadio@fmadio20v3-287:/mnt/store0/etc$ sudo iptables-save > iptables.conf
fmadio@fmadio20v3-287:/mnt/store0/etc$
```

This will generate a the looks like the following. This is a baseline recommended setting for SSH, HTTP and HTTPS access only.

```
# Generated by iptables-save v1.6.1 on Wed Jan 19 19:09:33 2022
*mangle
:PREROUTING ACCEPT [10620:58020130]
:INPUT ACCEPT [9091:57235443]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [10194:1992682]
:POSTROUTING ACCEPT [10194:1992682]
COMMIT
# Completed on Wed Jan 19 19:09:33 2022
# Generated by iptables-save v1.6.1 on Wed Jan 19 19:09:33 2022
*nat
:PREROUTING ACCEPT [1547:785695]
:INPUT ACCEPT [10:528]
:OUTPUT ACCEPT [733:74107]
:POSTROUTING ACCEPT [733:74107]
-A PREROUTING -i eno2 -p tcp -m tcp --dport 7010 -j DNAT --to-destination 192.168.20.215:22
-A POSTROUTING -o eno2 -j MASQUERADE
COMMIT
# Completed on Wed Jan 19 19:09:33 2022
# Generated by iptables-save v1.6.1 on Wed Jan 19 19:09:33 2022
*filter
:INPUT DROP [98:13320]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [10194:1992682]
-A INPUT -s 127.0.0.1/32 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m state --state ESTABLISHED -j ACCEPT
COMMIT
# Completed on Wed Jan 19 19:09:33 2022
```

To remove persistent IPTABLES setting, delete the `/opt/fmadio/etc/iptables.conf` file and reboot the system

## IPMI BMC Firewall

The firewall on the IPMI/BMC is a bit trickier, as there's no direct access to iptables and manipulation needs to be done using a very unfriendly `ipmitool raw` access.

IPMI does have firewall manipulation GUI but its impossible to use due to how it works. Theres no way to set a Policy on INPUT rules, instead you need to drop everything and build up the chain. Below is the final iptables rules we want to create.

```

sysadmin [~]# iptables -L -n -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
0 0 ACCEPT udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:623
2789 416K ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
124 9146 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:22
3131 965K DROP all -- * * 0.0.0.0/0 0.0.0.0/0 source IP range 0.0.0.0-255.255.255.255
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:53 length 1025:65535
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:53 length 1025:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:53 length 512:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 length 512:65535
1 40 ZERO_WINDOW_RECENT all -- * * 0.0.0.0/0 0.0.0.0/0 u32 "0x6&0xff=0x6&&0x4&0x1fff=0x0&

```

ITarget IPMI Firewall Rules

As you can see its a bit bastardized.. but theres no choice as each rule is always added to the top of chain. Our goal is to DROP everything, except SSH, HTTPS and IPMITOOL traffic.

## 1) Reset IPMI BMC Firewall

Start by resetting the BMC firewall state entirely. This effectively resets iptables to the default state

```

fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x76 0x8
fmadio@fmadio100v2-228U:~$

```

And then confirm this by listing the total number of Firewall rules as follows

```

fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x77 0x0
00
fmadio@fmadio100v2-228U:~$

```

The value returned should be 00 indicating there are NO custom firewall rules.

**NOTE: This can be used to clear/reset firewall settings if a mistake is made**

## 2) Drop everything

Next we need to drop everything, as we are building the rules backwards. This is also the reason we cant use the GUI. It wont let you set a network of 0.0.0.0/0 and once you set that the GUI is no longer accessible.

As such we need to use ipmitool on the FMADIO Packet Capture device directly as we build up the rules.

```

fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x76 0x01 0x00 0x00 0x00 0x00 0x00 0xff 0x
fmadio@fmadio100v2-228U:~$

```

The command above adds the drop everything rule to the system, this equates to the following in iptables.

```

chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
0 0 ACCEPT udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:623
2789 416K ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
124 9146 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:22
3131 965K DROP all -- * * 0.0.0.0/0 0.0.0.0/0 source IP range 0.0.0.0-255.255.255
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:53 length 1025:65535
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:53 length 1025:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:53 length 512:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 length 512:65535
1 40 ZERO_WINDOW_RECENT all -- * * 0.0.0.0/0 0.0.0.0/0 u32 "0x6&0xff=0x6&&0x4&0x1fff=0x0&&0x0>>0

```

IPMI FW Drop everything

Can confirm its working correctly by checking the total number of firewall rules as follows. The returned value should be 1

```

fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x77 0x0
01
fmadio@fmadio100v2-228U:~$

```

### 3) Enable SSH access

Next we will add SSH access to the firewall rules. This allows SMASH or shell access to the BMC device itself.

The following command opens TCP Port 22 (0x16 0x00 == 22 in hex bigedian format)

```

fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x76 0x02 0x01 0x00 0x16 0x00
fmadio@fmadio100v2-228U:~$

```

This adds the following iptables rule

```

sysadmin [~]# iptables -L -n -v
chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
0 0 ACCEPT udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:623
2789 416K ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
124 9146 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:22
3131 965K DROP all -- * * 0.0.0.0/0 0.0.0.0/0 source IP range 0.0.0.0-255.255.255
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:53 length 1025:65535
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:53 length 1025:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:53 length 512:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 length 512:65535
1 40 ZERO_WINDOW_RECENT all -- * * 0.0.0.0/0 0.0.0.0/0 u32 "0x6&0xff=0x6&&0x4&0x1fff=0x0&&0x0>>0
chain FORWARD (policy ACCEPT 0 packets, 0 bytes)

```

IPMI Firewall SSH Access

Then confirm there are 2 firewall rules enabled.

```

fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x77 0x0
02
fmadio@fmadio100v2-228U:~$

```

At this point you can SSH into the BMC to confirm access is working correctly

### 4) Enable HTTPS access

Next add HTTPS access enabling the IPMI BMC Web client to be accessed.

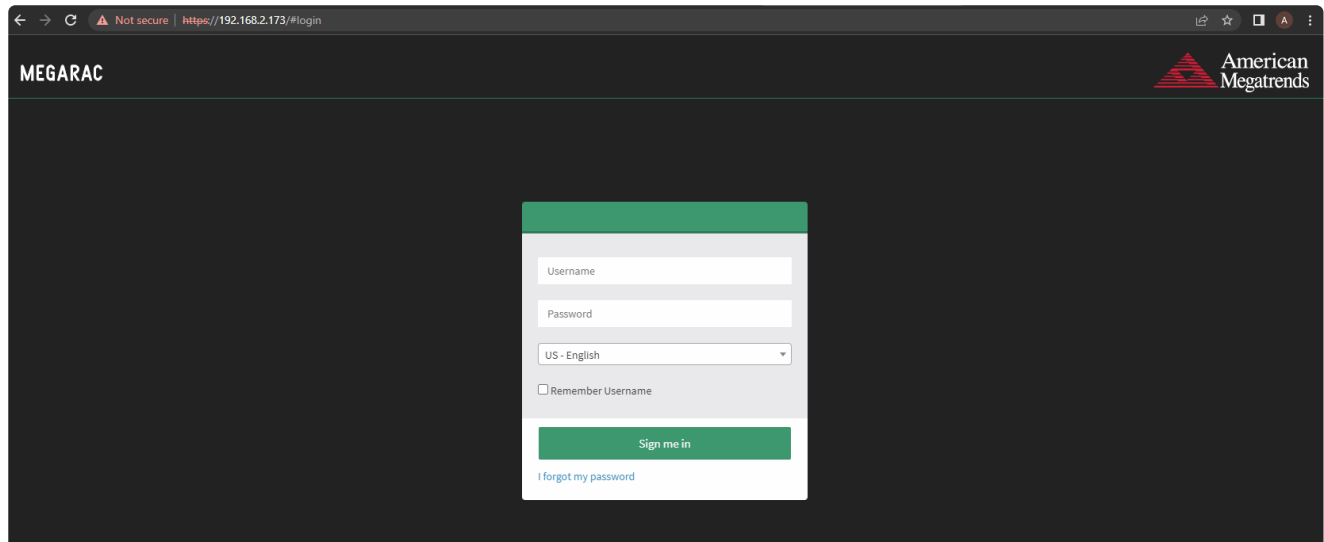
```
fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x76 0x02 0x01 0x00 0xbb 0x01
fmadio@fmadio100v2-228U:~$
```

This equates to the following iptables rules

```
sysadmin [~]# iptables -L -n -v
chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
0 0 ACCEPT udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:623
2789 416K ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
124 2140 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:22
3131 965K DROP all -- * * 0.0.0.0/0 0.0.0.0/0 source IP range 0.0.0.0-255.255.255
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:33 length 1025:65535
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:33 length 1025:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:33 length 512:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:33 length 512:65535
1 40 ZERO_WINDOW_RECENT all -- * * 0.0.0.0/0 0.0.0.0/0 u32 0x6&0xff=0x6&&0x4&0x1fff=0x0&
```

IPMI BMC HTTPS Access

At this point the IPMI BMC Webpage can be used such as the following



## 5) Add ipmitool access

Finally add ipmitool access which is on UDP port 623

```
fmadio@fmadio100v2-228U:~$ sudo ipmitool raw 0x32 0x76 0x02 0x01 0x01 0xf 0x02
fmadio@fmadio100v2-228U:~$
```

And the related iptables rule

```

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
0 0 ACCEPT udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:623
2789 416K ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:443
124 9146 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:22
3131 965K DROP all -- * * 0.0.0.0/0 0.0.0.0/0 source IP range 0.0.0.0-255.255.255
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:53 length 1025:65535
0 0 DROP tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp spt:53 length 1025:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp dpt:53 length 512:65535
0 0 DROP udp -- * * 0.0.0.0/0 0.0.0.0/0 udp spt:53 length 512:65535
1 40 ZERO_WINDOW_RECENT all -- * * 0.0.0.0/0 0.0.0.0/0 u32 "0x6&0xff=0x6&&0x4&0x1fff=0x0&&

```

IPMITOOL Firewall Setting

This enable ipmitool to work over the network, which can be extremely critical and helpful when troubleshooting problems. Such as the following

```

aaron@ingress:~$ ipmitool -U admin -P secret -H 192.168.2.173 power status
Chassis Power is on
aaron@ingress:~$

```

### Conclusion

While its quite cumbersome to use ipmitool raw mode to add and remove all these filters, the net result is a fairly secure BMC locked down with standard linux iptables.

Any questions or trouble please contact support.

# Custom CRON Job

FMADIO System uses a default cronjob that is located as below

```
/etc/crontab.root
```

Direct edits to this file will be lost every reboot, as the directory is on the RAMDisk and not persistent.

This can be customized and made persistent across reboots, for anything such as scheduled weekly reboots. By creating the cronjob file as bellow.

```
cp /etc/crontab.root /opt/fmadio/etc/
```

This puts the crontab.root file into persistent storage at

```
/opt/fmadio/etc/
```

Allowing custom cronjobs to run on the system.

## Scheduled Reboot

One example of this is to schedule a system reboot every Sunday night. To do this please see the below example.

```
fmadio@fmadio20v2-149:~$ cat /opt/fmadio/etc/crontab.root
* * * * * /opt/fmadio/bin/watchdog.lua --nocall >> /mnt/store0/log/watchdog.log
59 23 * * 0 /sbin/reboot
fmadio@fmadio20v2-149:~$
```

Adding in the last line in the cronjob will schedule the reboot at midnight every Sunday.

**NOTE:** Please reboot the system or manually reload cron for the new setting to take effect



# Disk Encryption

On select models of FMADIO capture systems full disk encryption is available. When available it uses the SSD drivers controller firmware to provide AES256 encryption with the OPAL interface standard.

States of the system is as follows

**Power Off:**

All data is encrypted accessing requires a password

**First Power On:**

Drives are accessible but data remains encrypted

**First Power On Unlock:**

Each drive in the system is unlocked by a shared password. This allows the drives media to be written/read from

**Warm Reboot:**

After Unlock the drives remain unlocked

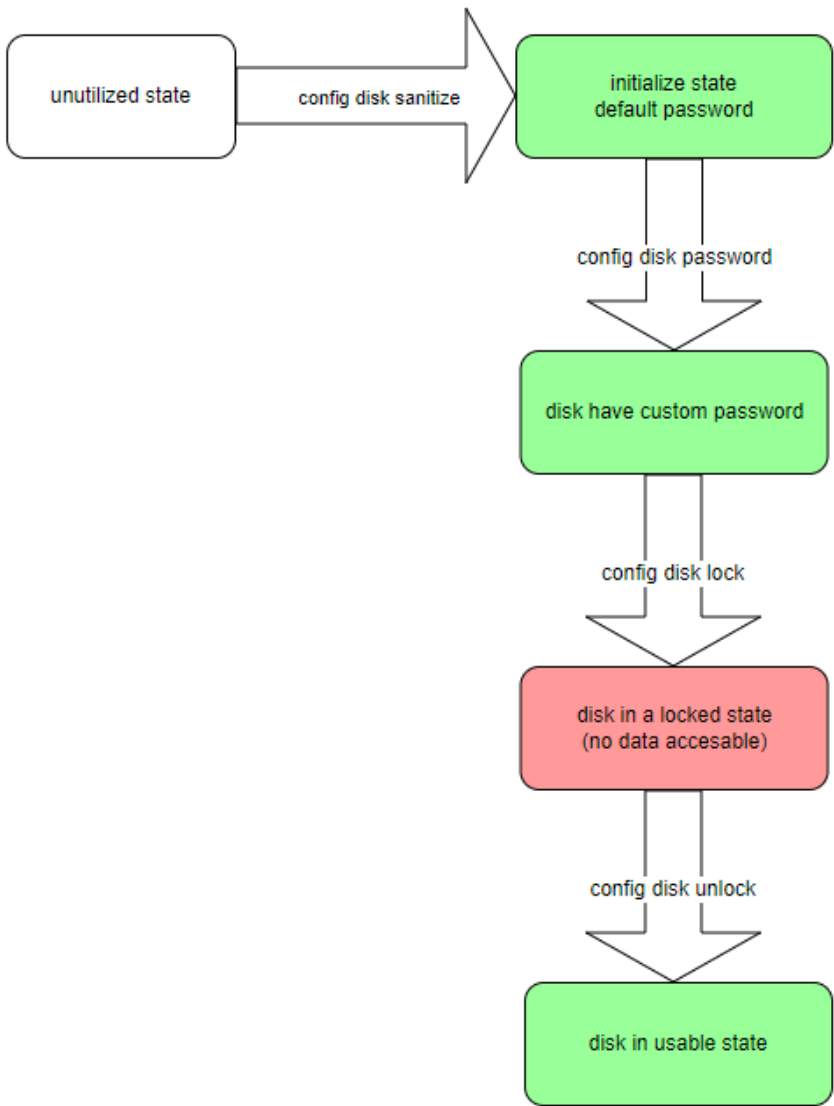
**Power Off:**

On power loss to the disks, all data becomes un-accessible and fully encrypted

Data is encrypted using AES256 and a random key generated by the SSD Controller. The Password specified encrypts/decrypts this AES256 key allowing the controller to read/write from the media. This encryption key is only kept in volatile RAM, thus when power to the drive is removed, the encryption key is lost. Once the encryption key is gone all data on the storage media can not be read.

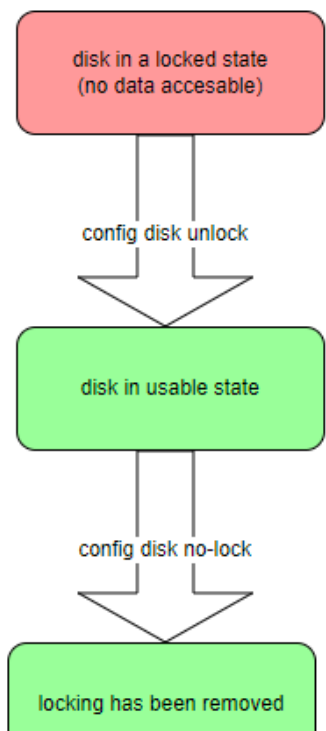
Drives can never be "bricked" as the drives can be reset by creating a new AES256 key. This reset however will remove all data previously written to the drive.

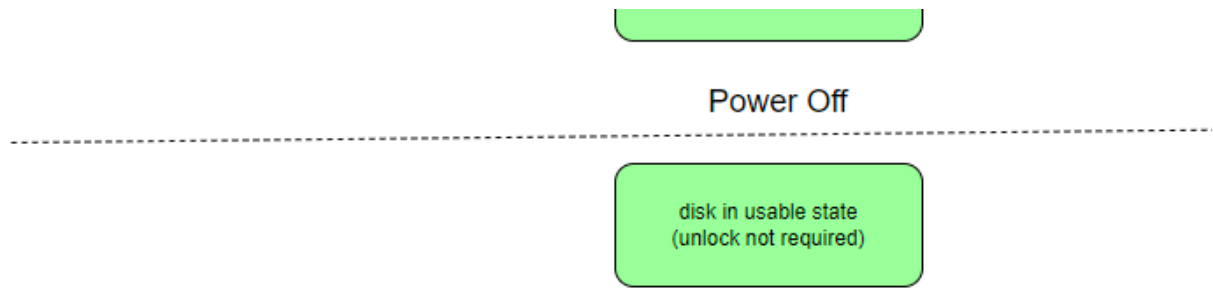
**Disk state flow chart**



Power Off

---





Encrypted Disk flow chart

## Drive Encryption Status

```
fmadiocli "show disk status"
```

This Operation displays the drive encryption state, documentation found here

<https://docs.fmad.io/fmadio-documentation/cli-reference/fmadiocli#show-disk-status>

## Sanitize / Format

```
fmadiocli "config disk sanitize"
```

This sets a new encryption password on all the disks. It will also WIPE ALL DATA ON THE CURRENT DISK. Please ensure there is no critical data on the system before running this command. After the operation the disks are in an UNLOCKED state.

<https://docs.fmad.io/fmadio-documentation/cli-reference/fmadiocli#config-disk-sanitize>

Setting Password

## Disk Password

Drives always have a password set, either a default password or a custom password the following command sets the disk password used for unlocking



fmadiocli  
fmadio documentation

## Locking Disk

By default disks are not a locked state. e.g. they do not require a password to be usable. This setting puts the disks into a locked state which when ever it loses power requires a password to unlock and make usable



## Disable Locking

There are cases where disabling the Locking function without destroying the data is helpful, note this is different from unlocking the disks.

This function disables requiring a password to access the disks persistently.

Unlocking the disks allows disk access but is not persistent

The functions are similar but quite different, documentation on how to perform this is

<https://docs.fmad.io/fmadio-documentation/cli-reference/fmadiocli#config-disk-no-lock>

## Unlocking

```
fmadiocli "config disk unlock"
```

Command will unlock and verify correct functionality of the disks. Starts by unlocking each disk using the supplied password, then reloading all storage components of the FMADIO system which reference the disks.

Reference documentation is



## Automatic Unlock

Generally entering passwords manually is fairly cumbersome / time intensive. The system has the ability to unlock the disks on boot.

The general purpose boot setup file `/opt/fmadio/etc/boot.lua` can be used to fetch the password from a key server, e.g. via CURL or copy a password file from the local disk

Below is an example boot.lua file to automatically unlock the disks at boot time, using the saved password in `/opt/fmadio/etc/disk-password`

```
-- unlock the disks
print(os.date() .. " : unlocking sed disks")
io.stdout:flush()

-- fetch password local
os.execute('cp /mnt/store0/etc/disk-password /tmp/disk-password')

-- fetch password remote
--os.execute('curl -s https://192.168.1.1/disk-password > /tmp/disk-password')

-- unlock
os.execute('/opt/fmadio/bin/fmadiocli "config disk unlock"')
os.execute('/opt/fmadio/bin/fmadiocli "show disk status"')
io.stdout:flush()

print(os.date() .. " : unlocking sed disks... done")
io.stdout:flush()
```

For debugging the logfile is located in

/mnt/store0/log/boot.log

example output is shown below

```

Fri Jul 21 14:53:53 2023 : unlocking sed disks
[Fri Jul 21 14:53:53 2023]      -----
[Fri Jul 21 14:53:53 2023]  _/  ____\_____  _____  __|  _/|__|  ____
[Fri Jul 21 14:53:53 2023]  \  __\  /  \  \  \  \  /  __  |  |  |  /  _  \
[Fri Jul 21 14:53:53 2023]  |  |  |  Y Y  \  /  __  \  /  /  /  |  |  |  (  <_>  )
[Fri Jul 21 14:53:53 2023]  |__|  |__|  |  /  (_____  /  \_____  |  |__|  \____/
[Fri Jul 21 14:53:53 2023]      \  /      \  /      \  /
[Fri Jul 21 14:53:53 2023]  =====
[Fri Jul 21 14:53:53 2023]    +- Packets confiscated by Customs +-
[Fri Jul 21 14:53:53 2023]
[Fri Jul 21 14:53:53 2023]    type '?' for command information
[Fri Jul 21 14:53:53 2023]    type '???' for verbose information
[Fri Jul 21 14:53:53 2023]
[Fri Jul 21 14:53:53 2023] History: 1804
[Fri Jul 21 14:53:53 2023] CmdLine [config disk unlock]
[Fri Jul 21 14:53:53 2023] Cmd [config disk unlock]
[Fri Jul 21 14:53:53 2023] Enter Current Password. or Blank for default:
[Fri Jul 21 14:53:53 2023] -----
[Fri Jul 21 14:53:53 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme8n1
[Fri Jul 21 14:53:53 2023] [par0] /dev/nvme8n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:53:53 2023]
[Fri Jul 21 14:53:53 2023] [par0] set to unlocked S64HNJ0T707660
[Fri Jul 21 14:53:53 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:53:54 2023] [par0] LockingRange0 set to RW
[Fri Jul 21 14:53:54 2023] -----
[Fri Jul 21 14:53:54 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme2n1
[Fri Jul 21 14:53:54 2023] [ssd0] /dev/nvme2n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:53:54 2023]
[Fri Jul 21 14:53:54 2023] [ssd0] set to unlocked S64HNJ0T707662
[Fri Jul 21 14:53:54 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:53:55 2023] [ssd0] LockingRange0 set to RW
[Fri Jul 21 14:53:55 2023] -----
[Fri Jul 21 14:53:55 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme9n1
[Fri Jul 21 14:53:55 2023] [ssd1] /dev/nvme9n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:53:55 2023]
[Fri Jul 21 14:53:55 2023] [ssd1] set to unlocked S64HNJ0T707794
[Fri Jul 21 14:53:55 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:53:56 2023] [ssd1] LockingRange0 set to RW
[Fri Jul 21 14:53:56 2023] -----
[Fri Jul 21 14:53:56 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme7n1
[Fri Jul 21 14:53:56 2023] [ssd2] /dev/nvme7n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:53:56 2023]
[Fri Jul 21 14:53:56 2023] [ssd2] set to unlocked S64HNJ0T707657
[Fri Jul 21 14:53:56 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:53:58 2023] [ssd2] LockingRange0 set to RW
[Fri Jul 21 14:53:58 2023] -----
[Fri Jul 21 14:53:58 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme6n1
[Fri Jul 21 14:53:58 2023] [ssd3] /dev/nvme6n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:53:58 2023]
[Fri Jul 21 14:53:58 2023] [ssd3] set to unlocked S64HNJ0T707659
[Fri Jul 21 14:53:58 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:53:59 2023] [ssd3] LockingRange0 set to RW

```

```

[Fri Jul 21 14:53:59 2023] -----
[Fri Jul 21 14:53:59 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme4n1
[Fri Jul 21 14:53:59 2023] [ssd4] /dev/nvme4n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:53:59 2023]
[Fri Jul 21 14:53:59 2023] [ssd4] set to unlocked S64HNJ0T707658
[Fri Jul 21 14:53:59 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:54:00 2023] [ssd4] LockingRange0 set to RW
[Fri Jul 21 14:54:00 2023] -----
[Fri Jul 21 14:54:00 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme5n1
[Fri Jul 21 14:54:00 2023] [ssd5] /dev/nvme5n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:54:00 2023]
[Fri Jul 21 14:54:00 2023] [ssd5] set to unlocked S64HNJ0T707656
[Fri Jul 21 14:54:00 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:54:01 2023] [ssd5] LockingRange0 set to RW
[Fri Jul 21 14:54:01 2023] -----
[Fri Jul 21 14:54:01 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme3n1
[Fri Jul 21 14:54:01 2023] [ssd6] /dev/nvme3n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:54:01 2023]
[Fri Jul 21 14:54:01 2023] [ssd6] set to unlocked S64HNJ0T707655
[Fri Jul 21 14:54:01 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:54:02 2023] [ssd6] LockingRange0 set to RW
[Fri Jul 21 14:54:02 2023] -----
[Fri Jul 21 14:54:02 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme1n1
[Fri Jul 21 14:54:02 2023] [ssd7] /dev/nvme1n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jul 21 14:54:02 2023]
[Fri Jul 21 14:54:02 2023] [ssd7] set to unlocked S64HNJ0T707793
[Fri Jul 21 14:54:02 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Fri Jul 21 14:54:03 2023] [ssd7] LockingRange0 set to RW
[Fri Jul 21 14:54:03 2023]
[Fri Jul 21 14:54:03 2023] _/ ____\_____  _____  __|_/_|__| ____
[Fri Jul 21 14:54:03 2023] \  __\ /  \ / \__ \ / __ | | | / _ \
[Fri Jul 21 14:54:03 2023] | | | Y Y \ / __ \ / / | | | ( <_> )
[Fri Jul 21 14:54:03 2023] |__| |__| | / (____ \ \____ | |__| \____/
[Fri Jul 21 14:54:03 2023]          \ /      \ /      \ /
[Fri Jul 21 14:54:03 2023] =====
[Fri Jul 21 14:54:03 2023] +- Packets confiscated by Customs +-
[Fri Jul 21 14:54:03 2023]
[Fri Jul 21 14:54:03 2023] type '?' for command information
[Fri Jul 21 14:54:03 2023] type '???' for verbose information
[Fri Jul 21 14:54:03 2023]
[Fri Jul 21 14:54:03 2023] History: 1804
[Fri Jul 21 14:54:03 2023] CmdLine [show disk status]
[Fri Jul 21 14:54:03 2023] Cmd [show disk status]
[Fri Jul 21 14:54:03 2023] SSD Cache
[Fri Jul 21 14:54:04 2023] Disk :                Serial :      Size : Temp  : Used : Error
[Fri Jul 21 14:54:04 2023] -----+-----+-----+-----+-----+-----
[Fri Jul 21 14:54:04 2023] os0   :                1949256760C6 : 0.00 TB : 0 C : 0 % : 0
[Fri Jul 21 14:54:04 2023] par0  :                S64HNJ0T707660 : 0.00 TB : 30 C : 0 % : 0
[Fri Jul 21 14:54:04 2023] ssd0  :                S64HNJ0T707662 : 3.84 TB : 35 C : 0 % : 0
[Fri Jul 21 14:54:04 2023] ssd1  :                S64HNJ0T707794 : 3.84 TB : 30 C : 0 % : 0
[Fri Jul 21 14:54:04 2023] ssd2  :                S64HNJ0T707657 : 3.84 TB : 31 C : 0 % : 0
[Fri Jul 21 14:54:04 2023] ssd3  :                S64HNJ0T707659 : 3.84 TB : 29 C : 0 % : 0
[Fri Jul 21 14:54:04 2023] ssd4  :                S64HNJ0T707658 : 3.84 TB : 30 C : 0 % : 0

```

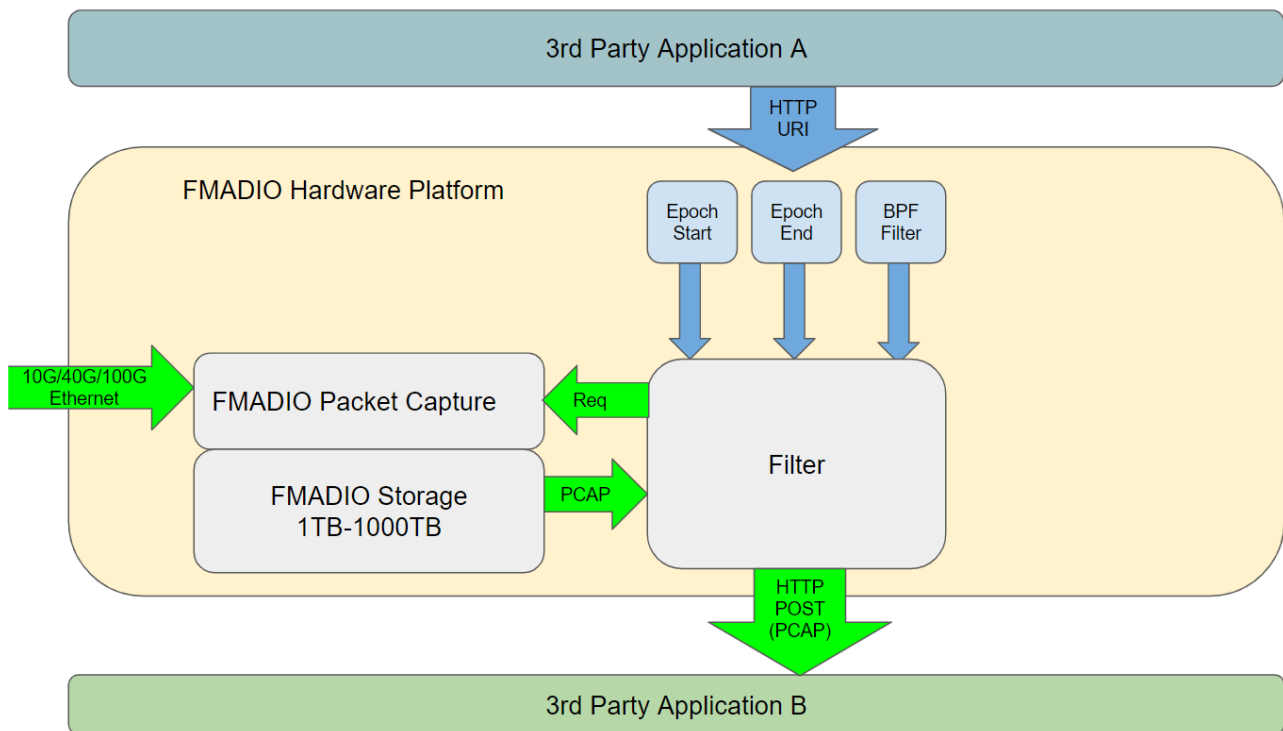




# External Web Application

FW: 7738+

FMADIO Packet capture systems have the ability to push specific PCAPs to a external 3rd Party application as follows



FMADIO External HTTP POST Packet Capture PCAP

This workflow enables a simple approach, using a URI to push a PCAP over HTTP using a POST request to a remote end application, then following a JSON redirect.

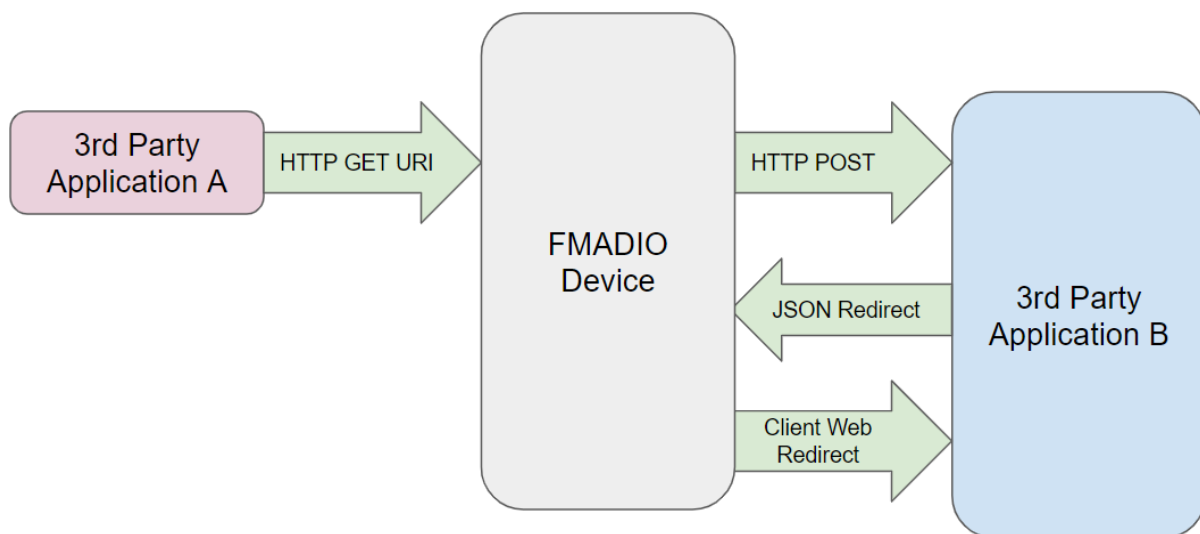
In this example, we are using our internally developed FMADIO Packet Scope and FMADIO Shark (FShark) as a reference example. This is for demonstration purposes only, any 3rd party web application will work.

## Request URI

The workflow process is as follows,

1. Web Application A generates a en.loader.html URI. For example Epoch Start/End and an BPF Filter
2. Web Application A directs the Client to this URI
3. FMADIO Packet Capture System presents the Client with the FMADIO Loader web page. This shows the progress of filtering and upload to Application B and any potential errors.
4. Application B completes upload, and returns a redirect URI
5. FMADIO Loader web page follows the redirect, allowing the client to load seamlessly into Web Application B with the new PCAP data.

The example is using FMADIO developed software for this, however there is no limitation, any 3rd party application will work.



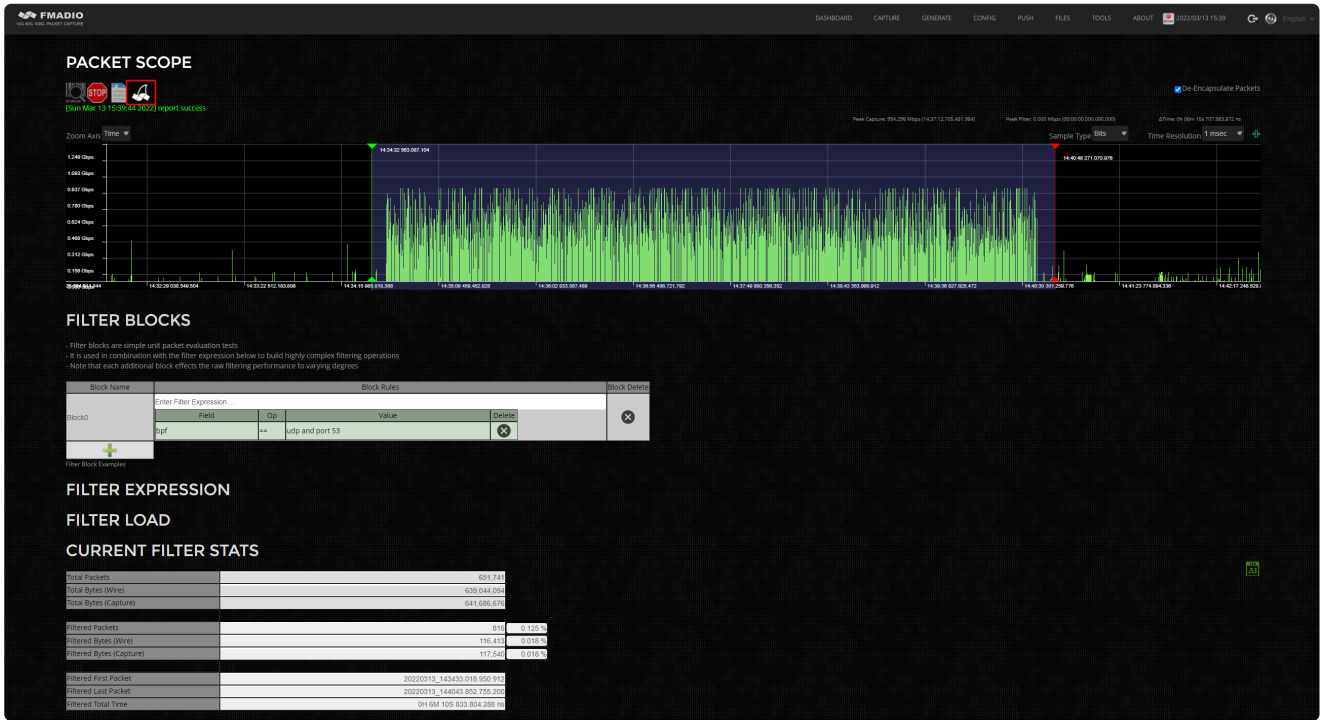
Web application Workflow

## Example (FMADIO Shark)

In the following example we are using FMADIO Packet Scope as as (Application A) and FMADIO Shark as (Application B).

### Web Application A

In this example, Web Application A is "FMADIO Packet Scope" with Web Application B "FMADIO Shark". PacketScope generates the loader link request as follows



FMADIO Shark URL

The above example has an Epoch Start time and Epoch End time as well as a BPF Filter applied, the end result is the following URI

```
http://192.168.1.100/en.search.html?
  Path=/api/v1/pcap/splittime&
  Target=fshark&
  StreamName=coffee_20220313_1227&
  TSBegin=1647149672563087104ULL&
  TSEnd=1647150048271070976ULL&
  FilterBPF=udp%20and%20port%20%2053&&
```

This redirects to the FMADIO Loader page which processes and pushes the above Filter specification to the Target "fshark"

Expanding on the details

```
Path=/api/v1/pcap/splittime&
```

**Path** indicates how the PCAP should be generated, in this case its a specific capture name with filters.

```
Target=fshark&
```

**Target**, informs what the End Point target is. "fshark" is an internally defined EP. Due to security reasons End Point definitions can only be configured on the FMADIO System. Only the enumerated name of the End Point is used in the URI.

```
StreamName=coffee_20220313_1227&
```

**StreamName** specifies the name of the Capture to process.

```
TSEgin=1647149672563087104ULL&
```

**TSEgin** is the Epoch in Nanoseconds for the start of the Filter

```
TSEnd=1647150048271070976ULL&
```

**TSEnd** is the Epoch in Nanoseconds for the end of the Filter

```
FilterBPF=udp%20and%20port%20%2053&&
```

**FilterBPF** is the Escape Encoded BPF filter, in this case "udp and port 53" e.g. extract DNS traffic.

For full details please check the API v1 Documentation page



Summary

fmdio documentation

[API Documentation](#)

## Search Page

Once clicked the following Search page is visible.

The screenshot shows a web browser window with the URL `192.168.8.95/en/loader.html?Path=/api/v1/pcap/splittime&Target=fshark&StreamName=coffee_20220313_1227&TSEgin=1647149672563087104ULL&TSEnd=1647150048271070976ULL&FilterBPF=udp%20and%20port%20%2053&NoRedirect=true&`. The page title is "PCAP LOADER" and it shows a status "Done... redirecting". Below this is a table of parameters:

Application	fshark
Redirect	fshark/en/pcapview.html?filename=fmdio20p3_1647154945535058944_pcap
FileName	coffee_20220313_1227
PCAP Time Start	1647149672563087104ULL
PCAP Time Stop	1647150048271070976ULL
PCAP BPF Filter	udp and port 53
PCAP Port Filter	

[Loader Page](#)

In addition we added the following URI to prevent automatic reload, this can be helpful for debugging purposes.

```
NoRedirect=true&
```

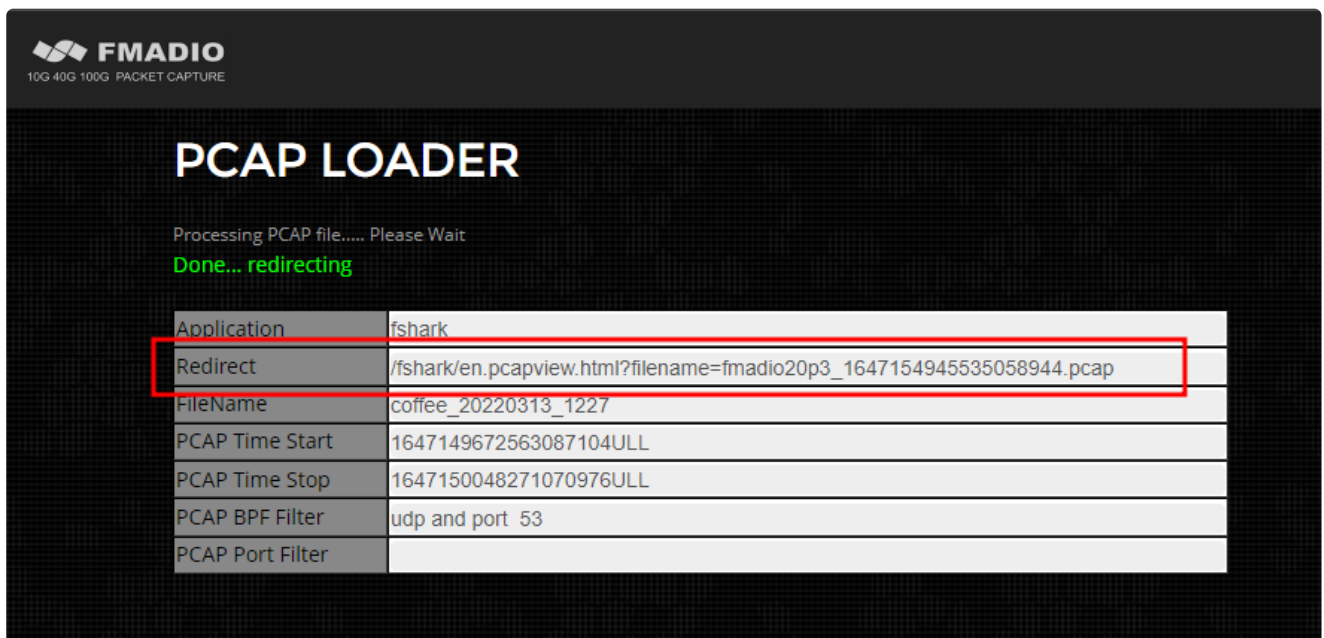
Internally the FMADIO Device is issuing the following HTTP POST command thru CURL on the filtered PCAP. This URL generator is configuration on the FMADIO Device, almost anything is possible. This example the "fshark" Target is built into the system firmware.

```
cat fmadio.pcap | /usr/local/bin/curl -s -X POST -F "data=@-" "http://192.168.1.101/api/v
```

Once completed the above HTTP POST request into Web Application B is completed, it returns a redirect as follows to FMADIO Packet Capture System

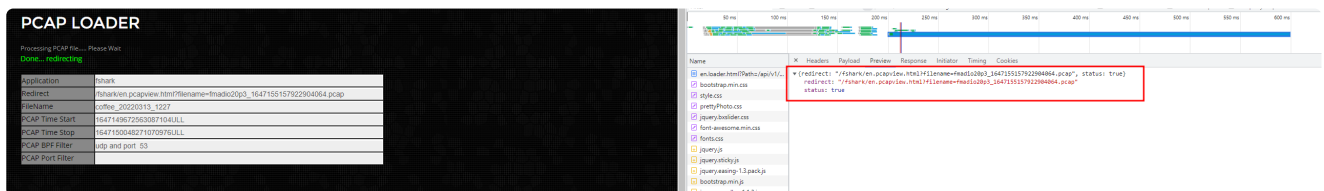
```
{"redirect": "\fshark\en.pcapview.html?filename=fmadio20p3_1647154945535058944.pcap", "statu
```

This redirect URL is forwarded to the Web Client is shown below



Completed Redirect

The 3rd Party Application returns the redirect as the response to the POST upload request. This is where the redirect URI is specific, or any error condition



Application B Redirect JSON URI

## Redirect Page

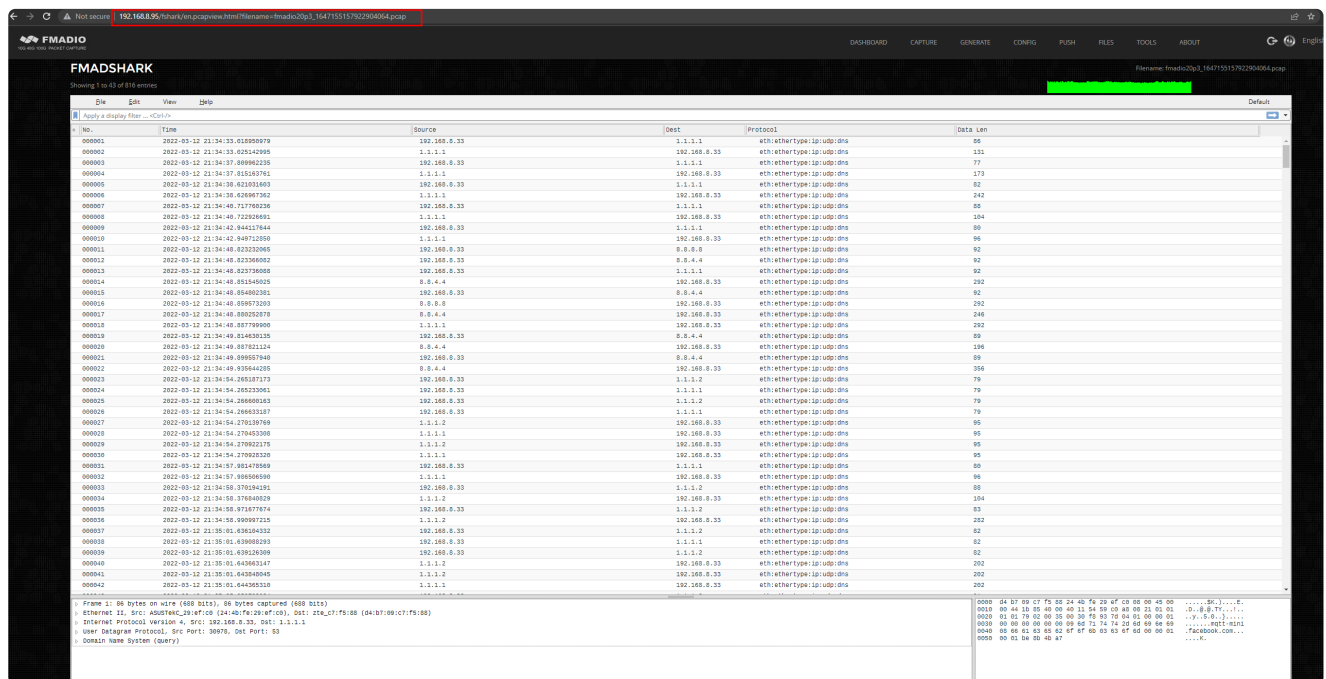
After the HTTP Post has been completed the 3rd Party Application B returns a redirect page. The FMADIO Loader then redirects to this location.

In this example it loads FMADIO Shark with the URI

```
/fshark/en.pcapview.html?filename=fmadio20p3_1647155157922904064.pcap
```

The following page is what the 3rd Party Application displays.

NOTE: Due to FMADIO Shark being internally developed the Web page does look like FMADIO products. This page could show anything, there's no iframes etc.



3rd Party Application B

## Recommended

The recommended URI is to use the /api/v1/pcap/timerange URI endpoint. As this does not require any stream names, just Epoch start/end times and a BPF filter.

Example as follows, please remember to Escape Encode the FilterBPF string.

```
http://192.168.1.100/en.search.html?
Path=/api/v1/pcap/timerange&
Target=fshark&
TSBegin=1647149672563087104ULL&
TEnd=1647150048271070976ULL&
FilterBPF=udp%20and%20port%20%2053&&
```

# BMC IPMI Settings

## Disable SSDP

In many cases disabling SSDP is required. This reduces the public visibility of the Packet Capture system.

To do this please use the Redfish API as follows

### Step 1 - Current Network Profile

Get the current BMC Network profile as follows

```
curl -u admin:xxxxxxx -k "https://{BMC Address}/redfish/v1/Managers/Self/NetworkProtocol" |
```

The output will be similar to this

```

fmadio@fmadio100v2-228U:~$ curl -u admin:xxxxxx -k "https://192.168.1.100/redfish/v1/Manager
{
  "@odata.context": "/redfish/v1/$metadata#ManagerNetworkProtocol.ManagerNetworkProtocol",
  "@odata.etag": "W/\\"1655790673\"",
  "@odata.id": "/redfish/v1/Managers/Self/NetworkProtocol",
  "@odata.type": "#ManagerNetworkProtocol.v1_4_1.ManagerNetworkProtocol",
  "Description": "Network Protocol Details",
  "HTTPS": {
    "Port": 443,
    "ProtocolEnabled": true
  },
  "HostName": "AMIE0D55E5D2156",
  "IPMI": {
    "Port": 623,
    "ProtocolEnabled": true
  },
  "Id": "NetworkProtocol",
  "KVMIP": {
    "Port": 443,
    "ProtocolEnabled": true
  },
  "NTP": {
    "NTPServers": [
      "time.nist.gov",
      "pool.ntp.org"
    ],
    "Port": 123,
    "ProtocolEnabled": false
  },
  "Name": "Manager Network Protocol",
  "SNMP": {
    "Port": 161,
    "ProtocolEnabled": true
  },
  "SSDP": {
    "Port": 1900,
    "ProtocolEnabled": false
  },
  "Status": {
    "Health": "OK",
    "State": "Enabled"
  },
  "VirtualMedia": {
    "Port": 443,
    "ProtocolEnabled": true
  }
}
fmadio@fmadio100v2-228U:~$

```

The important part is extracting the ekey required for the next step. In this case the value is "1655790673"





```
curl -u admin:xxxxxxx -k "https://{BMC Address}/redfish/v1/Managers/Self/NetworkProtocol" |
```

```
@odata.etag : w/\ 16557906/3\
"@odata.id": "/redfish/v1/Managers/Self/NetworkProtocol",
"@odata.type": "#ManagerNetworkProtocol.v1_4_1.ManagerNetworkProtocol",
"Description": "Network Protocol Details",
"HTTPS": {
  "Port": 443,
  "ProtocolEnabled": true
},
"HostName": "AMIE0D55E5D2156",
"IPMI": {
  "Port": 623,
  "ProtocolEnabled": true
},
"Id": "NetworkProtocol",
"KVMIP": {
  "Port": 443,
  "ProtocolEnabled": true
},
"NTP": {
  "NTPServers": [
    "time.nist.gov",
    "pool.ntp.org"
  ],
  "Port": 123,
  "ProtocolEnabled": false
},
"Name": "Manager Network Protocol",
"SNMP": {
  "Port": 161,
  "ProtocolEnabled": true
},
"SSDP": {
  "Port": 1900,
  "ProtocolEnabled": false
},
"Status": {
  "Health": "OK",
  "State": "Enabled"
},
"VirtualMedia": {
  "Port": 443,
  "ProtocolEnabled": true
}
}
}
fmdio@fmdio100v2-228u:~$
fmdio@fmdio100v2-228u:~$
```

# Disk Packs

FMADIO100 Capture system supports multiple data "Disk Packs" per capture system.

The system at boot time detect which "Disk Pack" that is currently installed, and uses that as the currently active config file.

## Disk Swap

FMADIO100Gv2 capture system has 10 x NVMe U.2 drive slots, of these 9 of the disks are used for data, and one disk is used for the OS. Below is the location of the OS disk. **The OS disk can not be removed or swapped, must always be inserted at the location below (top left hand corner).**



Location of the OS Disk

The location of the remaining 9 data disks when installed can be in any order.

### 1) Disk Swap

Power down the system

### 2) Disk Swap

Remove all 9 data disks from the current chassis and place into the provided storage case.

### 3) Disk Swap

Insert the new OS disk into the top left slot, then each of the new 9 data disks into any location. Be careful to not use excessive force as it may damage the backplane and device.

### 4) Disk Swap

Power on the device, it will take 90-120sec to fully boot. Once fully booted check the disks have been recognized by running "ls" in the directory

```
/opt/fmadio/disk/
```

Correct listing will look like below, `ssd0 - ssd7` and `par0` are mapped to a physical end point. If any disk is missing, please reseal the disk and reboot the system.

```

fmadio@fmadio100v2-228U:~$ ls -al /opt/fmadio/disk/
total 0
drwxr-xr-x    2 root    root          380 Sep 29 12:23 ./
drwxr-xr-x   13 fmadio  staff        460 Sep 29 12:23 ../
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 par0 -> /dev/nvme9n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd0 -> /dev/nvme5n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd0_capture -> /dev/nvme5n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd1 -> /dev/nvme6n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd1_capture -> /dev/nvme6n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd2 -> /dev/nvme7n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd2_capture -> /dev/nvme7n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd3 -> /dev/nvme8n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd3_capture -> /dev/nvme8n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd4 -> /dev/nvme4n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd4_capture -> /dev/nvme4n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd5 -> /dev/nvme2n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd5_capture -> /dev/nvme2n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd6 -> /dev/nvme3n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd6_capture -> /dev/nvme3n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd7 -> /dev/nvme1n1
lrwxrwxrwx    1 root    root           12 Sep 29 12:23 ssd7_capture -> /dev/nvme1n1
fmadio@fmadio100v2-228U:~$

```

5) Done

System is now ready for capture and operation

## One Time Setup

When a Disk Pack is first delivered, it requires configuration. This is a one time operation that can be done in a lab environment.

The steps are as follows

1) Install the new Disk Pack

Per the above instructions, install the new disk pack hardware. Including OS and 9 data disks.

2) Power on the system

Power on the system, it will have no network connectivity, but serial port and VGA are operational.

3) Configure Management Interface

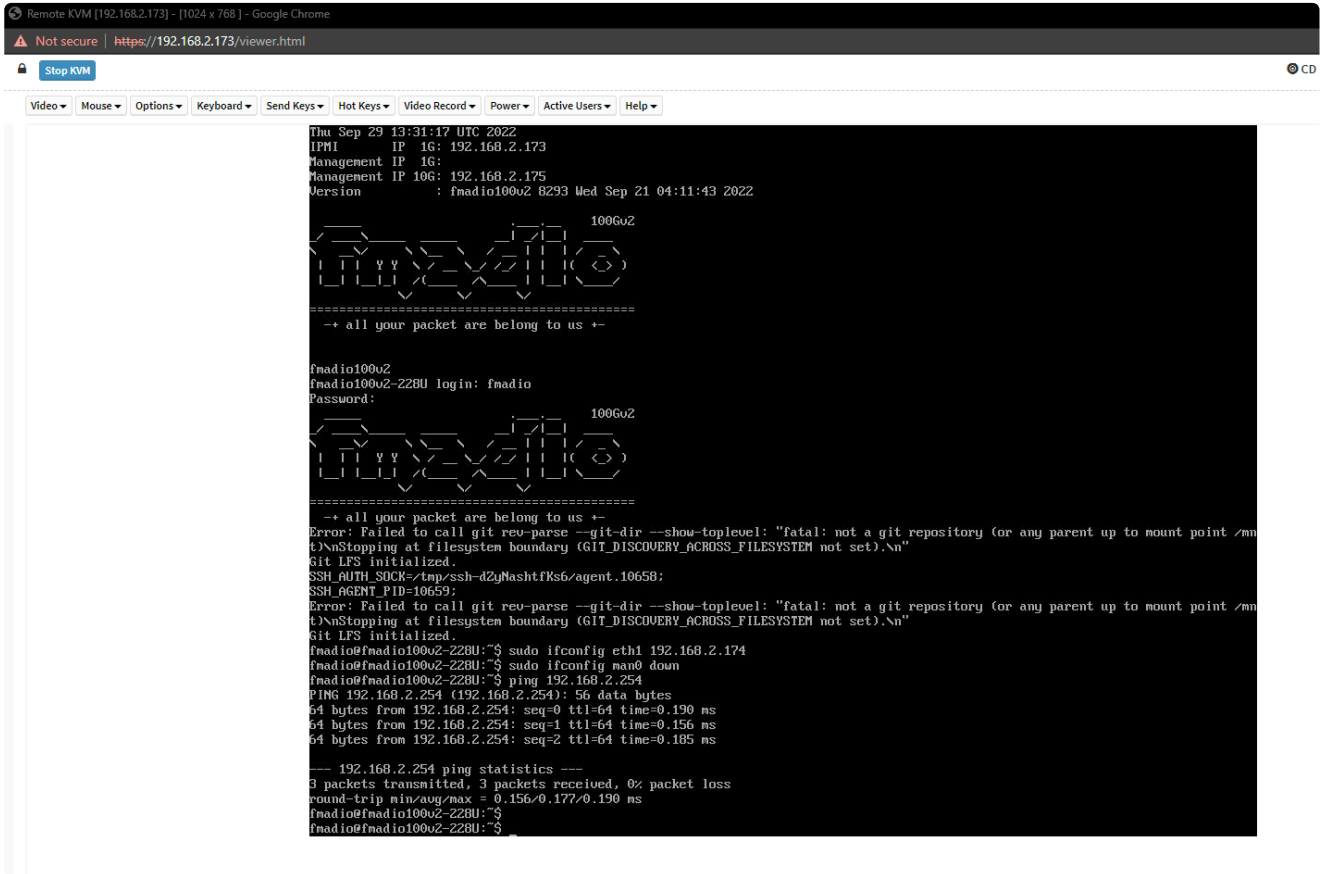
The system will be behaving oddly, and is partially operational. Connect to the VGA or Serial port and login with the default passwords.

NOTE: SSH network connectivity will not be functional.

#### 4) Setup Management Interface

Manually configure the 1G management interface with ifconfig, example below

```
sudo ifconfig eth0 192.168.2.173
```



The screenshot shows a terminal window within a remote KVM environment. The terminal output includes system information, a login prompt for 'fnadio', and the execution of the 'ifconfig' command to set the IP address of the 'eth1' interface to 192.168.2.174. It also shows the 'ifconfig' command being used to bring the 'man0' interface down, followed by a successful ping to 192.168.2.254. The terminal output is as follows:

```
Thu Sep 29 13:31:17 UTC 2022
IPMI IP 16: 192.168.2.173
Management IP 16:
Management IP 106: 192.168.2.175
Version : fmadio100u2 8293 Wed Sep 21 04:11:43 2022

=====
          100Gu2
=====
-- all your packet are belong to us --

fnadio@fnadio100u2
fnadio@fnadio100u2-228U login: fnadio
Password:
=====
          100Gu2
=====
-- all your packet are belong to us --

Error: Failed to call git rev-parse --git-dir --show-toplevel: "fatal: not a git repository (or any parent up to mount point /mnt)
\nStopping at filesystem boundary (GIT_DISCOVERY_ACROSS_FILESYSTEM not set).\n"
Git LFS initialized.
SSH_AUTH_SOCK=/tmp/ssh-d2yNashfKs6/agent.10658:
SSH_AGENT_PID=10659:
Error: Failed to call git rev-parse --git-dir --show-toplevel: "fatal: not a git repository (or any parent up to mount point /mnt)
\nStopping at filesystem boundary (GIT_DISCOVERY_ACROSS_FILESYSTEM not set).\n"
Git LFS initialized.
fnadio@fnadio100u2-228U:~$ sudo ifconfig eth1 192.168.2.174
fnadio@fnadio100u2-228U:~$ sudo ifconfig man0 down
fnadio@fnadio100u2-228U:~$ ping 192.168.2.254
PING 192.168.2.254 (192.168.2.254): 56 data bytes
64 bytes from 192.168.2.254: seq=0 ttl=64 time=0.190 ms
64 bytes from 192.168.2.254: seq=1 ttl=64 time=0.156 ms
64 bytes from 192.168.2.254: seq=2 ttl=64 time=0.185 ms

--- 192.168.2.254 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.156/0.177/0.190 ms
fnadio@fnadio100u2-228U:~$
fnadio@fnadio100u2-228U:~$
```

Manually configuring the 1G interface via HTML5 BMC

The interface may be eth0 or eth1 depending on the config

#### 4) SSH into the system

Once 1G connectivity is established ssh and scp work no problem. SSH into the system

#### 5) Update the udev rules

Update the udev rules with the vendor provided file. Either scp or use a text editor. The file is located in

```
/opt/fmadio/etc/60-persistent-ethernet.rules
```

#### 6) Reboot Done

After copying the file reboot the system. The system should have normal operation.



# CPU Mapping

**FW: 8325+**

FMADIO Packet Capture system have 16 to 96 CPUs, this can make configuration and allocation of all these CPU resources tedious and error prone.

To assist the configuration file located in

```
/opt/fmadio/etc/time.lua
```

There is a section name CPUMap, an example is shown below

```

["CPUMap"] =
{
  ["Enable"] = true,
  [ 0] = "fmadio",      -- CPU:  0 Core:  0 Thread:0 Node:  0 system
  [ 1] = "",           -- CPU:  1 Core:  1 Thread:0 Node:  0
  [ 2] = "fmadio",    -- CPU:  2 Core:  2 Thread:0 Node:  0 system
  [ 3] = "fmadio",    -- CPU:  3 Core:  3 Thread:0 Node:  0 system
  [ 4] = "",           -- CPU:  4 Core:  4 Thread:0 Node:  0
  [ 5] = "",           -- CPU:  5 Core:  5 Thread:0 Node:  0
  [ 6] = "",           -- CPU:  6 Core:  6 Thread:0 Node:  0
  [ 7] = "",           -- CPU:  7 Core:  7 Thread:0 Node:  0
  [ 8] = "",           -- CPU:  8 Core:  8 Thread:0 Node:  0
  [ 9] = "",           -- CPU:  9 Core:  9 Thread:0 Node:  0
  [10] = "",           -- CPU: 10 Core: 10 Thread:0 Node:  0
  [11] = "",           -- CPU: 11 Core: 11 Thread:0 Node:  0
  [12] = "",           -- CPU: 12 Core: 12 Thread:0 Node:  0
  [13] = "",           -- CPU: 13 Core: 13 Thread:0 Node:  0
  [14] = "",           -- CPU: 14 Core: 14 Thread:0 Node:  0
  [15] = "",           -- CPU: 15 Core: 15 Thread:0 Node:  0
  [16] = "",           -- CPU: 16 Core: 16 Thread:0 Node:  0
  [17] = "",           -- CPU: 17 Core: 17 Thread:0 Node:  0
  [18] = "",           -- CPU: 18 Core: 18 Thread:0 Node:  0
  [19] = "",           -- CPU: 19 Core: 19 Thread:0 Node:  0
  [20] = "",           -- CPU: 20 Core: 20 Thread:0 Node:  0
  [21] = "",           -- CPU: 21 Core: 21 Thread:0 Node:  0
  [22] = "",           -- CPU: 22 Core: 22 Thread:0 Node:  0
  [23] = "",           -- CPU: 23 Core: 23 Thread:0 Node:  0
  [24] = "fmadio",    -- CPU: 24 Core:  0 Thread:0 Node:  1 system
  [25] = "fmadio",    -- CPU: 25 Core:  1 Thread:0 Node:  1 system
  [26] = "fmadio",    -- CPU: 26 Core:  2 Thread:0 Node:  1 system
  [27] = "fmadio",    -- CPU: 27 Core:  3 Thread:0 Node:  1 system
  [28] = "fmadio",    -- CPU: 28 Core:  4 Thread:0 Node:  1 system
  [29] = "fmadio",    -- CPU: 29 Core:  5 Thread:0 Node:  1 system
  [30] = "fmadio",    -- CPU: 30 Core:  6 Thread:0 Node:  1 system
  [31] = "fmadio",    -- CPU: 31 Core:  7 Thread:0 Node:  1 system
  [32] = "",           -- CPU: 32 Core:  8 Thread:0 Node:  1
  [33] = "",           -- CPU: 33 Core:  9 Thread:0 Node:  1
  [34] = "",           -- CPU: 34 Core: 10 Thread:0 Node:  1
  [35] = "",           -- CPU: 35 Core: 11 Thread:0 Node:  1
  [36] = "",           -- CPU: 36 Core: 12 Thread:0 Node:  1
  [37] = "",           -- CPU: 37 Core: 13 Thread:0 Node:  1
  [38] = "",           -- CPU: 38 Core: 14 Thread:0 Node:  1
  [39] = "",           -- CPU: 39 Core: 15 Thread:0 Node:  1
  [40] = "",           -- CPU: 40 Core: 16 Thread:0 Node:  1
  [41] = "",           -- CPU: 41 Core: 17 Thread:0 Node:  1
  [42] = "",           -- CPU: 42 Core: 18 Thread:0 Node:  1
  [43] = "",           -- CPU: 43 Core: 19 Thread:0 Node:  1
  [44] = "",           -- CPU: 44 Core: 20 Thread:0 Node:  1
  [45] = "",           -- CPU: 45 Core: 21 Thread:0 Node:  1
  [46] = "",           -- CPU: 46 Core: 22 Thread:0 Node:  1
  [47] = "",           -- CPU: 47 Core: 23 Thread:0 Node:  1
  [48] = "",           -- CPU: 48 Core:  0 Thread:1 Node:  0

```



```
[ 49] = "", -- CPU: 49 Core: 1 Thread:1 Node: 0
[ 50] = "", -- CPU: 50 Core: 2 Thread:1 Node: 0
[ 51] = "", -- CPU: 51 Core: 3 Thread:1 Node: 0
[ 52] = "", -- CPU: 52 Core: 4 Thread:1 Node: 0
[ 53] = "", -- CPU: 53 Core: 5 Thread:1 Node: 0
[ 54] = "", -- CPU: 54 Core: 6 Thread:1 Node: 0
[ 55] = "", -- CPU: 55 Core: 7 Thread:1 Node: 0
[ 56] = "", -- CPU: 56 Core: 8 Thread:1 Node: 0
[ 57] = "", -- CPU: 57 Core: 9 Thread:1 Node: 0
[ 58] = "", -- CPU: 58 Core: 10 Thread:1 Node: 0
[ 59] = "", -- CPU: 59 Core: 11 Thread:1 Node: 0
[ 60] = "", -- CPU: 60 Core: 12 Thread:1 Node: 0
[ 61] = "", -- CPU: 61 Core: 13 Thread:1 Node: 0
[ 62] = "", -- CPU: 62 Core: 14 Thread:1 Node: 0
[ 63] = "", -- CPU: 63 Core: 15 Thread:1 Node: 0
[ 64] = "", -- CPU: 64 Core: 16 Thread:1 Node: 0
[ 65] = "", -- CPU: 65 Core: 17 Thread:1 Node: 0
[ 66] = "", -- CPU: 66 Core: 18 Thread:1 Node: 0
[ 67] = "", -- CPU: 67 Core: 19 Thread:1 Node: 0
[ 68] = "", -- CPU: 68 Core: 20 Thread:1 Node: 0
[ 69] = "", -- CPU: 69 Core: 21 Thread:1 Node: 0
[ 70] = "", -- CPU: 70 Core: 22 Thread:1 Node: 0
[ 71] = "", -- CPU: 71 Core: 23 Thread:1 Node: 0
[ 72] = "fmadio", -- CPU: 72 Core: 0 Thread:1 Node: 1 system
[ 73] = "", -- CPU: 73 Core: 1 Thread:1 Node: 1
[ 74] = "", -- CPU: 74 Core: 2 Thread:1 Node: 1
[ 75] = "", -- CPU: 75 Core: 3 Thread:1 Node: 1
[ 76] = "fmadio", -- CPU: 76 Core: 4 Thread:1 Node: 1 system
[ 77] = "fmadio", -- CPU: 77 Core: 5 Thread:1 Node: 1 system
[ 78] = "fmadio", -- CPU: 78 Core: 6 Thread:1 Node: 1 system
[ 79] = "fmadio", -- CPU: 79 Core: 7 Thread:1 Node: 1 system
[ 80] = "", -- CPU: 80 Core: 8 Thread:1 Node: 1
[ 81] = "", -- CPU: 81 Core: 9 Thread:1 Node: 1
[ 82] = "", -- CPU: 82 Core: 10 Thread:1 Node: 1
[ 83] = "", -- CPU: 83 Core: 11 Thread:1 Node: 1
[ 84] = "", -- CPU: 84 Core: 12 Thread:1 Node: 1
[ 85] = "", -- CPU: 85 Core: 13 Thread:1 Node: 1
[ 86] = "", -- CPU: 86 Core: 14 Thread:1 Node: 1
[ 87] = "", -- CPU: 87 Core: 15 Thread:1 Node: 1
[ 88] = "", -- CPU: 88 Core: 16 Thread:1 Node: 1
[ 89] = "", -- CPU: 89 Core: 17 Thread:1 Node: 1
[ 90] = "", -- CPU: 90 Core: 18 Thread:1 Node: 1
[ 91] = "", -- CPU: 91 Core: 19 Thread:1 Node: 1
[ 92] = "", -- CPU: 92 Core: 20 Thread:1 Node: 1
[ 93] = "", -- CPU: 93 Core: 21 Thread:1 Node: 1
[ 94] = "", -- CPU: 94 Core: 22 Thread:1 Node: 1
[ 95] = "", -- CPU: 95 Core: 23 Thread:1 Node: 1
```

```
},
```

CPUS which are dedicated to FMADIO Packet Capture system can not modified, these are marked "System" in the comments section per below

```
17] = "" , -- CPU: 17 Core: 17 Thread:0 Node: 0
18] = "" , -- CPU: 18 Core: 18 Thread:0 Node: 0
19] = "" , -- CPU: 19 Core: 19 Thread:0 Node: 0
20] = "" , -- CPU: 20 Core: 20 Thread:0 Node: 0
21] = "" , -- CPU: 21 Core: 21 Thread:0 Node: 0
22] = "" , -- CPU: 22 Core: 22 Thread:0 Node: 0
23] = "" , -- CPU: 23 Core: 23 Thread:0 Node: 0
24] = "fmadio", -- CPU: 24 Core: 0 Thread:0 Node: 1 system
25] = "fmadio", -- CPU: 25 Core: 1 Thread:0 Node: 1 system
26] = "fmadio", -- CPU: 26 Core: 2 Thread:0 Node: 1 system
27] = "fmadio", -- CPU: 27 Core: 3 Thread:0 Node: 1 system
28] = "fmadio", -- CPU: 28 Core: 4 Thread:0 Node: 1 system
29] = "fmadio", -- CPU: 29 Core: 5 Thread:0 Node: 1 system
30] = "fmadio", -- CPU: 30 Core: 6 Thread:0 Node: 1 system
31] = "fmadio", -- CPU: 31 Core: 7 Thread:0 Node: 1 system
32] = "" , -- CPU: 32 Core: 8 Thread:0 Node: 1
33] = "" , -- CPU: 33 Core: 9 Thread:0 Node: 1
34] = "" , -- CPU: 34 Core: 10 Thread:0 Node: 1
35] = "" , -- CPU: 35 Core: 11 Thread:0 Node: 1
```

FMADIO System CPUs

Any change to system CPUs will be overwritten.

### Enable Custom CPU Mapping

By default the system allocates all CPUs to the FMADIO Capture System. To enable a customer CPU Mapping change Enable = true per the below setting

```
["Enable"] = true,
```

After enabling a FW update is required. As the enable re-writes the isolcpu Linux kernel boot parameters.

NOTE: after Enable = true is set, all subsequent firmware updates will modify the isolcpu setting. To return to stock default setting. Set to false, and update the firmware again.

### Assign a CPU

To assign a CPU to a specific LXC container, use the naming convention

```
container.process.instance
```

For example Market Data Gap detector LXC (mdgap) for CME is allocated to CPU 84

```

.
.
[ 77] = "fmadio",           -- CPU: 77 Core: 5 Thread:1 Node: 1 system
[ 78] = "fmadio",           -- CPU: 78 Core: 6 Thread:1 Node: 1 system
[ 79] = "fmadio",           -- CPU: 79 Core: 7 Thread:1 Node: 1 system
[ 80] = "",                 -- CPU: 80 Core: 8 Thread:1 Node: 1
[ 81] = "",                 -- CPU: 81 Core: 9 Thread:1 Node: 1
[ 82] = "",                 -- CPU: 82 Core: 10 Thread:1 Node: 1
[ 83] = "",                 -- CPU: 83 Core: 11 Thread:1 Node: 1
[ 84] = "mdgap.cme.0",      -- CPU: 84 Core: 12 Thread:1 Node: 1
[ 85] = "",                 -- CPU: 85 Core: 13 Thread:1 Node: 1
[ 86] = "",                 -- CPU: 86 Core: 14 Thread:1 Node: 1
[ 87] = "",                 -- CPU: 87 Core: 15 Thread:1 Node: 1
.
.

```

This also requires re-running the LXC install script to correctly allocate the CPUs in the container.

### Update CPU Mapping

After the configuration files have been updated, a firmware update and reboot cycle are required. Its required as the Linux kernel boot command parameters (isolcpus) gets modified requiring the reboot.

Start by finding the current Firmware binary, in almost all cases this is the last upload firmware

```

fmadio@fmadio20v3-287:~$ cd /mnt/system/firmware/
fmadio@fmadio20v3-287:/mnt/sda1/firmware$ ls -ltr
total 2554288
drwxr-xr-x  7 root    root      8192 Jan  1  1970 ./
-rwxr-xr-x  1 root    root    107442176 Oct 13 13:16 fmadio20v3_20221013_1314.tcz
-rwxr-xr-x  1 root    root      820 Oct 13 13:16 fmadio20v3_20221013_1314.syslinux.c
-rwxr-xr-x  1 root    root     1425 Oct 13 13:16 fmadio20v3_20221013_1314.syslinux.c
-rwxr-xr-x  1 root    root      868 Oct 13 13:16 fmadio20v3_20221013_1314.sign
-rwxr-xr-x  1 root    root    98846660 Oct 13 13:16 fmadio20v3_20221013_1314.rom.2x1G
-rwxr-xr-x  1 root    root    98683200 Oct 13 13:16 fmadio20v3_20221013_1314.rom.2x10G
-rwxr-xr-x  1 root    root      87 Oct 13 13:16 fmadio20v3_20221013_1314.pre.lua
-rwxr-xr-x  1 root    root     754 Oct 13 13:16 fmadio20v3_20221013_1314.post.lua
-rwxr-xr-x  1 root    root    21356 Oct 13 13:16 fmadio20v3_20221013_1314.mydata
-rwxr-xr-x  1 root    root    4632672 Oct 13 13:16 fmadio20v3_20221013_1314.kernel
-rwxr-xr-x  1 root    root   312308358 Oct 13 13:16 fmadio20v3_20221013_1314.core
-rwxr-xr-x  1 root    root   427477506 Oct 13 13:16 fmadio20v3_20221013_1314.bin
-rwxr-xr-x  1 root    root   427590054 Oct 21 10:28 fmadio20v3_20221021_1026.bin
-rwxr-xr-x  1 root    root   107548672 Oct 21 10:28 fmadio20v3_20221021_1026.tcz
-rwxr-xr-x  1 root    root      820 Oct 21 10:28 fmadio20v3_20221021_1026.syslinux.c
-rwxr-xr-x  1 root    root     1425 Oct 21 10:28 fmadio20v3_20221021_1026.syslinux.c
-rwxr-xr-x  1 root    root      868 Oct 21 10:28 fmadio20v3_20221021_1026.sign
-rwxr-xr-x  1 root    root    98846660 Oct 21 10:28 fmadio20v3_20221021_1026.rom.2x1G
-rwxr-xr-x  1 root    root    98683200 Oct 21 10:28 fmadio20v3_20221021_1026.rom.2x10G
-rwxr-xr-x  1 root    root      87 Oct 21 10:28 fmadio20v3_20221021_1026.pre.lua
-rwxr-xr-x  1 root    root     754 Oct 21 10:28 fmadio20v3_20221021_1026.post.lua
-rwxr-xr-x  1 root    root    21356 Oct 21 10:28 fmadio20v3_20221021_1026.mydata
-rwxr-xr-x  1 root    root    4632672 Oct 21 10:28 fmadio20v3_20221021_1026.kernel
-rwxr-xr-x  1 root    root   312308358 Oct 21 10:28 fmadio20v3_20221021_1026.core
-rwxr-xr-x  1 root    root   107548672 Oct 21 11:27 fmadio20v3_20221021_1125.tcz
-rwxr-xr-x  1 root    root      820 Oct 21 11:27 fmadio20v3_20221021_1125.syslinux.c
-rwxr-xr-x  1 root    root     1425 Oct 21 11:27 fmadio20v3_20221021_1125.syslinux.c
-rwxr-xr-x  1 root    root      868 Oct 21 11:27 fmadio20v3_20221021_1125.sign
-rwxr-xr-x  1 root    root    98846660 Oct 21 11:27 fmadio20v3_20221021_1125.rom.2x1G
-rwxr-xr-x  1 root    root    98683200 Oct 21 11:27 fmadio20v3_20221021_1125.rom.2x10G
-rwxr-xr-x  1 root    root      87 Oct 21 11:27 fmadio20v3_20221021_1125.pre.lua
-rwxr-xr-x  1 root    root     754 Oct 21 11:27 fmadio20v3_20221021_1125.post.lua
-rwxr-xr-x  1 root    root    21356 Oct 21 11:27 fmadio20v3_20221021_1125.mydata
-rwxr-xr-x  1 root    root    4632672 Oct 21 11:27 fmadio20v3_20221021_1125.kernel
-rwxr-xr-x  1 root    root   312308358 Oct 21 11:27 fmadio20v3_20221021_1125.core
-rwxr-xr-x  1 root    root   427587155 Oct 21 11:27 fmadio20v3_20221021_1125.bin
drwxr-xr-x  2 root    root     40960 Oct 21 11:27 ./
fmadio@fmadio20v3-287:/mnt/sda1/firmware$

```

In this case the latest firmware is named

```
fmadio20v3_20221021_1125.bin
```

Next re-install the firmware as follows. An upload is not required as the firmware is already on the system

```
sudo firmware_install.lua --install fmadio20v3_20221021_1125.bin
```

Example output as follows, note the "Update CPU Map" output

```
fmadio@fmadio20v3-287:/mnt/sda1/firmware$ sudo firmware_install.lua --install fmadio20v3_202
fmad fmadlua Oct 21 2022 (/usr/local/bin/fmadiolua /opt/fmadio/bin/firmware_install.lua --in
calibrating...
0 : 2095063152          2.0951 cycles/nsec offset:4.937 Mhz
Cycles/Sec 2095063152.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
Install Firmware [fmadio20v3_20221021_1125.bin]
Open fSysCapture_t* [/opt/fmadio/status/capture:0/100]
Force capture to exit
Force capture to exit
Force capture to exit
Force capture to exit
Force capture to exit
Force capture to exit
Force capture to exit
FW [fmadio20v3] System[fmadio20v3]
PreInstall script [/mnt/system//firmware/fmadio20v3_20221021_1125.pre.lua]
fmad fmadlua Oct 21 2022 (/opt/fmadio/bin/fmadiolua /mnt/system//firmware/fmadio20v3_2022102
calibrating...
0 : 2095065834          2.0951 cycles/nsec offset:4.934 Mhz
Cycles/Sec 2095065834.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
done 0.000043Sec 0.000001Min
PreInstall: pre install script
PreInstall: PREINSTALL_GOOD
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.tcz /mnt/system//tce/optional/fmadio
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.core /mnt/system//boot/fmadio20v3-co
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.kernel /mnt/system//boot/vmlinuz64]
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.mydata /mnt/system//tce/mydata.tgz]
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.syslinux.cfg /mnt/system//boot/syqli
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.rom.2x1G /mnt/system//boot/bitstrea
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.rom.2x10G /mnt/system//boot/bitstrea
Bitstream Config [2x10G]
Copy [cp /mnt/system//firmware/fmadio20v3_20221021_1125.rom.2x10G /mnt/system//boot/bitstrea
os[sudo /opt/fmadio/bin/bitstream_update.lua --noreboot --write /mnt/system//boot/bitstream.
fmad fmadlua Oct 21 2022 (/opt/fmadio/bin/fmadiolua /opt/fmadio/bin/bitstream_update.lua --n
calibrating...
0 : 2095065974          2.0951 cycles/nsec offset:4.934 Mhz
Cycles/Sec 2095065974.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
cp: '/mnt/system//boot/bitstream.rom' and '/mnt/system//boot/bitstream.rom' are the same fil
done 0.004990Sec 0.000083Min
Updating CPU Map
renaming syslinux.cfg files
/mnt/system//firmware/fmadio20v3_20221021_1125.post.lua
fmad fmadlua Oct 21 2022 (/opt/fmadio/bin/fmadiolua /mnt/system//firmware/fmadio20v3_2022102
calibrating...
0 : 2095064668          2.0951 cycles/nsec offset:4.935 Mhz
Cycles/Sec 2095064668.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
PostInstall: post install script
done 0.001261Sec 0.000021Min
PostInstall: POSTINSTALL_GOOD
Firmware Install Complete
done 36.463587Sec 0.607726Min
fmadio@fmadio20v3-287:/mnt/sda1/firmware$
```

Then reboot the system, and the new CPU mapping is reflected.

```
sudo reboot
```

# Scratch Disk

Scratch Disks are helpful for keeping temporary or persistent data outside of the FMADIO host capture system. Many times the OS disks is unable to be used due to capacity or location.

Scratch Disks can be run as a RAID0 or RAIDx group of drives, allow you the system flexible allocation of the total storage capacity of the system between Capture vs Analytics.

## ScratchDisk Configuration

### Step 1) Assign physical disks

Start by editing the disk configuration file

```
/opt/fmadio/etc/disk.lua
```

Then moving serial numbers from the CacheDisk section to the ScratchDisk section.

For example moving a single disk (serial number "22443E9DC54E") from `ssd8` -> `scr0`

Before:



```
CacheDisk =
{
  ["22443E9D204F"] = "ssd0",
  ["22223AD5BFC3"] = "ssd1",
  ["22443E9D3AFF"] = "ssd2",
  ["22443E9DC543"] = "ssd3",
  ["22443E9D2076"] = "ssd4",
  ["22443E9D3B41"] = "ssd5",
  ["22443E9D3B65"] = "ssd6",
  ["22443E9D20A4"] = "ssd7",
  ["22443E9DC54E"] = "ssd8",
}
,
ParDisk =
{
  ["22443E9D2087"] = "par0",
}
,
RaidDisk =
{
}
,
ScratchDisk =
{
}
,
OSDisk =
{
  ["50026B7685513F33"] = "os0",
}
,
```

renaming ssd8 to scr0

After:

```

CacheDisk =
{
    ["22443E9D204F"]      = "ssd0",
    ["22223AD5BFC3"]     = "ssd1",
    ["22443E9D3AFF"]     = "ssd2",
    ["22443E9DC543"]     = "ssd3",
    ["22443E9D2076"]     = "ssd4",
    ["22443E9D3B41"]     = "ssd5",
    ["22443E9D3B65"]     = "ssd6",
    ["22443E9D20A4"]     = "ssd7",
}
,
ParDisk =
{
    ["22443E9D2087"]     = "par0",
}
,
RaidDisk =
{
}
,
ScratchDisk =
{
    ["22443E9DC54E"]     = "scr0",
}
,
OSDisk =
{
    ["50026B7685513F33"] = "os0",
}

```

Save the file.

This can be repeated for as many disks as you require, please keep the numbering sequential e.g. scr0, scr1, scr2, scr3 etc so the system can map it correctly.

### Step 2) Reboot the system

Rebooting is required as the system needs to rename the mount points in /opt/fmadio/disk

### Step 3) Create RAID partition

Create the RAID0 partition using mdadm as follows. Change the appropriate fields to create a RAID0 2 disk partition or more.

```
sudo mdadm --create /dev/md1 --force --verbose --level=raid0 --raid-devices=1 /opt/fmadio/di
```

Or for a 2 disk Scratch Partition

```
sudo mdadm --create /dev/md1 --force --verbose --level=raid0 --raid-devices=2 /opt/fmadio/di
```

The output will look something like below

```
fmadio@fmadio100p3-539:/mnt/store0/develop/system$ sudo mdadm --create /dev/md1 --force --verbose --level=raid0 --raid-devices=1 /opt/fmadio/disk/scr0
mdadm: chunk size defaults to 512K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md1 started.
fmadio@fmadio100p3-539:/mnt/store0/develop/system$
```

Create RAID0 Scratch Drive

### Step 3) Create Filesystem

The partition now exists on /dev/md1 we need to create a filesystem next so it can be used in a general purpose way. Run the following command to create an EXT4 based Scratch Disk partition

```
sudo mkfs.ext4 /dev/md1
```

Example output looks like below

```
fmadio@fmadio100p3-539:/mnt/store0/develop/system$ sudo mkfs.ext4 /dev/md1
mke2fs 1.46.5 (30-Dec-2021)
Discarding device blocks: done
Creating filesystem with 937651712 4k blocks and 234414080 inodes
Filesystem UUID: 5a5c49d6-e87c-47fb-86df-ceFef5e2b25f
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
    102400000, 214990848, 512000000, 550731776, 644972544

Allocating group tables: done
Writing inode tables: done
Creating journal (262144 blocks): done
Writing superblocks and filesystem accounting information: done
fmadio@fmadio100p3-539:/mnt/store0/develop/system$
```

### Step 4) Confirm the filesystem mounts correct

Next confirm th file system mounts correctly, by creating a test file on the partition

```
sudo mount /dev/md1 /mnt/store1
```

Then creating a file and confirming, such as below

```
fmadio@fmadio100p3-539:/mnt/store0/develop/system$ sudo touch /mnt/store1/test
fmadio@fmadio100p3-539:/mnt/store0/develop/system$ ls -al /mnt/store1/
total 20
drwxr-xr-x  3 root  root    4096 Sep  7 20:01 ./
drwxr-xr-x 10 root  root    220 Sep  7 19:38 ../
drwx-----  2 root  root   16384 Sep  7 19:57 lost+found/
-rw-r--r--  1 root  root     0 Sep  7 20:01 test
fmadio@fmadio100p3-539:/mnt/store0/develop/system$
```

### Step 5) Confirm automatic mounting

Reboot the system and confirm the ScratchDisk gets mounted automatically on boot to

```
/mnt/store1
```

The test file created above should exist, example below

```
fmadio@fmadio100p3-539:~$ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
nbd0         43:0     0    0B  0 disk
nbd1         43:32    0    0B  0 disk
nbd2         43:64    0    0B  0 disk
nbd3         43:96    0    0B  0 disk
nbd4         43:128   0    0B  0 disk
nbd5         43:160   0    0B  0 disk
nbd6         43:192   0    0B  0 disk
nbd7         43:224   0    0B  0 disk
zram0        251:0    0    0B  0 disk
nvme0n1      259:0    0 476.9G 0 disk
|-nvme0n1p1  259:8    0  14.9G 0 part /mnt/nvme0n1p1
|-nvme0n1p2  259:9    0 462G   0 part /mnt/store0
nvme1n1      259:1    0   3.5T 0 disk
nvme5n1      259:2    0   3.5T 0 disk
nvme10n1     259:3    0   3.5T 0 disk
nvme4n1      259:4    0   3.5T 0 disk
nvme3n1      259:5    0   3.5T 0 disk
nvme9n1      259:6    0   3.5T 0 disk
nvme8n1      259:7    0   3.5T 0 disk
|-md1         9:1     0   3.5T 0 raid0 /mnt/store1
|-nvme6n1     259:10   0   3.5T 0 disk
nvme2n1      259:11   0   3.5T 0 disk
nvme7n1      259:12   0   3.5T 0 disk
nbd8         43:256   0    0B  0 disk
nbd9         43:288   0    0B  0 disk
nbd10        43:320   0    0B  0 disk
nbd11        43:352   0    0B  0 disk
nbd12        43:384   0    0B  0 disk
nbd13        43:416   0    0B  0 disk
nbd14        43:448   0    0B  0 disk
nbd15        43:480   0    0B  0 disk
fmadio@fmadio100p3-539:~$ cd /mnt/store1
fmadio@fmadio100p3-539:/mnt/store1$ ls
lost+found test
fmadio@fmadio100p3-539:/mnt/store1$
```

## Capture Metadata on ScratchDisk

In some cases its best to locate the Capture Filesystems Metadata onto the scratch disk. A few extra steps in addition to the above steps are required.

### Step 1) Create stream directory

Create a stream directory on the ScratchDisk such as below

```
sudo mkdir stream
```

Example output looks like the following

```
fmadio@fmadio100p3-539:/mnt/store1$ sudo mkdir stream
fmadio@fmadio100p3-539:/mnt/store1$ ls -al stream
total 8
drwxr-xr-x  2 root  root    4096 Sep  7 20:26 ./
drwxr-xr-x  4 root  root    4096 Sep  7 20:26 ../
fmadio@fmadio100p3-539:/mnt/store1$
```

### Step 2) set location of the Capture systems metadata.

Edit the file

```
/opt/fmadio/etc/disk.lua
```

Near the end of the file create a one line entry as follows

```
StreamPath = "/mnt/store1/stream/",
```

Example output looks like

```
return
CacheDisk =
{
  ["22443E9D204F"] = "ssd0",
  ["22223AD58FC3"] = "ssd1",
  ["22443E9D3AFF"] = "ssd2",
  ["22443E9DC543"] = "ssd3",
  ["22443E9D2076"] = "ssd4",
  ["22443E9D3B41"] = "ssd5",
  ["22443E9D3B65"] = "ssd6",
  ["22443E9D20A4"] = "ssd7",
}
ParDisk =
{
  ["22443E9D2087"] = "par0",
}
RaidDisk =
{
}
ScratchDisk =
{
  ["22443E9DC54E"] = "scr0",
}
OSDisk =
{
  ["5002687685513F33"] = "os0",
}
IndexDisk = "ssd",
CacheLevel = "full",
RaidLevel = "raid0",
StreamPath = "/mnt/store1/stream/",
["PSID"] =
{
  ["22443E9D204F"] = "7306CD8CE0BE5F694793F8D8409E7CFE",
  ["22223AD58FC3"] = "75983A185E65017D81991352F44A6852",
  ["22443E9D3AFF"] = "1DC0C7EAS719926B9F3984445F856A03",
  ["22443E9DC543"] = "882988DE45F10E67A1F789D7DB890F63",
  ["22443E9D2076"] = "1E01C0112EF6587BACE8D02B96540F1C",
  ["22443E9D3B41"] = "D60FCE577389D409293E56A56BA23D25",
  ["22443E9D3B65"] = "EB48843D4CE8AED46EE257DA19186F32",
  ["22443E9D20A4"] = "5D8445E23592823E8B98513C28130C4C",
  ["22443E9DC54E"] = "8287AD621B440D756AB1D1E8273C6F94",
  ["22443E9D2087"] = "11EDDF5767CA7C0B8C0FD8643803B657",
}
},
```

### Step 3) Reboot the system

Reboot the system

### Step 4) Confirm stream path is pointing to the correct location

Run the command

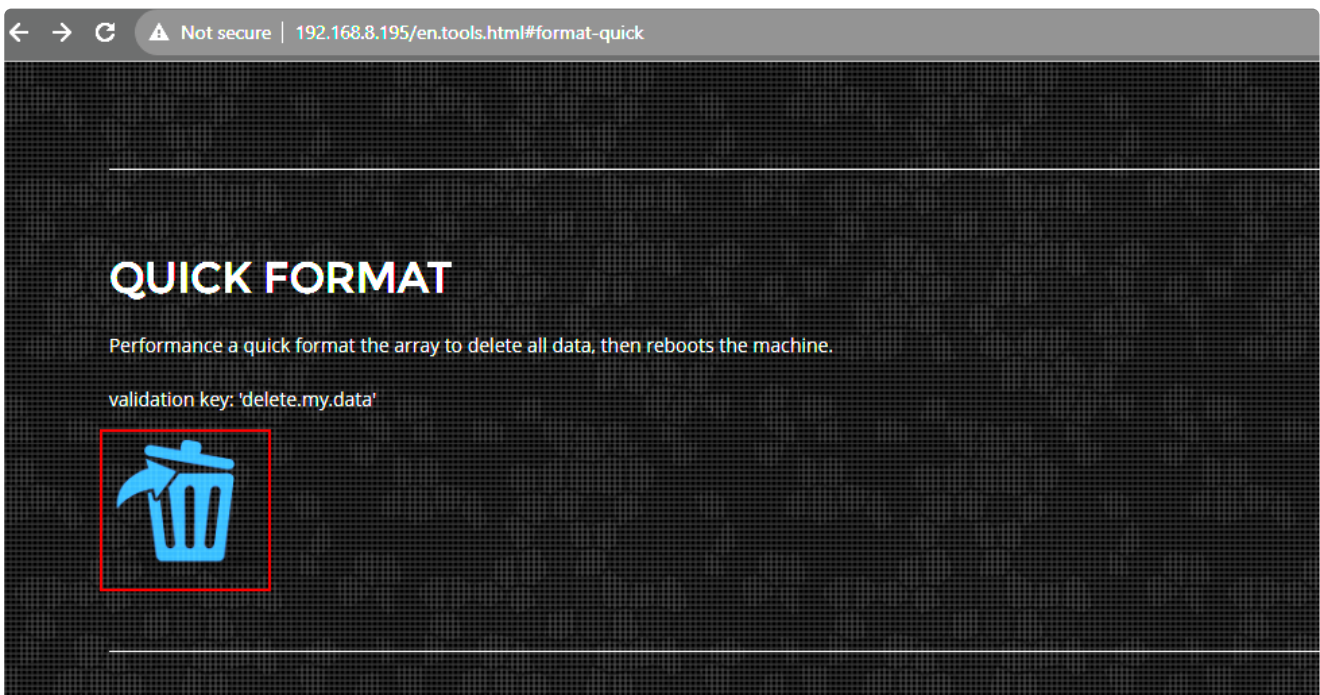
```
ls -al /opt/fmadio/
```

Confirm the /opt/fmadio/stream symbolic link is pointing to the correct location, example below.

```
fmadio@fmadio100p3-539:~$ ls -al /opt/fmadio/
total 8
drwxr-xr-x 12 fmadio staff 400 Sep 7 20:32 ./
drwxr-xr-x 3 fmadio staff 180 Aug 13 18:57 ../
drwxr-xr-x 2 fmadio staff 180 Aug 13 18:57 analytics/
drwxr-xr-x 2 fmadio staff 2500 Aug 13 18:57 bin/
-rw-r--r-- 1 root root 12 Jun 22 13:42 device_name
drwxr-xr-x 2 root root 400 Sep 7 20:32 disk/
drwxr-xr-x 12 fmadio staff 260 Sep 7 20:32 drivers/
lrwxrwxrwx 1 root root 15 Jun 17 17:46 etc -> /mnt/store0/etc/
drwxr-xr-x 2 fmadio staff 260 Aug 13 18:57 etc_ro/
lrwxrwxrwx 1 root root 18 Sep 7 20:32 filter -> /mnt/store0/filter/
drwxr-xr-x 3 fmadio staff 240 Aug 13 18:57 firmware/
drwxr-xr-x 2 fmadio staff 320 Aug 13 18:57 include/
lrwxrwxrwx 1 root root 23 Jul 9 18:04 lxc -> /mnt/store0/lxc/lib/lxc/
lrwxrwxrwx 1 root root 23 Jul 9 18:05 plugin_data -> /mnt/store0/plugin_data
drwxr-xr-x 2 root root 40 Jul 9 20:15 queue/
-rw-r--r-- 1 root root 25 Sep 7 20:32 serial
drwxr-xr-x 2 root root 240 Sep 7 20:32 status/
lrwxrwxrwx 1 root root 19 Sep 7 20:32 stream -> /mnt/store1/stream//
lrwxrwxrwx 1 root root 50 Sep 7 20:31 version -> /tmp/cc100p/fmadio100p3_current
drwxr-xr-x 9 fmadio staff 820 Aug 13 18:57 www/
fmadio@fmadio100p3-539:~$
```

### Step 5) Quick Format

Run a quick format via the GUI to reset the storage partition. This changes the number of drives used for the capture file system.



The above will take about 5 minutes and reboot 2 times by itself.

### Step 6) Confirm filesystem metadata is on the ScratchDisk

Check the directory listing

```
ls -al /mnt/store1/stream/
```

Example looks as follows

```
fmadio@fmadio100p3-539:~$ ls -al /mnt/store1/stream/
total 14046888
drwxr-xr-x  2 root   root    4096 Sep  7 20:39 ./
drwxr-xr-x  4 root   root    4096 Sep  7 20:26 ../
-rw-rw-rw-  1 root   root  2813053632 Sep  7 20:39 cache_index.block
-rw-rw-rw-  1 root   root 3047469568 Sep  7 20:39 chunk_index.bin
-rw-rw-rw-  1 root   root 134217728 Sep  7 20:39 chunk_index.fifo
-rw-rw-rw-  1 root   root 3516311040 Sep  7 20:39 chunk_packet.bin
-rw-rw-rw-  1 root   root 2813048832 Sep  7 20:39 chunk_time.bin
-rw-rw-rw-  1 root   root 134217728 Sep  7 20:39 chunk_write.fifo
-rw-rw-rw-  1 root   root 1073745920 Sep  7 20:39 store_ecc.fifo
-rw-rw-rw-  1 root   root 703262208 Sep  7 20:39 store_index.block
-rw-rw-rw-  1 root   root 131861664 Sep  7 20:39 store_index.crc32
-rw-rw-rw-  1 root   root    1024 Sep  7 20:39 stream_ctrl.bin
-rw-rw-rw-  1 root   root 16777216 Sep  7 20:39 stream_index.bin
fmadio@fmadio100p3-539:~$
```

### Step 7) Finished

After this the file system can be moved along with the diskpacks.

# Settings



# Capture

## Advanced Capture Settings

Advanced capture settings require changing the configuration file directly as follows. As many of these options are customizations, please discuss with [support@fmad.io](mailto:support@fmad.io) any issues or side effects seen

```
/opt/fmadio/etc/time.lua
```

## Validate Config file

After modifying time.lua configuration file, please confirm syntax is correct by running the following

```
fmadiolua /opt/fmadio/etc/time.lua
```

The correct output looks like the following

```
fmadio@fmadio100v2-228U:~$ fmadiolua /opt/fmadio/etc/time.lua
fmad fmadlua Jun  3 2022 (fmadiolua /opt/fmadio/etc/time.lua )
calibrating...
0 : 2095078948          2.0951 cycles/nsec offset:4.921 Mhz
Cycles/Sec 2095078948.0000 Std:          0 cycle std( 0.000000000) Target:2.10 Ghz
failed to open self? [fmadiolua]
done 0.000110Sec 0.000002Min
fmadio@fmadio100v2-228U:~$
```

## Midnight Roll Disable

By default the capture system will automatically stop all captures at midnight per the local timezone set on the capture device. This can be disabled which is helpful for capture periods which are < 24H but cross the midnight boundary.

In the `["Scheduler"]` section

```
["Scheduler"] =
{
  ["ManualOffset"] = 0,
},
```

Add the entry set to true, to disable the Midnight roll

```
["ManualOffset"] = 1,
```

The config requires capture to be stopped and started for the the new setting to load.

### Capturing Across Midnight

In some cases capturing across midnight and rolling at a different time has advantages, such as Futures/Options exchanges who typically run 23H a day restarting very early in the morning.

This is achieved using the ManualOffset to offset when the capture rolls. To roll the capture as 5AM every day, setting 5 H \* 60min \* 60sec \* 1e9 nanoseconds

```
["ManualOffset"] = 5 * 60 * 60 * 1e9,
```

Then setting the scheduler to start/stop at 00:00:59 and 23:59:59 as follows

**CAPTURE SCHEDULE**  
Lists currently active capture schedule. Click Plus icon to add a new capture start/end time. Alternatively the X icon to remove a scheduling entry. Note that times are relative to the current time.

[Wed Jun 15 11:15:18 2022] scheduler(wan\_colo0) updated day of week (Sun) enables (true)

Delete	24/7	Start Time	Stop Time	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Capture Name
<input checked="" type="checkbox"/>	<input type="checkbox"/>	00:00:00	23:59:59	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	wan_colo0

This will create captures from 05:00AM until 04:59:59 the next day

StartTime = ManualOffset (5H) + Scheduler Start Time 00:00 = 05:00 AM

StopTime = ManualOffset (5H) + Scheduler Stop Time 23:59:59 = 04:59AM + 1 Day

# Download

Various settings related to downloading PCAP.

## API Version 1 Enable

Depending on the packet capture system the Download interface may be in Legacy mode. This modes functionality is limited in the range of API calls, specifically `/api/*` calls can not be used

To enable API Version 1, edit the configuration file

```
/opt/fmadio/etc/time.lua
```

Find the section ["PCAP"] example as below

```
["PCAP"] =
{
  ["TimeStampMode"]      = "nic",
  ["TimeResolution"]    = "nsec",
  ["TimeSortDepth"]     = 256,
  ["Decap"]              = false,
  ["Slice"]              = 0,
  ["DownloadIdleTimeout"] = 30000000000,
  ["DownloadAPI"]       = "legacy",
  ["FShark"]             = true,
},
```

Please change ["DownloadAPI"]="legacy" to "v1" as below. If theres no entry, create the entry.

```
["DownloadAPI"]      = "v1",
```

It requires a browser refresh for the setting to become active.

## Time Stamp Selection

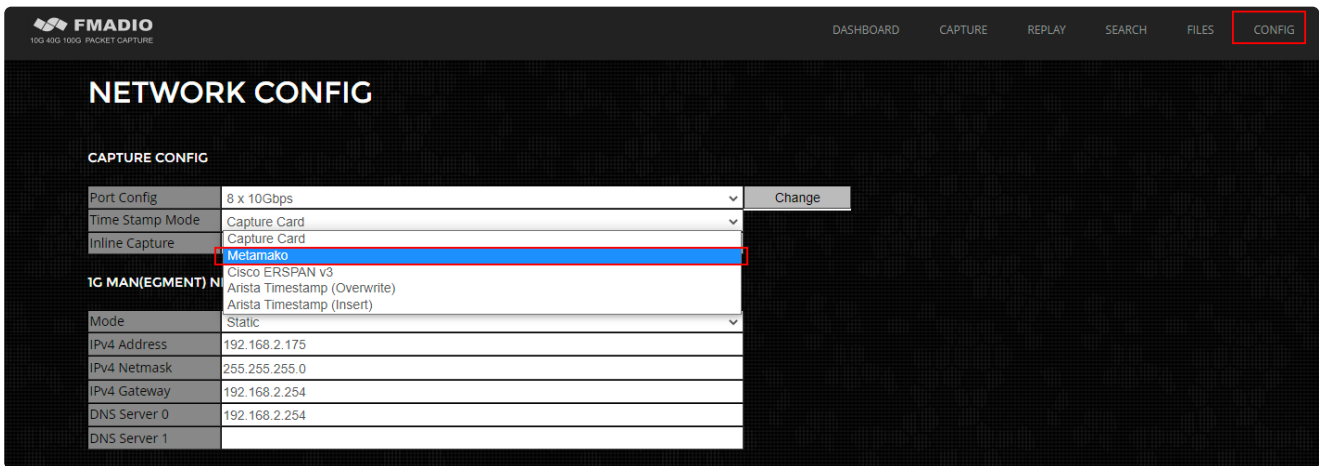
FMADIO Devices use our internal FPGA NIC card for hardware timestamping on all packets. This may not be sufficient for some customers as Tap/Aggregation layers add additional non-deterministic latency due to egress port buffering.

FMADIO Packet Capture has the option to use Packet Brokers and Switch's metadata either in headers or footers. This allows ingress timestamping at the Tap/Agg layer plus gives additional port visibility and filtering capabilities.

By default FMADIO uses our internal hardware timestamps when downloading PCAPs. We support the following formats

- FMADIO Capture Card (Default)
- Cisco ERSPAN v3
- Arista 7130(Metamako)
- Arista 7150 (Overwrite mode)
- Arista 7150 (Insert Mode)

Use the Config page on the FMADIO Capture system to select as follows



FMADIO Timestamp Selection Mode

NOTE: This can be done retro-actively. e.g. Downloaded PCAPs are generated based on the timestamp selection mode. This setting does not effect the raw captured data.

## Packet De-encapsulation

FMADIO System can de-encapsulate various formats both in PacketScope, Search and via the API. Please contact support if you require an additional format.

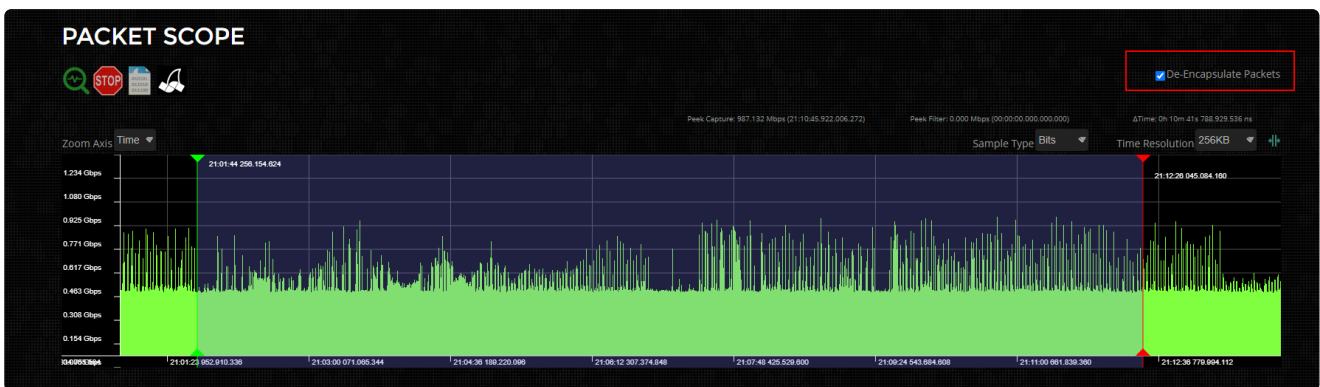
De-encapsulation means the system will strip away the encapsulation, and BPF filters are run on the inner most packet.

## Supported De-Encapsulation

- VLAN Single Tag
- VLAN Double Tag Old style (Ether Proto 0x9100)
- VLAN Double Tag Old style (Ether Proto 0x9200)
- VLAN Double Tag QinQ (Ether Proto 0x88a8)
- VNTag Cisco (Ether Proto 0x8926)
- VXLAN (Tunnel over UDP)
- MPLS Single
- MPLS Double
- MPLS Triple
- Cisco ERSPAN v1
- Cisco ERSPAN v2 (Over GRE tunnel)
- Cisco ERSPAN v3 (Over GRE tunnel)
- CAPWAP (Wireless Encapsulation over UDP)
- Arista 7150 (Insert Mode)
- Arista 7150 (Overwrite Mode)
- Arista 7130 (Metamako)
- Ixia xStream
- Exablaze Footer

## PacketScope De-encapsulation

By default PacketScope enables the de-encapsulation shown below as the checkbox enabled. To disable de-encapsulation uncheck the checkbox and re-run the filters.



PacketScope De-Encapsulation

# User

## Advanced User Settings

Advanced capture settings require changing the configuration file directly as follows. As many of these options are customizations, please discuss with [support@fmad.io](mailto:support@fmad.io) any issues or side effects seen

```
/opt/fmadio/etc/time.lua
```

### Validate Config file

After modifying time.lua configuration file, please confirm syntax is correct by running the following

```
fmadiolua /opt/fmadio/etc/time.lua
```

The correct output looks like the following

```
fmadio@fmadio100v2-228U:~$ fmadiolua /opt/fmadio/etc/time.lua
fmad fmadlua Jun  3 2022 (fmadiolua /opt/fmadio/etc/time.lua )
calibrating...
0 : 2095078948          2.0951 cycles/nsec offset:4.921 Mhz
Cycles/Sec 2095078948.0000 Std:          0 cycle std( 0.000000000) Target:2.10 Ghz
failed to open self? [fmadiolua]
done 0.000110Sec 0.000002Min
fmadio@fmadio100v2-228U:~$
```

### Disable Web Admin Permission

By default the Web user account has full permissions. There may be cases where Web access is for download and analyzing PCAP data only. Where configuration can be not modified.

To disable Web Admin mode SSH into the system and change the setting in

```
/opt/fmadio/etc/time.lua
```

In the ["Security"] section

```
["Security"] =
{
  ["HTTPAccess"] = "enabled",
  ["Auth"] = "BASIC",
  ["ConfigAccess"] = "full",

  ["GUIMode"] = "full",
  ["EnableWebDAV"] = false,
  ["RADIUS_Secret"] = "nil",
  ["RADIUS_Host"] = "nil",
  ["RADIUS_Protocol"] = "nil",
  ["RADIUS_Timeout"] = nil,
},
```

Change ConfigAccess setting to be "user"

```
["ConfigAccess"] = "user",
```

The config file should be checked for syntax errors. Then restart the system. Once restarted the Web GUI no longer has Admin privileges'.

Operation



# Performance Download

Download performance examples using the CLI interface to extract PCAP data to downstream systems.

Please use "stream\_dump" to list all captures which can be used with

```
sudo stream_cat -v <capture name> | ....
```

the output of stream\_cat is a standard PCAP via the stdout pipe

NOTE: throughput is heavily dependent on the packet size mix of the capture. Lager packet sizes get a better over all throughput

Transport	Physical Interface	64B Packet	9200B Jumbo Packet	IMIX Packet
Local File System	local		8.8Gbps	
Local File System (rclone)	local		2.4Gbps	
NFS Remote	10G		5.4Gbps	
SSH Pipe	1G		0.48Gbps	
SSH Pipe	10G		048Gbps	
SSH Pipe (arcfour)	10G		1.2Gbps	
NetCat TCP	10G		8.8Gbps	
HTTP localhost	local		6.2Gbps	
HTTPS localhost	local		3.3Gbps	
HTTP remote	1G		0.9Gbps	
HTTP remote	10G		4.0Gbps	
HTTP remote (MTU 9182)	10G		4.4Gbps	
HTTPS remote	10G		1.7Gbps	
S3 (MINIO Server)	10G		2.4Gbps	

## Manual local file system

Standard pipe to file = 8.8Gbps using linux pipe to file

```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 > /r  
  
0M Offset:      0GB Pkt:1622282576_055251189 Length: 240 Capture: 240 ChunkID:29484752 0.823Gb  
0M Offset:      1GB Pkt:1622282585_715451427 Length:9200 Capture:9200 ChunkID:29489109 8.997Gb  
0M Offset:      2GB Pkt:1622282585_944249315 Length:9200 Capture:9200 ChunkID:29493535 9.136Gb  
0M Offset:      3GB Pkt:1622282586_162785050 Length:9200 Capture:9200 ChunkID:29497764 8.730Gb  
0M Offset:      4GB Pkt:1622282586_391276769 Length:9200 Capture:9200 ChunkID:29502183 9.124Gb  
0M Offset:      5GB Pkt:1622282586_566815046 Length:9200 Capture:13296 ChunkID:29505578 7.009G  
0M Offset:      6GB Pkt:1622282586_787764113 Length:9200 Capture:9200 ChunkID:29509852 8.823Gb  
0M Offset:      7GB Pkt:1622282587_009307881 Length:9200 Capture:9200 ChunkID:29514137 8.846Gb  
0M Offset:      8GB Pkt:1622282587_231564003 Length:9200 Capture:9200 ChunkID:29518438 8.879Gb  
1M Offset:      9GB Pkt:1622282587_454907757 Length:9200 Capture:9200 ChunkID:29522758 8.918Gb  
1M Offset:     10GB Pkt:1622282587_663225292 Length:9200 Capture:9200 ChunkID:29526788 7.426Gb  
1M Offset:     11GB Pkt:1622282587_883709136 Length:9200 Capture:9200 ChunkID:29531053 8.804Gb  
1M Offset:     12GB Pkt:1622282588_104470038 Length:9200 Capture:9200 ChunkID:29535325 8.819Gb  
1M Offset:     13GB Pkt:1622282588_320895306 Length:9200 Capture:9200 ChunkID:29539511 8.642Gb  
1M Offset:     14GB Pkt:1622282588_544388720 Length:9200 Capture:9200 ChunkID:29543834 8.924Gb
```

## Manual Rclone local file system

local file system writes ~ 2.4Gbps

```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 | rc
```

```
0M Offset: 0GB Pkt:1622282576_055251189 Length: 240 Capture: 240 ChunkID:29484752 0.637Gb
0M Offset: 0GB Pkt:1622282585_557930242 Length:9200 Capture:9200 ChunkID:29486062 2.707Gb
0M Offset: 0GB Pkt:1622282585_576658386 Length:9200 Capture:9200 ChunkID:29486424 0.364Gb
0M Offset: 0GB Pkt:1622282585_636531964 Length:9200 Capture:13296 ChunkID:29487582 2.391G
0M Offset: 0GB Pkt:1622282585_697875404 Length:9200 Capture:9200 ChunkID:29488769 2.446Gb
0M Offset: 1GB Pkt:1622282585_757957634 Length:9200 Capture:9200 ChunkID:29489931 0.821Gb
0M Offset: 1GB Pkt:1622282585_818039752 Length:9200 Capture:9200 ChunkID:29491093 2.395Gb
0M Offset: 1GB Pkt:1622282585_878753413 Length:9200 Capture:9200 ChunkID:29492268 2.419Gb
0M Offset: 2GB Pkt:1622282585_939467105 Length:9200 Capture:9200 ChunkID:29493442 2.421Gb
0M Offset: 2GB Pkt:1622282585_964257052 Length:9200 Capture:9200 ChunkID:29493922 0.409Gb
0M Offset: 2GB Pkt:1622282586_024760327 Length:9200 Capture:9200 ChunkID:29495092 2.409Gb
0M Offset: 2GB Pkt:1622282586_085053021 Length:9200 Capture:9200 ChunkID:29496260 2.408Gb
0M Offset: 3GB Pkt:1622282586_146261525 Length:9200 Capture:9200 ChunkID:29497444 2.444Gb
0M Offset: 3GB Pkt:1622282586_164884418 Length:9200 Capture:9200 ChunkID:29497804 0.742Gb
0M Offset: 3GB Pkt:1622282586_225177096 Length:9200 Capture:9200 ChunkID:29498970 2.401Gb
0M Offset: 3GB Pkt:1622282586_286941475 Length:9200 Capture:9200 ChunkID:29500165 2.463Gb
0M Offset: 3GB Pkt:1622282586_348284877 Length:9200 Capture:9200 ChunkID:29501352 2.448Gb
0M Offset: 4GB Pkt:1622282586_408367173 Length:9200 Capture:9200 ChunkID:29502514 2.398Gb
0M Offset: 4GB Pkt:1622282586_469693866 Length:9200 Capture:9200 ChunkID:29503700 2.449Gb
0M Offset: 4GB Pkt:1622282586_529584009 Length:9200 Capture:9200 ChunkID:29504858 2.389Gb
0M Offset: 5GB Pkt:1622282586_590508196 Length:9200 Capture:9200 ChunkID:29506037 2.428Gb
```

## Manual NFS remote filesystem (10G Management)

Commands to write data to an NFS share over a 10G management interface ~ 5Gbps throughput

```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 > /r
```

```
0M Offset: 0GB Pkt:1622282576_055251189 Length: 240 Capture: 240 ChunkID:29484752 0.561Gb
0M Offset: 0GB Pkt:1622282585_629796243 Length:9200 Capture:9200 ChunkID:29487452 5.576Gb
0M Offset: 1GB Pkt:1622282585_749333062 Length:9200 Capture:9200 ChunkID:29489764 4.771Gb
0M Offset: 1GB Pkt:1622282585_869080067 Length:9200 Capture:9200 ChunkID:29492081 4.771Gb
0M Offset: 2GB Pkt:1622282586_016069259 Length:9200 Capture:9200 ChunkID:29494924 5.869Gb
0M Offset: 3GB Pkt:1622282586_158294580 Length:9200 Capture:9200 ChunkID:29497677 5.683Gb
0M Offset: 3GB Pkt:1622282586_299258901 Length:9200 Capture:9200 ChunkID:29500403 5.611Gb
0M Offset: 4GB Pkt:1622282586_438808972 Length:9200 Capture:9200 ChunkID:29503103 5.572Gb
0M Offset: 5GB Pkt:1622282586_574100988 Length:9200 Capture:9200 ChunkID:29505719 5.403Gb
0M Offset: 5GB Pkt:1622282586_720528885 Length:9200 Capture:9200 ChunkID:29508552 5.847Gb
0M Offset: 6GB Pkt:1622282586_862964788 Length:9200 Capture:9200 ChunkID:29511307 5.686Gb
0M Offset: 6GB Pkt:1622282586_987963325 Length:9200 Capture:9200 ChunkID:29513725 4.975Gb
0M Offset: 7GB Pkt:1622282587_103929106 Length:9200 Capture:9200 ChunkID:29515970 4.629Gb
0M Offset: 8GB Pkt:1622282587_228624524 Length:9200 Capture:9200 ChunkID:29518382 4.979Gb
1M Offset: 8GB Pkt:1622282587_368959022 Length:9200 Capture:9200 ChunkID:29521096 5.604Gb
```

## Manual SSH (1G Management)

Stock SSH over 1G management port = 0.48Gbps

```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 | ss
```

```
OM Offset:    0GB Pkt:1622282585_492207223 Length:9200 Capture:9200 ChunkID:29484791 0.048Gb
OM Offset:    0GB Pkt:1622282585_503385400 Length:9200 Capture:9200 ChunkID:29485007 0.446Gb
OM Offset:    0GB Pkt:1622282585_514746376 Length:9200 Capture:9200 ChunkID:29485227 0.454Gb
OM Offset:    0GB Pkt:1622282585_526153512 Length:9200 Capture:13296 ChunkID:29485447 0.455Gb
OM Offset:    0GB Pkt:1622282585_537787748 Length:9200 Capture:9200 ChunkID:29485673 0.465Gb
OM Offset:    0GB Pkt:1622282585_549433060 Length:9200 Capture:9200 ChunkID:29485898 0.465Gb
OM Offset:    0GB Pkt:1622282585_561237164 Length:9200 Capture:9200 ChunkID:29486126 0.471Gb
OM Offset:    0GB Pkt:1622282585_573192673 Length:9200 Capture:9200 ChunkID:29486357 0.477Gb
OM Offset:    0GB Pkt:1622282585_585146336 Length:9200 Capture:9200 ChunkID:29486589 0.477Gb
OM Offset:    0GB Pkt:1622282585_597101846 Length:9200 Capture:9200 ChunkID:29486820 0.477Gb
OM Offset:    0GB Pkt:1622282585_609070280 Length:9200 Capture:9200 ChunkID:29487051 0.477Gb
OM Offset:    0GB Pkt:1622282585_621018404 Length:9200 Capture:9200 ChunkID:29487282 0.477Gb
OM Offset:    0GB Pkt:1622282585_632968384 Length:9200 Capture:9200 ChunkID:29487514 0.477Gb
OM Offset:    0GB Pkt:1622282585_644910995 Length:9200 Capture:9200 ChunkID:29487745 0.477Gb
OM Offset:    0GB Pkt:1622282585_656883149 Length:9200 Capture:9200 ChunkID:29487976 0.477Gb
OM Offset:    0GB Pkt:1622282585_668831299 Length:9200 Capture:9200 ChunkID:29488207 0.477Gb
OM Offset:    0GB Pkt:1622282585_680736982 Length:9200 Capture:9200 ChunkID:29488438 0.475Gb
OM Offset:    0GB Pkt:1622282585_692709135 Length:9200 Capture:9200 ChunkID:29488669 0.477Gb
OM Offset:    0GB Pkt:1622282585_704675750 Length:9200 Capture:9200 ChunkID:29488901 0.478Gb
```

## Manual SSH (10G Management)

Stock SSH over 10G management port also ~ 0.48 Gbps

```

fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 | ss
OM Offset:      0GB Pkt:1622282585_492207223 Length:9200 Capture:9200 ChunkID:29484791 0.070Gb
OM Offset:      0GB Pkt:1622282585_500733952 Length:9200 Capture:9200 ChunkID:29484956 0.340Gb
OM Offset:      0GB Pkt:1622282585_512887039 Length:9200 Capture:9200 ChunkID:29485191 0.485Gb
OM Offset:      0GB Pkt:1622282585_525032740 Length:9200 Capture:9200 ChunkID:29485426 0.485Gb
OM Offset:      0GB Pkt:1622282585_537183972 Length:9200 Capture:9200 ChunkID:29485661 0.485Gb
OM Offset:      0GB Pkt:1622282585_549337047 Length:9200 Capture:9200 ChunkID:29485896 0.485Gb
OM Offset:      0GB Pkt:1622282585_561488276 Length:9200 Capture:9200 ChunkID:29486131 0.485Gb
OM Offset:      0GB Pkt:1622282585_573639505 Length:9200 Capture:9200 ChunkID:29486366 0.485Gb
OM Offset:      0GB Pkt:1622282585_585790734 Length:9200 Capture:9200 ChunkID:29486601 0.485Gb
OM Offset:      0GB Pkt:1622282585_597936423 Length:9200 Capture:9200 ChunkID:29486836 0.485Gb
OM Offset:      0GB Pkt:1622282585_610063649 Length:9200 Capture:9200 ChunkID:29487071 0.484Gb
OM Offset:      0GB Pkt:1622282585_622211185 Length:9200 Capture:13296 ChunkID:29487305 0.485G
OM Offset:      0GB Pkt:1622282585_634360584 Length:9200 Capture:13296 ChunkID:29487540 0.485G
OM Offset:      0GB Pkt:1622282585_646524765 Length:9200 Capture:9200 ChunkID:29487776 0.485Gb
OM Offset:      0GB Pkt:1622282585_658696328 Length:9200 Capture:9200 ChunkID:29488011 0.485Gb
OM Offset:      0GB Pkt:1622282585_670834659 Length:9200 Capture:9200 ChunkID:29488246 0.485Gb
OM Offset:      0GB Pkt:1622282585_682980376 Length:9200 Capture:9200 ChunkID:29488481 0.485Gb
OM Offset:      0GB Pkt:1622282585_695113167 Length:9200 Capture:9200 ChunkID:29488716 0.484Gb
OM Offset:      1GB Pkt:1622282585_707268116 Length:9200 Capture:9200 ChunkID:29488951 0.485Gb
OM Offset:      1GB Pkt:1622282585_718389073 Length:9200 Capture:9200 ChunkID:29489166 0.444Gb
OM Offset:      1GB Pkt:1622282585_728784387 Length:9200 Capture:9200 ChunkID:29489367 0.413Gb

```

## Manual SSH arcfour (10G Management)

Using arcfour cipher "-c arcfour" performance is about 1.2Gbps

```

fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 | ss
OM Offset:      0GB Pkt:1622282585_506241800 Length:9200 Capture:9200 ChunkID:29485062 0.643Gb
OM Offset:      0GB Pkt:1622282585_539874192 Length:9200 Capture:9200 ChunkID:29485713 1.343Gb
OM Offset:      0GB Pkt:1622282585_571565986 Length:9200 Capture:9200 ChunkID:29486326 1.265Gb
OM Offset:      0GB Pkt:1622282585_603374103 Length:9200 Capture:9200 ChunkID:29486941 1.270Gb
OM Offset:      0GB Pkt:1622282585_635289327 Length:9200 Capture:9200 ChunkID:29487558 1.273Gb
OM Offset:      0GB Pkt:1622282585_667396635 Length:9200 Capture:13296 ChunkID:29488179 1.282G
OM Offset:      0GB Pkt:1622282585_697670452 Length:9200 Capture:9200 ChunkID:29488765 1.208Gb
OM Offset:      1GB Pkt:1622282585_729663278 Length:9200 Capture:9200 ChunkID:29489384 1.277Gb
OM Offset:      1GB Pkt:1622282585_763941843 Length:9200 Capture:9200 ChunkID:29490047 1.369Gb
OM Offset:      1GB Pkt:1622282585_797193805 Length:9200 Capture:9200 ChunkID:29490690 1.328Gb
OM Offset:      1GB Pkt:1622282585_824173520 Length:9200 Capture:9200 ChunkID:29491212 1.077Gb
OM Offset:      1GB Pkt:1622282585_851123723 Length:9200 Capture:9200 ChunkID:29491733 1.075Gb
OM Offset:      1GB Pkt:1622282585_877660338 Length:9200 Capture:9200 ChunkID:29492247 1.059Gb
OM Offset:      1GB Pkt:1622282585_904750875 Length:9200 Capture:9200 ChunkID:29492771 1.081Gb
OM Offset:      2GB Pkt:1622282585_935715193 Length:9200 Capture:9200 ChunkID:29493369 1.236Gb
OM Offset:      2GB Pkt:1622282585_967401494 Length:9200 Capture:9200 ChunkID:29493982 1.265Gb
OM Offset:      2GB Pkt:1622282585_999126569 Length:9200 Capture:9200 ChunkID:29494596 1.266Gb

```

## Manual unencrypted TCP netcat (10G Management)

using netcat on a 10G Management interface over TCP results in near line rate throughput

```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ sudo stream_cat -v test9k_20210529_1902 | nc

0M Offset:    0GB Pkt:1622282576_055251189 Length: 240 Capture: 240 ChunkID:29484752 0.621Gb
0M Offset:    0GB Pkt:1622282585_704588968 Length:9200 Capture:9200 ChunkID:29488899 8.563Gb
0M Offset:    2GB Pkt:1622282585_927155222 Length:9200 Capture:9200 ChunkID:29493204 8.887Gb
0M Offset:    2GB Pkt:1622282586_132276814 Length:9200 Capture:9200 ChunkID:29497173 8.194Gb
0M Offset:    3GB Pkt:1622282586_334553100 Length:9200 Capture:9200 ChunkID:29501086 8.077Gb
0M Offset:    4GB Pkt:1622282586_561683869 Length:9200 Capture:9200 ChunkID:29505479 9.064Gb
0M Offset:    6GB Pkt:1622282586_785653665 Length:9200 Capture:9200 ChunkID:29509811 8.943Gb
0M Offset:    7GB Pkt:1622282587_003006067 Length:9200 Capture:9200 ChunkID:29514016 8.679Gb
0M Offset:    8GB Pkt:1622282587_226000777 Length:9200 Capture:9200 ChunkID:29518331 8.898Gb
1M Offset:    9GB Pkt:1622282587_450201254 Length:9200 Capture:9200 ChunkID:29522667 8.950Gb
1M Offset:   10GB Pkt:1622282587_682339341 Length:9200 Capture:9200 ChunkID:29527158 9.263Gb
1M Offset:   11GB Pkt:1622282587_903967999 Length:9200 Capture:9200 ChunkID:29531444 8.844Gb
1M Offset:   12GB Pkt:1622282588_109508894 Length:9200 Capture:9200 ChunkID:29535422 8.211Gb
1M Offset:   13GB Pkt:1622282588_338390081 Length:9200 Capture:9200 ChunkID:29539849 9.139Gb
1M Offset:   14GB Pkt:1622282588_557542580 Length:9200 Capture:9200 ChunkID:29544088 8.751Gb
1M Offset:   15GB Pkt:1622282588_773973543 Length:9200 Capture:9200 ChunkID:29548274 8.636Gb
```

## HTTP on local host

local host HTTP download ~ 6.2Gbps

```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ curl -u user:pass -s "http://localhost/pcap

LocalPipe 818937856 Bytes 6534731708 Bps 00:00:01 | 0.819 GB 6534.732 Mbps | Target 0.000 Mb
LocalPipe 1628307456 Bytes 6459537883 Bps 00:00:02 | 1.628 GB 6459.538 Mbps | Target 0.000 M
LocalPipe 2438987776 Bytes 6469417812 Bps 00:00:03 | 2.439 GB 6469.418 Mbps | Target 0.000 M
LocalPipe 3231449088 Bytes 6324536906 Bps 00:00:04 | 3.231 GB 6324.537 Mbps | Target 0.000 M
LocalPipe 4039114752 Bytes 6444755845 Bps 00:00:05 | 4.039 GB 6444.756 Mbps | Target 0.000 M
LocalPipe 4839702528 Bytes 6389507958 Bps 00:00:06 | 4.840 GB 6389.508 Mbps | Target 0.000 M
LocalPipe 5641994240 Bytes 6402493926 Bps 00:00:07 | 5.642 GB 6402.494 Mbps | Target 0.000 M
LocalPipe 6444417024 Bytes 6403559078 Bps 00:00:08 | 6.444 GB 6403.559 Mbps | Target 0.000 M
LocalPipe 7245660160 Bytes 6392818727 Bps 00:00:09 | 7.246 GB 6392.819 Mbps | Target 0.000 M
LocalPipe 8045068288 Bytes 6380023149 Bps 00:00:10 | 8.045 GB 6380.023 Mbps | Target 0.000 M
LocalPipe 8848146432 Bytes 6409364455 Bps 00:00:11 | 8.848 GB 6409.364 Mbps | Target 0.000 M
LocalPipe 9659613184 Bytes 6473375523 Bps 00:00:12 | 9.660 GB 6473.376 Mbps | Target 0.000 M
```

## HTTPS on local host

local host HTTPS download = 3.3Gbps



```
fmadio@fmadio20n40v3-363:/opt/fmadio/analytics$ curl -u user:pass -s -k "https://localhost/
```

```
LocalPipe 605159424 Bytes 4828459859 Bps 00:00:01 | 0.605 GB 4828.460 Mbps | Target 0.000 Mb  
LocalPipe 1197211648 Bytes 4724728813 Bps 00:00:02 | 1.197 GB 4724.729 Mbps | Target 0.000 M  
LocalPipe 1787559936 Bytes 4710501317 Bps 00:00:03 | 1.788 GB 4710.501 Mbps | Target 0.000 M  
LocalPipe 2376728576 Bytes 4701632651 Bps 00:00:04 | 2.377 GB 4701.633 Mbps | Target 0.000 M  
LocalPipe 2967601152 Bytes 4715474849 Bps 00:00:05 | 2.968 GB 4715.475 Mbps | Target 0.000 M  
LocalPipe 3553230848 Bytes 4673974271 Bps 00:00:06 | 3.553 GB 4673.974 Mbps | Target 0.000 M  
LocalPipe 4000186368 Bytes 3567065368 Bps 00:00:07 | 4.000 GB 3567.065 Mbps | Target 0.000 M  
LocalPipe 4413456384 Bytes 3298046933 Bps 00:00:08 | 4.413 GB 3298.047 Mbps | Target 0.000 M  
LocalPipe 4827774976 Bytes 3306335798 Bps 00:00:09 | 4.828 GB 3306.336 Mbps | Target 0.000 M  
LocalPipe 5238030336 Bytes 3273864766 Bps 00:00:10 | 5.238 GB 3273.865 Mbps | Target 0.000 M  
LocalPipe 5650382848 Bytes 3290488579 Bps 00:00:11 | 5.650 GB 3290.489 Mbps | Target 0.000 M  
LocalPipe 6062342144 Bytes 3287665615 Bps 00:00:12 | 6.062 GB 3287.666 Mbps | Target 0.000 M  
LocalPipe 6475087872 Bytes 3294125805 Bps 00:00:13 | 6.475 GB 3294.126 Mbps | Target 0.000 M  
LocalPipe 6889275392 Bytes 3303543281 Bps 00:00:14 | 6.889 GB 3303.543 Mbps | Target 0.000 M  
LocalPipe 7301234688 Bytes 3287167179 Bps 00:00:15 | 7.301 GB 3287.167 Mbps | Target 0.000 M  
LocalPipe 7717650432 Bytes 3322749934 Bps 00:00:16 | 7.718 GB 3322.750 Mbps | Target 0.000 M  
LocalPipe 8135770112 Bytes 3336692453 Bps 00:00:17 | 8.136 GB 3336.692 Mbps | Target 0.000 M  
LocalPipe 8552710144 Bytes 3326999809 Bps 00:00:18 | 8.553 GB 3327.000 Mbps | Target 0.000 M  
LocalPipe 8973058048 Bytes 3354286823 Bps 00:00:19 | 8.973 GB 3354.287 Mbps | Target 0.000 M  
LocalPipe 9399566336 Bytes 3403930909 Bps 00:00:20 | 9.400 GB 3403.931 Mbps | Target 0.000 M
```

## HTTP on remote 1G

1G link is the bottleneck ~ 0.9Gbps

```
fmadio@fmadio100v2-228U:~$ curl -u user:pass -s -k "https://192.168.2.225/pcap/single?Stre
```

```
LocalPipe 166330368 Bytes 927814503 Bps 00:00:01 | 0.166 GB 927.815 Mbps | Target 0.000 Mbps  
LocalPipe 333316096 Bytes 932791270 Bps 00:00:02 | 0.333 GB 932.791 Mbps | Target 0.000 Mbps  
LocalPipe 500301824 Bytes 932874648 Bps 00:00:04 | 0.500 GB 932.875 Mbps | Target 0.000 Mbps  
LocalPipe 665321472 Bytes 921817653 Bps 00:00:05 | 0.665 GB 921.818 Mbps | Target 0.000 Mbps  
LocalPipe 831651840 Bytes 929096659 Bps 00:00:07 | 0.832 GB 929.097 Mbps | Target 0.000 Mbps  
LocalPipe 996016128 Bytes 917807466 Bps 00:00:08 | 0.996 GB 917.807 Mbps | Target 0.000 Mbps  
LocalPipe 1163132928 Bytes 933238679 Bps 00:00:10 | 1.163 GB 933.239 Mbps | Target 0.000 Mbp  
LocalPipe 1329332224 Bytes 928205077 Bps 00:00:11 | 1.329 GB 928.205 Mbps | Target 0.000 Mbp  
LocalPipe 1495138304 Bytes 926206867 Bps 00:00:12 | 1.495 GB 926.207 Mbps | Target 0.000 Mbp  
LocalPipe 1660813312 Bytes 925473395 Bps 00:00:14 | 1.661 GB 925.473 Mbps | Target 0.000 Mbp  
LocalPipe 1827667968 Bytes 931651862 Bps 00:00:15 | 1.828 GB 931.652 Mbps | Target 0.000 Mbp  
LocalPipe 1994260480 Bytes 930385558 Bps 00:00:17 | 1.994 GB 930.386 Mbps | Target 0.000 Mbp  
LocalPipe 2160852992 Bytes 930127777 Bps 00:00:18 | 2.161 GB 930.128 Mbps | Target 0.000 Mbp  
LocalPipe 2327445504 Bytes 930038204 Bps 00:00:20 | 2.327 GB 930.038 Mbps | Target 0.000 Mbp  
LocalPipe 2494431232 Bytes 932802994 Bps 00:00:21 | 2.494 GB 932.803 Mbps | Target 0.000 Mbp  
LocalPipe 2660237312 Bytes 925971515 Bps 00:00:22 | 2.660 GB 925.972 Mbps | Target 0.000 Mbp  
LocalPipe 2826436608 Bytes 928112424 Bps 00:00:24 | 2.826 GB 928.112 Mbps | Target 0.000 Mbp  
LocalPipe 2992111616 Bytes 925527034 Bps 00:00:25 | 2.992 GB 925.527 Mbps | Target 0.000 Mbp
```

## HTTP on remote 10G (MTU 1500)

10G link HTTP download ~ 4.0 Gbps

```
fmadio@fmadio100v2-228U:~$ curl -s -u user:pass "http://192.168.15.225/pcap/single?StreamName=1"
LocalPipe 878444544 Bytes 4906299507 Bps 00:00:01 | 0.878 GB 4906.300 Mbps | Target 0.000 Mbps
LocalPipe 1713897472 Bytes 4667223979 Bps 00:00:02 | 1.714 GB 4667.224 Mbps | Target 0.000 Mbps
LocalPipe 2434269184 Bytes 4024425782 Bps 00:00:04 | 2.434 GB 4024.426 Mbps | Target 0.000 Mbps
LocalPipe 3163947008 Bytes 4075939934 Bps 00:00:05 | 3.164 GB 4075.940 Mbps | Target 0.000 Mbps
LocalPipe 3892314112 Bytes 4069109857 Bps 00:00:07 | 3.892 GB 4069.110 Mbps | Target 0.000 Mbps
LocalPipe 4604035072 Bytes 3976111408 Bps 00:00:08 | 4.604 GB 3976.111 Mbps | Target 0.000 Mbps
LocalPipe 5435555840 Bytes 4645208470 Bps 00:00:10 | 5.436 GB 4645.208 Mbps | Target 0.000 Mbps
LocalPipe 6163529728 Bytes 4067032394 Bps 00:00:11 | 6.164 GB 4067.032 Mbps | Target 0.000 Mbps
LocalPipe 6893862912 Bytes 4079623528 Bps 00:00:12 | 6.894 GB 4079.624 Mbps | Target 0.000 Mbps
LocalPipe 7627341824 Bytes 4097538820 Bps 00:00:14 | 7.627 GB 4097.539 Mbps | Target 0.000 Mbps
LocalPipe 8359641088 Bytes 4090988777 Bps 00:00:15 | 8.360 GB 4090.989 Mbps | Target 0.000 Mbps
LocalPipe 9091284992 Bytes 4087039352 Bps 00:00:17 | 9.091 GB 4087.039 Mbps | Target 0.000 Mbps
LocalPipe 9885974528 Bytes 4439451338 Bps 00:00:18 | 9.886 GB 4439.451 Mbps | Target 0.000 Mbps
LocalPipe 10612899840 Bytes 4060950187 Bps 00:00:20 | 10.613 GB 4060.950 Mbps | Target 0.000 Mbps
LocalPipe 11343495168 Bytes 4081107805 Bps 00:00:21 | 11.343 GB 4081.108 Mbps | Target 0.000 Mbps
LocalPipe 12132679680 Bytes 4408756581 Bps 00:00:22 | 12.133 GB 4408.757 Mbps | Target 0.000 Mbps
LocalPipe 12841648128 Bytes 3960847586 Bps 00:00:24 | 12.842 GB 3960.848 Mbps | Target 0.000 Mbps
LocalPipe 13564641280 Bytes 4039011477 Bps 00:00:25 | 13.565 GB 4039.011 Mbps | Target 0.000 Mbps
LocalPipe 14252638208 Bytes 3843236198 Bps 00:00:27 | 14.253 GB 3843.236 Mbps | Target 0.000 Mbps
LocalPipe 14963441664 Bytes 3970578070 Bps 00:00:28 | 14.963 GB 3970.578 Mbps | Target 0.000 Mbps
LocalPipe 15635972096 Bytes 3756948613 Bps 00:00:30 | 15.636 GB 3756.949 Mbps | Target 0.000 Mbps
LocalPipe 16293822464 Bytes 3674738834 Bps 00:00:31 | 16.294 GB 3674.739 Mbps | Target 0.000 Mbps
LocalPipe 17020223488 Bytes 4057956090 Bps 00:00:32 | 17.020 GB 4057.956 Mbps | Target 0.000 Mbps
LocalPipe 17753178112 Bytes 4094898722 Bps 00:00:34 | 17.753 GB 4094.899 Mbps | Target 0.000 Mbps
LocalPipe 18486394880 Bytes 4096251714 Bps 00:00:35 | 18.486 GB 4096.252 Mbps | Target 0.000 Mbps
LocalPipe 18512000024 Bytes 3851990522 Bps 00:00:35 | 18.512 GB 3851.991 Mbps | Target 0.000 Mbps
fmadio@fmadio100v2-228U:~$
```

## HTTP on remote 10G (MTU 9182)

10G HTTP download ~ 4.4Gbps



```
fmadio@fmadio100v2-228U:~$ curl -s -u user:pass "http://192.168.15.225/api/v1/pcap/single?St
LocalPipe 679870464 Bytes 3797577932 Bps 00:00:02 | 0.680 GB 3797.578 Mbps | Target 0.000 Mb
LocalPipe 1650196480 Bytes 5420372866 Bps 00:00:03 | 1.650 GB 5420.373 Mbps | Target 0.000 M
LocalPipe 2471231488 Bytes 4586265714 Bps 00:00:05 | 2.471 GB 4586.266 Mbps | Target 0.000 M
LocalPipe 3308650496 Bytes 4678462232 Bps 00:00:06 | 3.309 GB 4678.462 Mbps | Target 0.000 M
LocalPipe 4149346304 Bytes 4696431127 Bps 00:00:08 | 4.149 GB 4696.431 Mbps | Target 0.000 M
LocalPipe 4989255680 Bytes 4692021452 Bps 00:00:09 | 4.989 GB 4692.021 Mbps | Target 0.000 M
LocalPipe 5814091776 Bytes 4608138730 Bps 00:00:10 | 5.814 GB 4608.139 Mbps | Target 0.000 M
LocalPipe 6647316480 Bytes 4654626580 Bps 00:00:12 | 6.647 GB 4654.627 Mbps | Target 0.000 M
LocalPipe 7481720832 Bytes 4661095997 Bps 00:00:13 | 7.482 GB 4661.096 Mbps | Target 0.000 M
LocalPipe 8314814464 Bytes 4654060117 Bps 00:00:15 | 8.315 GB 4654.060 Mbps | Target 0.000 M
LocalPipe 9053667328 Bytes 4127496691 Bps 00:00:16 | 9.054 GB 4127.497 Mbps | Target 0.000 M
LocalPipe 9772859392 Bytes 4017858012 Bps 00:00:18 | 9.773 GB 4017.858 Mbps | Target 0.000 M
LocalPipe 10581442560 Bytes 4517126615 Bps 00:00:19 | 10.581 GB 4517.127 Mbps | Target 0.000
LocalPipe 11311775744 Bytes 4079828636 Bps 00:00:21 | 11.312 GB 4079.829 Mbps | Target 0.000
LocalPipe 12048269312 Bytes 4114598393 Bps 00:00:22 | 12.048 GB 4114.598 Mbps | Target 0.000
LocalPipe 12806914048 Bytes 4238419296 Bps 00:00:23 | 12.807 GB 4238.419 Mbps | Target 0.000
LocalPipe 13567393792 Bytes 4247994788 Bps 00:00:25 | 13.567 GB 4247.995 Mbps | Target 0.000
LocalPipe 14335213568 Bytes 4289693665 Bps 00:00:26 | 14.335 GB 4289.694 Mbps | Target 0.000
LocalPipe 15137505280 Bytes 4481923241 Bps 00:00:28 | 15.138 GB 4481.923 Mbps | Target 0.000
LocalPipe 15995109376 Bytes 4791161288 Bps 00:00:29 | 15.995 GB 4791.161 Mbps | Target 0.000
LocalPipe 16791633920 Bytes 4449621635 Bps 00:00:31 | 16.792 GB 4449.622 Mbps | Target 0.000
LocalPipe 17599430656 Bytes 4512679664 Bps 00:00:32 | 17.599 GB 4512.680 Mbps | Target 0.000
LocalPipe 18419023872 Bytes 4578803395 Bps 00:00:33 | 18.419 GB 4578.803 Mbps | Target 0.000
LocalPipe 18512000024 Bytes 578692144 Bps 00:00:35 | 18.512 GB 578.692 Mbps | Target 0.000 M
fmadio@fmadio100v2-228U:~$
```

## HTTPS on remote 10G

10G link HTTPS download ~ 1.7Gbps

```
fmadio@fmadio100v2-228U:~$ curl -u user:pass -s -k "https://192.168.15.225/pcap/single?Str
```

```
LocalPipe 305790976 Bytes 1707039342 Bps 00:00:01 | 0.306 GB 1707.039 Mbps | Target 0.000 Mb
LocalPipe 612237312 Bytes 1711520992 Bps 00:00:02 | 0.612 GB 1711.521 Mbps | Target 0.000 Mb
LocalPipe 919339008 Bytes 1715025565 Bps 00:00:04 | 0.919 GB 1715.026 Mbps | Target 0.000 Mb
LocalPipe 1227751424 Bytes 1722767466 Bps 00:00:05 | 1.228 GB 1722.767 Mbps | Target 0.000 M
LocalPipe 1535246336 Bytes 1717408524 Bps 00:00:07 | 1.535 GB 1717.409 Mbps | Target 0.000 M
LocalPipe 1841430528 Bytes 1709901716 Bps 00:00:08 | 1.841 GB 1709.902 Mbps | Target 0.000 M
LocalPipe 2147745792 Bytes 1711122237 Bps 00:00:10 | 2.148 GB 1711.122 Mbps | Target 0.000 M
LocalPipe 2455633920 Bytes 1720100662 Bps 00:00:11 | 2.456 GB 1720.101 Mbps | Target 0.000 M
LocalPipe 2762866688 Bytes 1716251145 Bps 00:00:12 | 2.763 GB 1716.251 Mbps | Target 0.000 M
LocalPipe 3070361600 Bytes 1717679541 Bps 00:00:14 | 3.070 GB 1717.680 Mbps | Target 0.000 M
LocalPipe 3378380800 Bytes 1720775250 Bps 00:00:15 | 3.378 GB 1720.775 Mbps | Target 0.000 M
LocalPipe 3684040704 Bytes 1707462492 Bps 00:00:17 | 3.684 GB 1707.462 Mbps | Target 0.000 M
LocalPipe 4003463168 Bytes 1783957320 Bps 00:00:18 | 4.003 GB 1783.957 Mbps | Target 0.000 M
LocalPipe 4303486976 Bytes 1675754912 Bps 00:00:20 | 4.303 GB 1675.755 Mbps | Target 0.000 M
LocalPipe 4611768320 Bytes 1721719137 Bps 00:00:21 | 4.612 GB 1721.719 Mbps | Target 0.000 M
LocalPipe 4916510720 Bytes 1702167210 Bps 00:00:22 | 4.917 GB 1702.167 Mbps | Target 0.000 M
LocalPipe 5225971712 Bytes 1724637586 Bps 00:00:24 | 5.226 GB 1724.638 Mbps | Target 0.000 M
LocalPipe 5550112768 Bytes 1810395082 Bps 00:00:25 | 5.550 GB 1810.395 Mbps | Target 0.000 M
```

### netcat remote 10G (1500 MTU)

Remove NGINX web server from the setup, throughput only slightly faster 4.8Gbps

```

fmadio@fmadio100v2-228U:~$ nc -l -p 5000 | pipe_mon --local-bytes > /dev/null
outputing local byte counts
calibrating...
0 : 2095077578          2.0951 cycles/nsec offset:4.922 Mhz
Cycles/Sec 2095077578.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
LocalPipe 131072 Bytes 0 Bps 00:00:02 | 0.000 GB 0.000 Mbps | Target 0.000 Mbps
LocalPipe 903217152 Bytes 5045163062 Bps 00:00:03 | 0.903 GB 5045.163 Mbps | Target 0.000 Mbps
LocalPipe 1769340928 Bytes 4838382982 Bps 00:00:05 | 1.769 GB 4838.383 Mbps | Target 0.000 Mbps
LocalPipe 2632056832 Bytes 4819594803 Bps 00:00:06 | 2.632 GB 4819.595 Mbps | Target 0.000 Mbps
LocalPipe 3475374080 Bytes 4711430757 Bps 00:00:07 | 3.475 GB 4711.431 Mbps | Target 0.000 Mbps
LocalPipe 4344905728 Bytes 4857474473 Bps 00:00:09 | 4.345 GB 4857.474 Mbps | Target 0.000 Mbps
LocalPipe 5217452032 Bytes 4874427604 Bps 00:00:10 | 5.217 GB 4874.428 Mbps | Target 0.000 Mbps
LocalPipe 6088032256 Bytes 4863780460 Bps 00:00:12 | 6.088 GB 4863.780 Mbps | Target 0.000 Mbps
LocalPipe 6961364992 Bytes 4878977682 Bps 00:00:13 | 6.961 GB 4878.978 Mbps | Target 0.000 Mbps
LocalPipe 7815692288 Bytes 4772724863 Bps 00:00:15 | 7.816 GB 4772.725 Mbps | Target 0.000 Mbps
LocalPipe 8680243200 Bytes 4829775317 Bps 00:00:16 | 8.680 GB 4829.775 Mbps | Target 0.000 Mbps
LocalPipe 9554231296 Bytes 4882563917 Bps 00:00:17 | 9.554 GB 4882.564 Mbps | Target 0.000 Mbps
LocalPipe 10392961024 Bytes 4685389001 Bps 00:00:19 | 10.393 GB 4685.389 Mbps | Target 0.000 Mbps
LocalPipe 11296571392 Bytes 5047718396 Bps 00:00:20 | 11.297 GB 5047.718 Mbps | Target 0.000 Mbps
LocalPipe 12204507136 Bytes 5072057752 Bps 00:00:22 | 12.205 GB 5072.058 Mbps | Target 0.000 Mbps
LocalPipe 13068271616 Bytes 4825085443 Bps 00:00:23 | 13.068 GB 4825.085 Mbps | Target 0.000 Mbps
LocalPipe 13931118592 Bytes 4820259720 Bps 00:00:25 | 13.931 GB 4820.260 Mbps | Target 0.000 Mbps
LocalPipe 14800781312 Bytes 4858162581 Bps 00:00:26 | 14.801 GB 4858.163 Mbps | Target 0.000 Mbps
LocalPipe 15670181888 Bytes 4856844009 Bps 00:00:27 | 15.670 GB 4856.844 Mbps | Target 0.000 Mbps
LocalPipe 16515072000 Bytes 4720283935 Bps 00:00:29 | 16.515 GB 4720.284 Mbps | Target 0.000 Mbps
LocalPipe 17386962944 Bytes 4871055601 Bps 00:00:30 | 17.387 GB 4871.056 Mbps | Target 0.000 Mbps
LocalPipe 18260426752 Bytes 4879737193 Bps 00:00:32 | 18.260 GB 4879.737 Mbps | Target 0.000 Mbps

```

### S3 HTTP Push rclone (Minio S3 Server 10G)

Push over HTTP (S3 EP is HTTP ) using a 10G interface

```
fmadio@fmadio20n40v3-363:/mnt/store0/tmp2$ sudo stream_cat -v test8k_20210704_0821 | rclone
0M Offset:    0GB Pkt:1625354491_474798026 Length:8192 Capture:8192 ChunkID:3241656 7.669Gbp
2021/07/04 08:59:01 NOTICE: Time may be set wrong - time from "192.168.15.132:9000" is 6m35.
2021/07/04 08:59:01 NOTICE: S3 bucket test: Streaming uploads using chunk size 5M will have
0M Offset:    0GB Pkt:1625354491_537006262 Length:8192 Capture:8192 ChunkID:3242876 2.452Gbp
0M Offset:    0GB Pkt:1625354518_184996807 Length:8192 Capture:8192 ChunkID:3244132 2.544Gbp
0M Offset:    0GB Pkt:1625354518_245005849 Length:8192 Capture:8192 ChunkID:3245309 2.396Gbp
0M Offset:    1GB Pkt:1625354518_308970777 Length:8192 Capture:8192 ChunkID:3246564 2.526Gbp
0M Offset:    1GB Pkt:1625354518_372007938 Length:8192 Capture:8192 ChunkID:3247800 2.461Gbp
0M Offset:    1GB Pkt:1625354518_435007189 Length:8192 Capture:8192 ChunkID:3249036 2.515Gbp
0M Offset:    2GB Pkt:1625354518_495064051 Length:8192 Capture:8192 ChunkID:3250214 2.398Gbp
0M Offset:    2GB Pkt:1625354518_554816611 Length:8192 Capture:8192 ChunkID:3251385 2.374Gbp
0M Offset:    2GB Pkt:1625354518_617361954 Length:8192 Capture:8192 ChunkID:3252612 2.497Gbp
0M Offset:    2GB Pkt:1625354518_676690089 Length:8192 Capture:8192 ChunkID:3253776 2.362Gbp
0M Offset:    3GB Pkt:1625354518_739200955 Length:8192 Capture:8192 ChunkID:3255002 2.496Gbp
0M Offset:    3GB Pkt:1625354720_421690032 Length:8192 Capture:8192 ChunkID:3256310 2.658Gbp
0M Offset:    3GB Pkt:1625354720_484727221 Length:8192 Capture:8192 ChunkID:3257546 2.499Gbp
0M Offset:    4GB Pkt:1625354720_544612914 Length:8192 Capture:8192 ChunkID:3258721 2.369Gbp
0M Offset:    4GB Pkt:1625354720_608701112 Length:8192 Capture:8192 ChunkID:3259978 2.533Gbp
0M Offset:    4GB Pkt:1625354720_672789301 Length:8192 Capture:8192 ChunkID:3261235 2.522Gbp
0M Offset:    4GB Pkt:1625354720_731623958 Length:8192 Capture:8192 ChunkID:3262388 2.320Gbp
0M Offset:    5GB Pkt:1625354720_794661149 Length:8192 Capture:8192 ChunkID:3263625 2.511Gbp
0M Offset:    5GB Pkt:1625354720_856647344 Length:8192 Capture:8192 ChunkID:3264840 2.462Gbp
0M Offset:    5GB Pkt:1625354720_918635074 Length:8192 Capture:8192 ChunkID:3266056 2.465Gbp
0M Offset:    6GB Pkt:1625354720_983937126 Length:8192 Capture:8192 ChunkID:3267337 2.602Gbp
0M Offset:    6GB Pkt:1625354721_040506940 Length:8192 Capture:8192 ChunkID:3268446 2.258Gbp
0M Offset:    6GB Pkt:1625354736_575673633 Length:8192 Capture:8192 ChunkID:3269641 2.368Gbp
0M Offset:    6GB Pkt:1625354736_635559460 Length:8192 Capture:8192 ChunkID:3270816 2.340Gbp
0M Offset:    7GB Pkt:1625354736_696494591 Length:8192 Capture:8192 ChunkID:3272011 2.388Gbp
0M Offset:    7GB Pkt:1625354736_755330823 Length:8192 Capture:8192 ChunkID:3273165 2.345Gbp
1M Offset:    7GB Pkt:1625354736_817316894 Length:8192 Capture:15888 ChunkID:3274380 2.417Gb
1M Offset:    8GB Pkt:1625354736_879302992 Length:8192 Capture:8192 ChunkID:3275596 2.421Gbp
1M Offset:    8GB Pkt:1625354736_941083437 Length:8192 Capture:8192 ChunkID:3276808 2.466Gbp
1M Offset:    8GB Pkt:1625354737_001176332 Length:8192 Capture:8192 ChunkID:3277986 2.380Gbp
1M Offset:    8GB Pkt:1625354737_063162399 Length:8192 Capture:8192 ChunkID:3279202 2.472Gbp
1M Offset:    9GB Pkt:1625354737_127961151 Length:8192 Capture:8192 ChunkID:3280473 2.587Gbp
packet stream end
20210704_085936 : SUCCESS
```

# Performance Capture

## Firmware: 7768

FMADIO100G Capture system is for 100% lossless half duplex 100G link.

The system can write 100Gbps+ to Disk and 148M Packets+ all without any loss. This assumes data is on a single 1x100G capture port

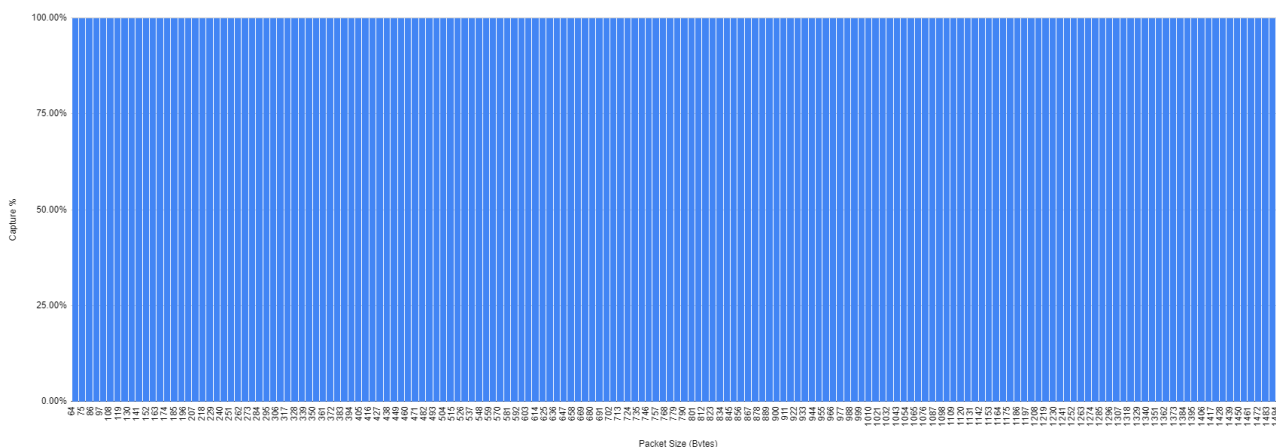
As the system has 2x100G capture ports there is a point where due to data rate or per packet rate, the system will drop packets, once it exceed 148M packets / sec or 100Gbps data rate.

The following graphs show the performance characteristics when exceeding 100Gbps line rate

## Microburst 1 x 100G @ 1,000,000,000 Packets

FMADIO100G Packet Capture system is fully lossless on a single 1 x 100G capture port across all packet sizes. Including up to 9218 Jumbo frame sizes.

1x100G Microburst 1M Packets

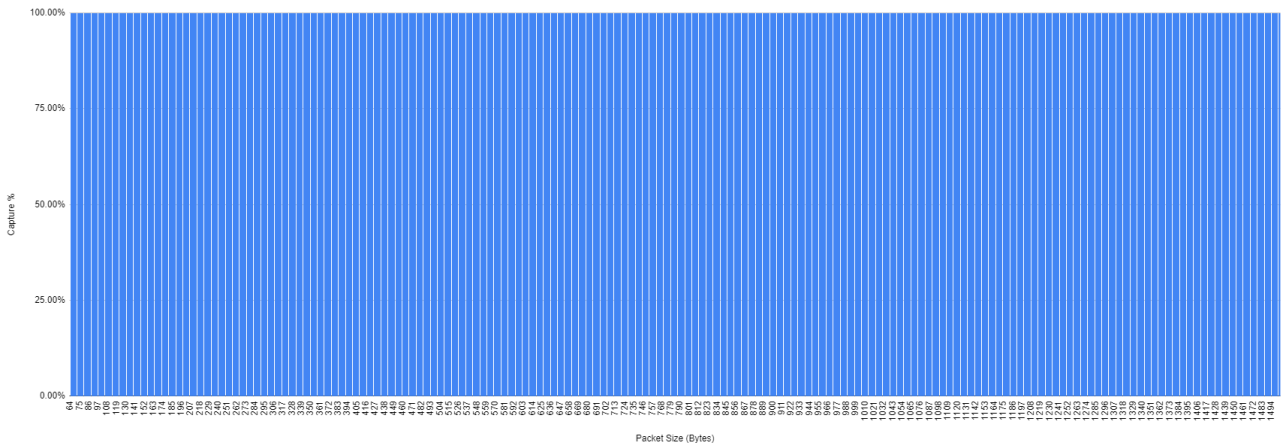


## Microburst 2 x 100G @ 100 Packets

Microburst of 100 packets at 2 x 100G line rate from 64B to 1500B packets.

Microbursts @ 200Gbps up to 100packets in length is 100% full capture no loss.

2x100G Microburst 100 Packets

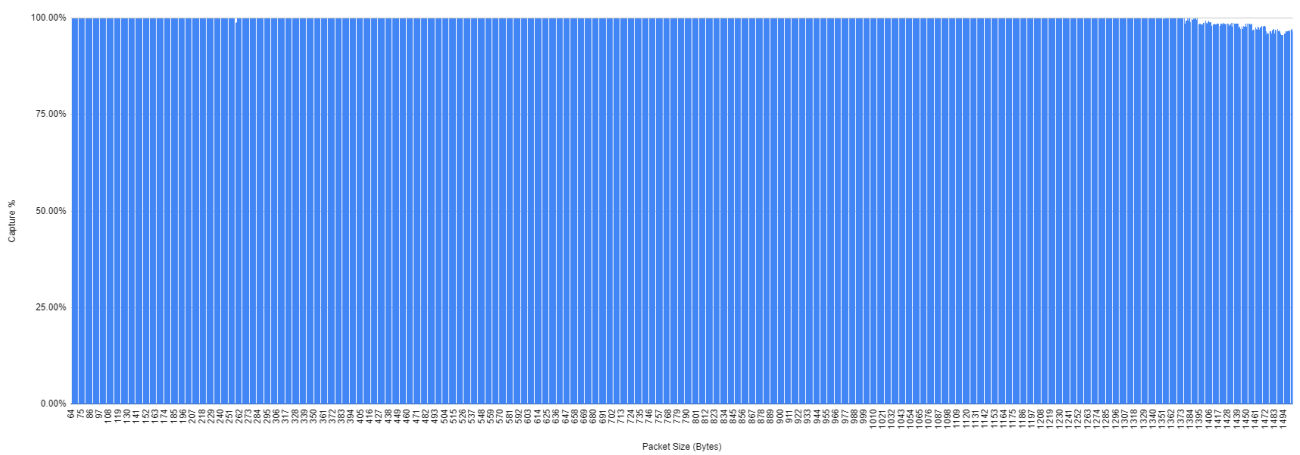


2x100G Microburst @ 100 Packets

### Microburst 2 x 100G @ 200 Packets

Microburst of 200 packets per port 2 x 100G line rate from 64B to 1500B packets. Slight capture loss on the larger packet sizes.

2x100G Microburst 200 Packets

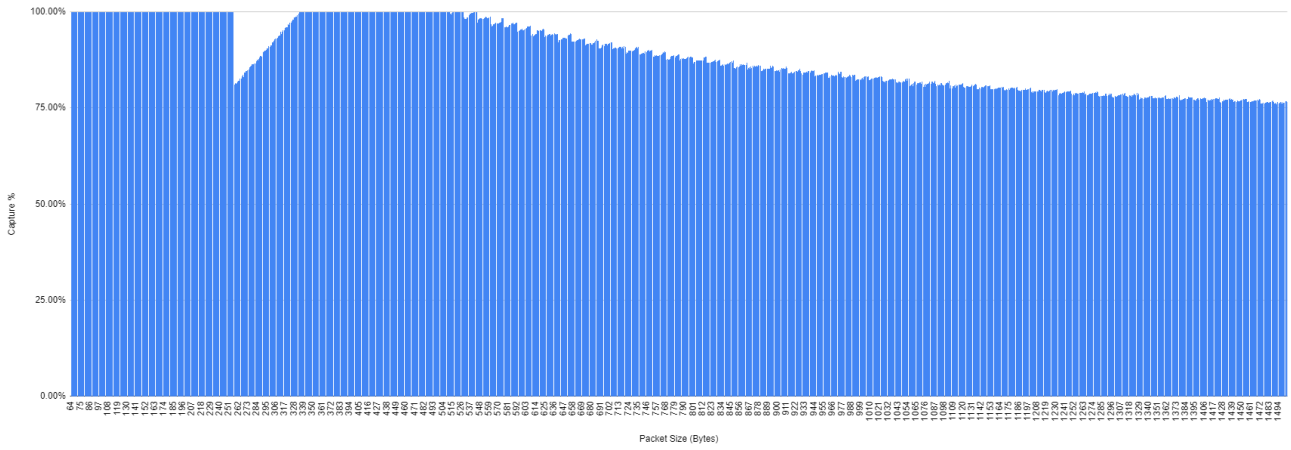


2x100G Microburst @ 200 Packets

### Microburst 2 x 100G @ 500Packets

Microburst of 500 packets per port 2 x 100G line rate from 64B to 1500B packets. Start to see more significant packet loss.

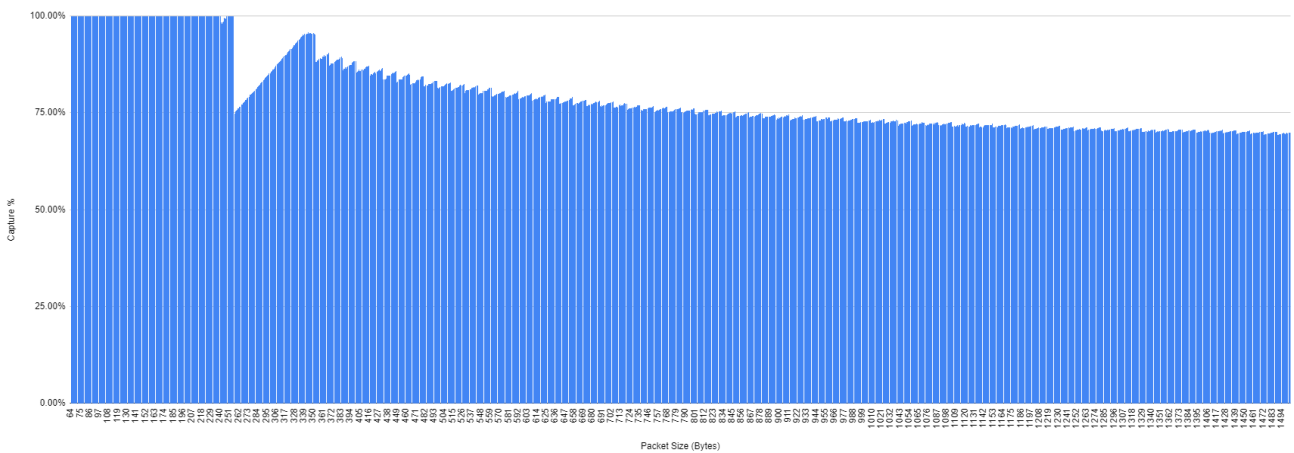
2x100G Microburst 500 Packets



### Microburst 2 x 100G @ 1,000 Packets

Microburst of 1K packets per port 2 x 100G line rate from 64B to 1500B packets.

2x100G Microburst 1K Packets

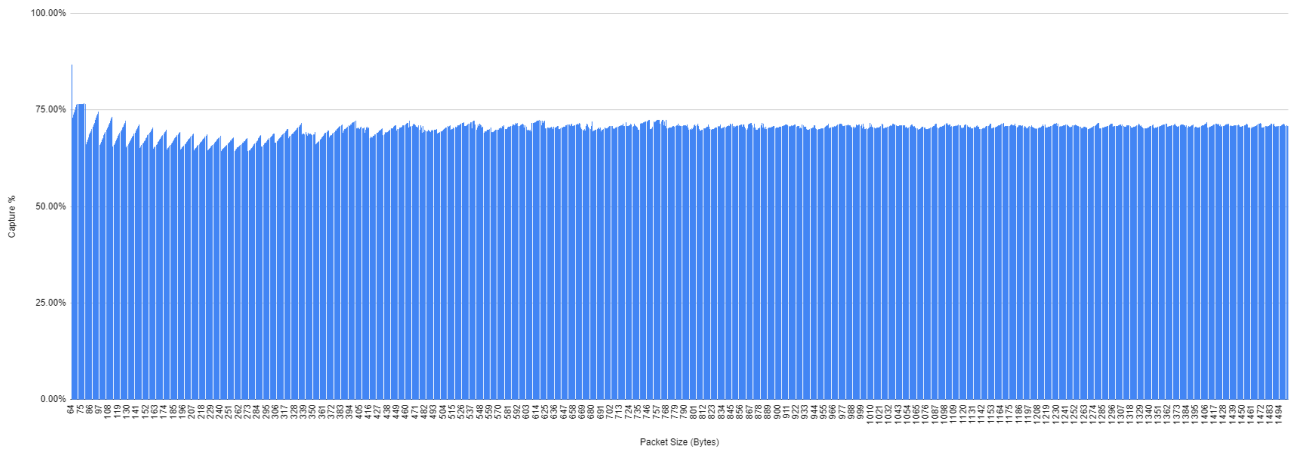


2x100G Microburst @ 1K Packets

### Microburst 2 x 100G @ 10,000 Packets

Microburst of 10K packets per port 2 x 100G line rate 64B to 1500B packets

2x100G Microburst 10K Packets

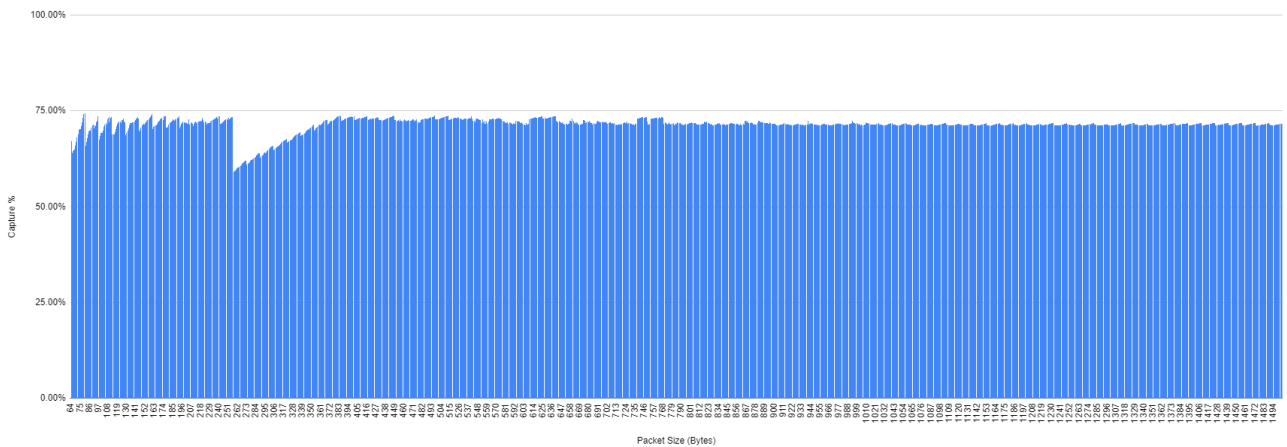


2x100G Microburst @ 10K Packets

### Microburst 2 x 100G @ 100,000 Packets

Microburst of 100K packets per port 2 x 100G line rate 64B to 1500B packets

2x100G Microburst 100K Packets



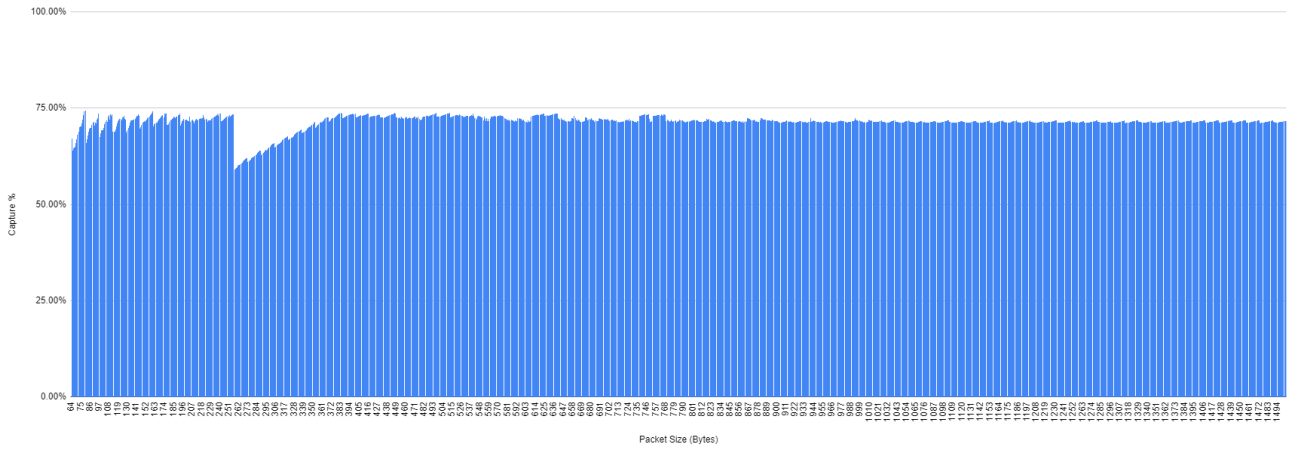
2x100G Microburst @ 100K Packets

### Microburst 2 x 100G @ 1,000,000 Packets

Microburst of 1M Packets per port 2 x 100G line rate 64B to 1500B packets



# 2x100G Microburst 1M Packets



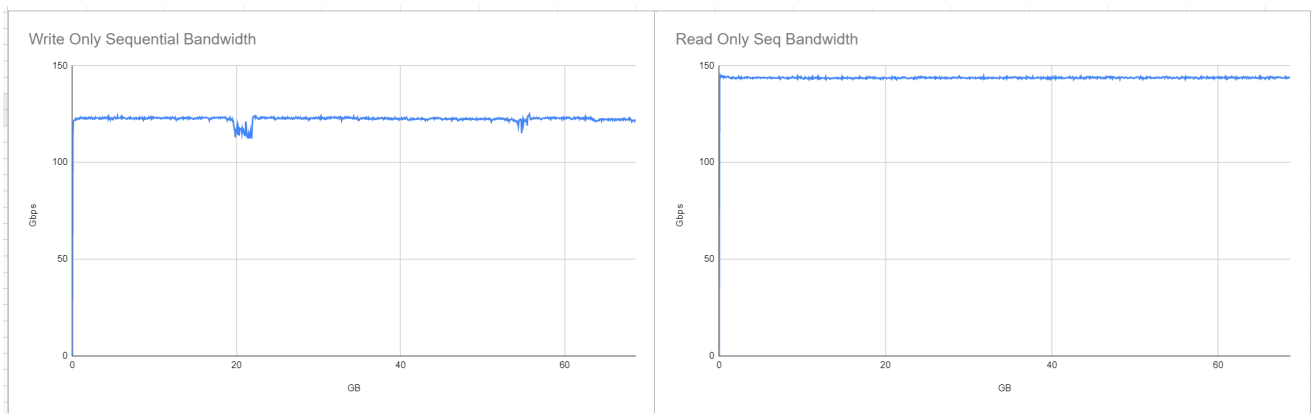
2x100G Microburst @ 1M packets

# Performance Capture Analyze

In many scenarios there is a requirement for simultaneous capture and analysis of data in pseudo-realtime. The target performance is bursting (up to 156TB worth) of 100Gbps line rate traffic, with an average sustained data rate of ~ 50Gbps. We target 50Gbps as per the graphs below is the hardware limitation of simultaneous sustained 50Gbps write + sustained 50Gbps read.

## Capture Only / Analyze Only

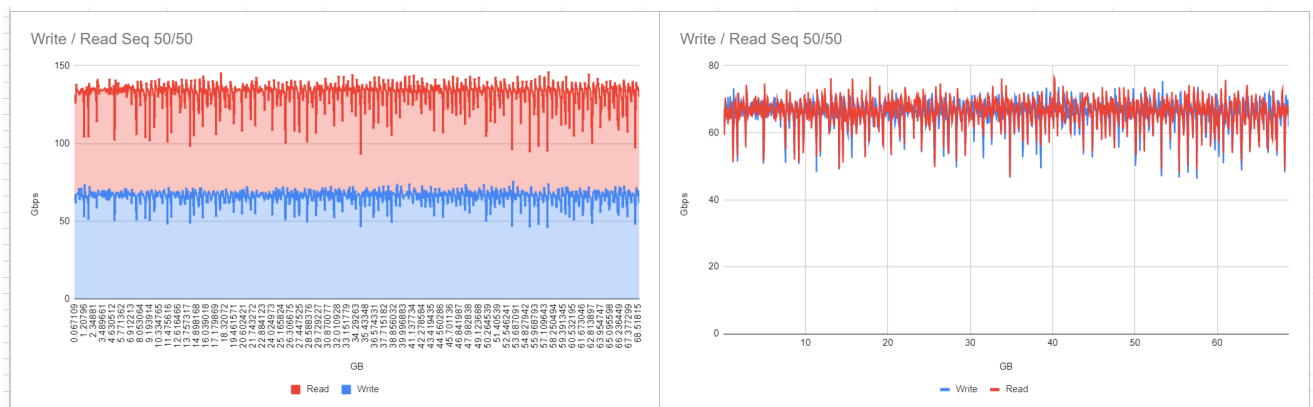
In the below image shows Capture/Write only, or Analyze/Read only FMADIO 100G Gen2 system easily does 100Gbps worth of IO bandwidth



Capture Only and Analyze Only Raw IO throughput

## Burst 100Gbps Capture + Sustained 50Gbps Capture/Analyze

The ideal IO profile is an simultaneous sustained 50Gbps Capture/Write and 50Gbps Analyze/Read profile as shown below. This allows the capture to burst to 100Gbps without significantly effecting the Analyze/Read performance of the system.



100Gbps Burst Capture + Sustained 50Gbps Capture/Analyze raw IO Throughput

## Sustained 100Gbps Capture / 26Gbps Analyze

Due to hardware IO limitations (~ 150Gbps max aggregate bandwidth) at a sustained 100Gbps Capture/Write only a 26Gbps sustained Analyze/Read performance is achievable.



Sustained 100Gbps Capture + 26Gbps sustained Analyze raw IO Throughput

Please note all the above numbers, are the maximum limits. Typically the Analyze software performance is impacted more by the packet rate than the data rate, As such the Analysis software performance is usually the bottleneck, not the raw IO hardware limits.

# PCAP Upload

All FMADIO Packet capture systems can also upload raw 3rd party PCAP files into the system. This allows Packetscope, Tcpscope, Analysis and Replay plugins to work on external and archived historical data. The upload functionality is heavily used internally for our own testing and regression frameworks.

Please note:

**Capturing must be stopped. Running Capture and Upload simultaneously results in undefined behavior**

## Upload Local PCAP CLI

If the PCAP your uploading is small, you can

### Step 1)

scp the PCAP onto the OS disk. e.g. /mnt/store0/tmp2/

```
scp upload.pcap fmadio@192.168.1.1:/mnt/store0/tmp2/
```

### Step 2)

Upload using the utility **stream\_upload**. The upload fetches data via stdin allowing a wide range of options from a local PCAP file, to remote PCAP, to a curl URL or PCAP generation utility running on the system. The following example is a simple upload a PCAP that's on the local filesystem.

```
fmadio@fmadio20-049:/mnt/store0/tmp2$ cat hitcon_small.pcap | sudo stream_upload --name test
FSPrefetch
FSPrefetch Chunk: 5000.184ms 5000ms
FSPrefetch Chunk timeout: 5000.000ms max
create stream [test_upload_20170725_1625]
  0.178GB Uploaded
  0.610GB Uploaded
  0.864GB Uploaded
  1.116GB Uploaded
  1.335GB Uploaded
  1.472GB Uploaded
  1.667GB Uploaded
  1.793GB Uploaded
  1.947GB Uploaded
  2.225GB Uploaded
fmadio@fmadio20-049:/mnt/store0/tmp2$
```

Note: the timestamp resolution of the uploaded PCAP is automatically detected and converted to FMADIO native nanosecond format.

### Step 3)

Confirm upload.

```
fmadio@fmadio20-049:/mnt/store0/tmp2$ sudo stream_dump
Streams:
[0000] [this should be empty]      0GB Chunk(Cnt:      0 Start:      1 End:      0) I
[0001] test_upload_20170725_1625    2GB Chunk(Cnt:    9341 Start:      8 End:    9348) I
```

## Upload Remote PCAP CLI

Sometimes you need to upload very large multi TB PCAP to the FMADIO Packet Capture System. In such cases there isn't enough local storage on the OS disk for the scp method to work. To upload a large PCAP use the streaming/pipe functionality of the stream\_upload utility.

In this example we are uploading a raw PCAP over SSH into the FMADIO Packet Capture system. From an SSH shell on the capture system the command SSH's into the remote system where the PCAP is stored and issues a "cat" command on the PCAP to be uploaded.

Effectively piping the remote PCAP down the ssh connection. This is then read by the stream\_upload command in --stdin mode, instead of reading from the local file system. For maximum performance its best to use the 10G management port for the connection.

```
$ ssh user@10.10.10.10 cat path_to_pcap.pcap | sudo /opt/fmadio/bin/stream_upload --name rer
FSPrefetch
FSPrefetch Chunk: 5000.184ms 5000ms
FSPrefetch Chunk timeoued: 5000.000ms max
create stream [remote_upload_20170725_1625]
  0.178GB Uploaded
  0.610GB Uploaded
  0.864GB Uploaded
  1.116GB Uploaded
  1.335GB Uploaded
  1.472GB Uploaded
  1.667GB Uploaded
  1.793GB Uploaded
  1.947GB Uploaded
  2.225GB Uploaded
  .
  .
  .
  .
fmadio@fmadio20-049:~$
```

Using this approach the PCAP is streamed onto the system via SSH, with no temporarily files created. The maximum PCAP that can be uploaded is limited by the capture systems total storage capacity.

# Packet Blaster

All Gen2+ FMADIO devices have a built in packet blaster / Layer 2 packet generator. This allows a single system to be entirely self contained for unit and system testing. In addition FMADIO devices can also load test network devices such as switching and firewalls, checking for physical layer links and measuring network path latency.

Packet blaster is a layer 2 (Ethernet level) packet generator that runs at full line rate @ 64B to 9218 Jumbo sized packets. Generation is performed entirely on the FPGA Capture card thus up to full 100Gbps @ 64B 148Mpps packets can be generated without generation variance. Packet generation and capture can run simultaneously, thus verification the capture device is operating correctly is achieved.

The payload of each packet is a per physical port MAC Address followed by a 32bit incrementally increasing sequence number. This sequence number is used later post capture to ensure data of all packets has been captured correctly without error. An example packet is shown in Wireshark below.

test64\_20191004\_1151\_split\_115146.872.180.915-115146.876.576.168.pcap

ファイル(F) 編集(E) 表示(V) 移動(G) キャプチャ(C) 分析(A) 統計(S) 電話(y) 無線(W) ツール(T) ヘルプ(H)

表示フィルタ ... <Ctrl-/> を適用

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	Private_11:11:11	Intel_11:11:11	0x1111	64	Ethernet II
2	0.000000000	22:22:22:22:22:22	Schaffne_22:22:22	0x2222	64	Ethernet II
3	0.000000064	Private_11:11:11	Intel_11:11:11	0x1111	64	Ethernet II
4	0.000000064	22:22:22:22:22:22	Schaffne_22:22:22	0x2222	64	Ethernet II
5	0.000000128	Private_11:11:11	Intel_11:11:11	0x1111	64	Ethernet II
6	0.000000128	22:22:22:22:22:22	Schaffne_22:22:22	0x2222	64	Ethernet II
7	0.000000198	Private_11:11:11	Intel_11:11:11	0x1111	64	Ethernet II
8	0.000000205	22:22:22:22:22:22	Schaffne_22:22:22	0x2222	64	Ethernet II
9	0.000000262	Private_11:11:11	Intel_11:11:11	0x1111	64	Ethernet II
10	0.000000269	22:22:22:22:22:22	Schaffne_22:22:22	0x2222	64	Ethernet II
11	0.000000333	22:22:22:22:22:22	Schaffne_22:22:22	0x2222	64	Ethernet II
12	0.000000330	Private_11:11:11	Intel_11:11:11	0x1111	64	Ethernet II

> Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)

> Ethernet II, Src: Private\_11:11:11 (11:11:11:11:11:11), Dst: Intel\_11:11:11 (00:11:11:11:11:11)

> Data (50 bytes)

```
0000 00 11 11 11 11 11 11 11 11 11 11 11 11 11 11 .....
0010 c7 0d 56 ca c8 0d 56 ca c9 0d 56 ca ca 0d 56 ca ..V..V..V..V..
0020 cb 0d 56 ca cc 0d 56 ca cd 0d 56 ca ce 0d 56 ca ..V..V..V..V..
0030 cf 0d 56 ca d0 0d 56 ca d1 0d 56 ca f0 49 3e 31 ..V..V..V..I>1
```

Packet Blaster

In the above wireshark picture, you can see 2 different MAC address 11:11:11:11:11:11 (Physical Port 1) and 22:22:22:22:22:22 (Physical Port 2). The payload is a 32bit little endian sequence numbers

```
0x11111111 (Per Physical Port MAC Address)
0x11111111 (Per Physical Port MAC Address)
0x11111111 (Per Physical Port MAC Address)
0x11111111 (Per Physical Port MAC Address)
0x11111111 (Per Physical Port MAC Address)
```

```
0xca560dc7 (Data payload Seq Number + 0)
0xca560dc8 (Data payload Seq Number + 1)
0xca560dc9 (Data payload Seq Number + 2)
0xca560dca (Data payload Seq Number + 3)
0xca560dcb (Data payload Seq Number + 4)
0xca560dcc (Data payload Seq Number + 5)
0xca560dce (Data payload Seq Number + 5)
0xca560dcf (Data payload Seq Number + 6)
0xca560dd0 (Data payload Seq Number + 7)
0xca560dd1 (Data payload Seq Number + 8)
0x313e49f0 (Frame Check Sequence)
```

This sequence number and MAC address allow the analysis software to not only check the total number of packets captured by the device, but also check every byte of the payload has been captured without error. As the analysis software knows exactly what the packet payload data should be via the sequence number.

## Operation

Packet blaster is operated only by the CLI interface, each FMAD SKU has a slightly different operation

### FMADIO100v2

```
fmadio@fmadio100v2-228:~$ sudo stream_generate_f100 --blaster --help
Packet Gen: Oct  3 2019 18:44:18
stream_generate_f100 --blaster ::: FMADIO 100G Packet Blaster :::

Command line options:
--pktsize      <packet size>      : Size of each packet (64-9218) Default 64B
--pktcnt       <packet count>     : Total number of packets to generate. Scientific notation
--gbps         <data rate>        : Data rate to generate at, e.g. 50.0 Gbps (Default is 100
--port-enable  <port mask>        : Which ports to enable 0 is disable, 1 enable (Default is

fmadio@fmadio100v2-228:~$
```

Example operation, generate 1 billion 64B packets on a single 100G port at full line rate



```

fmadio@fmadio100v2-228:$ sudo stream_generate_f100 --blaster --pktcnt 1e9 --pktsize 64 --por
Packet Gen: Oct  3 2019 18:44:18
PktCnt: 1000000000
PktSize: 64
PortEnable: 0 1
Generate: PktSize:64 PktCnt:1000.000M Gbps:100.000000
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
GenConfig(0, 1,  1, 1, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 0, 0x0, 0x4,  1, 0, 1, 0x0, 0
Config Length: 40
TargetPkt Time: 51.200001ns per packet
DataPktTime   : 51.200001
Add extra     : 0.000000 ns
Pad Cycles    : 0
Final Wait    : 1
PortMode: 0 1
update 191 35712 0.000000 : 03010122
update 14212540 909614080 72.734941 : 05000134
update 28424336 1819170048 145.459036 : 05000134
update 42635968 2728713536 218.187497 : 05000134
update 56842067 3637903488 290.999828 : 05000134
update 71052937 4547399168 363.629380 : 05000134
update 85262420 5456806464 436.391969 : 03010122
update 99468647 6366004608 509.219176 : 03010122
.
.
.
.

```

## Data Verification

Payload data verification on the device is achieved by "linux-cat-ing" a capture down a linux pipe to the builtin utility capinfos2. The syntax looks as follows

### FMADIO100v2

```

fmadio@fmadio100v2-228:$ sudo stream_cat --ignore_fcs <capture name>| capinfos2 -v --seq --
No PortID
PCAP nano
Port:4 new seq: af000000 Packets: 0 Length: 68
0.00GB    0.000 Gbps    0.000 Mpps
0.33GB    2.584 Gbps    3.845 Mpps
0.74GB    3.295 Gbps    4.903 Mpps
1.09GB    2.831 Gbps    4.213 Mpps
1.40GB    2.487 Gbps    3.701 Mpps
1.78GB    3.034 Gbps    4.514 Mpps
2.15GB    2.932 Gbps    4.363 Mpps
2.50GB    2.816 Gbps    4.191 Mpps
2.82GB    2.556 Gbps    3.804 Mpps
3.23GB    3.260 Gbps    4.852 Mpps
3.59GB    2.893 Gbps    4.305 Mpps
3.90GB    2.487 Gbps    3.701 Mpps
4.26GB    2.877 Gbps    4.281 Mpps
4.56GB    2.362 Gbps    3.514 Mpps
4.86GB    2.439 Gbps    3.629 Mpps
5.16GB    2.385 Gbps    3.549 Mpps
5.47GB    2.434 Gbps    3.622 Mpps
5.87GB    3.221 Gbps    4.793 Mpps
6.17GB    2.406 Gbps    3.581 Mpps
6.59GB    3.306 Gbps    4.920 Mpps
6.95GB    2.859 Gbps    4.254 Mpps
7.29GB    2.705 Gbps    4.025 Mpps
7.63GB    2.773 Gbps    4.126 Mpps
7.99GB    2.862 Gbps    4.259 Mpps
8.35GB    2.883 Gbps    4.290 Mpps
packet stream end
SUCCESS
8.40GB    0.321 Gbps    0.478 Mpps
Total Packets   : 100000000
TotalBytes      : 6800000000
TotalPackets    : 100000000
PayloadCRC      : 4791a6a7add60780
ErrorSeq        : 0
ErrorPktSize    : 0
LastByte        : 0x0e5e0fff
SeqStart        : 0x00000000 0x00000000 0x00000000 0x00000000 : 0xaf000000
SeqEnd          : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x0e5e0fff
PacketCnt       : 0 0 0 0
TimeOrder       : 0
CRCFail         : 0
TotalPCAPTime   : 0 ns
Bandwidth       : 77.273 Gbps
Packet Rate     : 142.045 Mpps

Complete
fmadio@fmadio100v2-228:$

```



# Replay & Capture (Slow)

FMADIO can replay PCAP traffic at low speed while simultaneously capturing at full line rate. This can be very helpful for debugging or other troubleshooting and does not require changing the FPGA firmware from Capture to Replay mode. This Replay mode uses PIO (Programmed IO) who's bandwidth is quite low, and is not suitable for full high bandwidth packet relay. Usage is as follows

## FMADIO20v3

Example command on the 20G Packet Capture system

```
/opt/fmadio/bin/stream_generate_f20 --replay
```

Example below pipes a PCAP previously captured on the device back down the the capture interfaces.

```

fmadio@fmadio20v2-149:~$ sudo stream_cat --ignore_fcs test64_20191004_1151 | sudo stream_gen
Packet Gen: Sep 22 2019 01:05:06
stream_cat ioqueue: 4
SetAffinity: CPU 30 Index:16
StartChunkID: 8994
PCAP Nano
Replay Pkts:          1 Drop:          0 Total:0.000GB 0.396Gbps
Replay Pkts:   649312 Drop:          0 Total:0.042GB 0.332Gbps
Replay Pkts:  1299210 Drop:          0 Total:0.083GB 0.333Gbps
Replay Pkts:  1949573 Drop:          0 Total:0.125GB 0.333Gbps
Replay Pkts:  2599309 Drop:          0 Total:0.166GB 0.333Gbps
Replay Pkts:  3249180 Drop:          0 Total:0.208GB 0.333Gbps
Replay Pkts:  3899671 Drop:          0 Total:0.250GB 0.333Gbps
Replay Pkts:  4549914 Drop:          0 Total:0.291GB 0.333Gbps
Replay Pkts:  5199871 Drop:          0 Total:0.333GB 0.333Gbps
Replay Pkts:  5828751 Drop:          0 Total:0.373GB 0.332Gbps
Replay Pkts:  6478494 Drop:          0 Total:0.415GB 0.332Gbps
Replay Pkts:  7128289 Drop:          0 Total:0.456GB 0.332Gbps
Replay Pkts:  7777780 Drop:          0 Total:0.498GB 0.332Gbps
Replay Pkts:  8425123 Drop:          0 Total:0.539GB 0.332Gbps
Replay Pkts:  9074288 Drop:          0 Total:0.581GB 0.332Gbps
Replay Pkts:  9723647 Drop:          0 Total:0.622GB 0.332Gbps
Replay Pkts: 10379903 Drop:          0 Total:0.664GB 0.332Gbps
Replay Pkts: 11029807 Drop:          0 Total:0.706GB 0.332Gbps
Replay Pkts: 11679917 Drop:          0 Total:0.748GB 0.332Gbps
Replay Pkts: 12327988 Drop:          0 Total:0.789GB 0.332Gbps
Replay Pkts: 12977595 Drop:          0 Total:0.831GB 0.332Gbps
Replay Pkts: 13629543 Drop:          0 Total:0.872GB 0.332Gbps
Replay Pkts: 14285072 Drop:          0 Total:0.914GB 0.332Gbps
Replay Pkts: 14928107 Drop:          0 Total:0.955GB 0.332Gbps
Replay Pkts: 15575152 Drop:          0 Total:0.997GB 0.332Gbps
Replay Pkts: 16222518 Drop:          0 Total:1.038GB 0.332Gbps
Replay Pkts: 16871634 Drop:          0 Total:1.080GB 0.332Gbps
Replay Pkts: 17521499 Drop:          0 Total:1.121GB 0.332Gbps
Replay Pkts: 18177284 Drop:          0 Total:1.163GB 0.332Gbps
Replay Pkts: 18823901 Drop:          0 Total:1.205GB 0.332Gbps
Replay Pkts: 19472977 Drop:          0 Total:1.246GB 0.332Gbps
packet stream end
SUCCESS
STDIN Read fail: 0
Replay Pkts: 20000000 Drop:          0 Total:1.280GB 0.322Gbps
fmadio@fmadio20v2-149:~$

```

## FMADIO100v2

For SKUs FMADIO100v2 and FMADIO40v3

```
PCAP linux stdin pipe | sudo /opt/fmadio/bin/stream_generate_f100 --replay_pio
```

You can pipe a PCAP from a local file system or use stream\_cat to pipe from the capture system. Example below pipes from a previous captures.

```
fmadio@fmadio100v2-228:~$ sudo stream_cat --ignore_fcs testk9k_b_20200727_0106 | sudo ./strea
```

```
Packet Gen: Jul 27 2020 00:24:51
```

```
map 0x7f73115e0000
```

```
0x7f7308c4a000
```

```
StartChunk: 3856856
```

```
PCAP Nano
```

```
Replay Pkts:      1 Drop:      0 Total:0.000GB 0.579Gbps
```

```
Replay Pkts:    40225 Drop:      0 Total:0.056GB 0.451Gbps
```

```
Replay Pkts:    80542 Drop:      0 Total:0.113GB 0.451Gbps
```

```
Replay Pkts:   120730 Drop:      0 Total:0.169GB 0.451Gbps
```

```
Replay Pkts:   160842 Drop:      0 Total:0.225GB 0.450Gbps
```

```
Replay Pkts:   200879 Drop:      0 Total:0.281GB 0.450Gbps
```

```
Replay Pkts:   240992 Drop:      0 Total:0.337GB 0.450Gbps
```

```
Replay Pkts:   281222 Drop:      0 Total:0.394GB 0.450Gbps
```

```
Replay Pkts:   321452 Drop:      0 Total:0.450GB 0.450Gbps
```

```
Replay Pkts:   361580 Drop:      0 Total:0.506GB 0.450Gbps
```

```
Replay Pkts:   401815 Drop:      0 Total:0.563GB 0.450Gbps
```

```
Replay Pkts:   442053 Drop:      0 Total:0.619GB 0.450Gbps
```

```
Replay Pkts:   482276 Drop:      0 Total:0.675GB 0.450Gbps
```

```
Replay Pkts:   522380 Drop:      0 Total:0.731GB 0.450Gbps
```

```
Replay Pkts:   562478 Drop:      0 Total:0.787GB 0.450Gbps
```

```
Replay Pkts:   602678 Drop:      0 Total:0.844GB 0.450Gbps
```

```
Replay Pkts:   642979 Drop:      0 Total:0.900GB 0.450Gbps
```

```
Replay Pkts:   683165 Drop:      0 Total:0.956GB 0.450Gbps
```

```
Replay Pkts:   723334 Drop:      0 Total:1.013GB 0.450Gbps
```

```
Replay Pkts:   763528 Drop:      0 Total:1.069GB 0.450Gbps
```

```
Replay Pkts:   803801 Drop:      0 Total:1.125GB 0.450Gbps
```

```
Replay Pkts:   843917 Drop:      0 Total:1.181GB 0.450Gbps
```

```
Replay Pkts:   884163 Drop:      0 Total:1.238GB 0.450Gbps
```

```
Replay Pkts:   924157 Drop:      0 Total:1.294GB 0.450Gbps
```

```
.  
. .  
. .  
. .
```

The following example pipes a PCAP on the localfile system (/mnt/store0/tmp/imix.pcap) to replay the traffic

```
fmadio@fmadio100v2-228$ cat imix10.pcap | sudo stream_generate_f100 --replay_pio
```

```
Packet Gen: Jul 28 2020 01:05:12
```

```
SetAffinity: CPU 1 Index:18
```

```
PCAP Nano
```

Replay Pkts:	1	Drop:	0	Total:	0.000GB	0.506Gbps
Replay Pkts:	137652	Drop:	0	Total:	0.049GB	0.389Gbps
Replay Pkts:	273955	Drop:	0	Total:	0.097GB	0.390Gbps
Replay Pkts:	411187	Drop:	0	Total:	0.146GB	0.391Gbps
Replay Pkts:	548794	Drop:	0	Total:	0.196GB	0.391Gbps
Replay Pkts:	686374	Drop:	0	Total:	0.244GB	0.391Gbps
Replay Pkts:	822256	Drop:	0	Total:	0.293GB	0.390Gbps
Replay Pkts:	955237	Drop:	0	Total:	0.340GB	0.389Gbps
Replay Pkts:	1092409	Drop:	0	Total:	0.389GB	0.389Gbps
Replay Pkts:	1227813	Drop:	0	Total:	0.437GB	0.389Gbps
Replay Pkts:	1361926	Drop:	0	Total:	0.485GB	0.388Gbps
Replay Pkts:	1498830	Drop:	0	Total:	0.534GB	0.388Gbps
Replay Pkts:	1634867	Drop:	0	Total:	0.582GB	0.388Gbps
Replay Pkts:	1771895	Drop:	0	Total:	0.631GB	0.388Gbps
Replay Pkts:	1907879	Drop:	0	Total:	0.680GB	0.388Gbps
Replay Pkts:	2039344	Drop:	0	Total:	0.726GB	0.387Gbps
Replay Pkts:	2176207	Drop:	0	Total:	0.775GB	0.388Gbps
Replay Pkts:	2310531	Drop:	0	Total:	0.823GB	0.387Gbps
Replay Pkts:	2446486	Drop:	0	Total:	0.871GB	0.387Gbps
Replay Pkts:	2583970	Drop:	0	Total:	0.921GB	0.388Gbps
Replay Pkts:	2722008	Drop:	0	Total:	0.970GB	0.388Gbps
Replay Pkts:	2859852	Drop:	0	Total:	1.019GB	0.388Gbps
Replay Pkts:	2997241	Drop:	0	Total:	1.068GB	0.388Gbps
Replay Pkts:	3134724	Drop:	0	Total:	1.117GB	0.388Gbps

```
.  
.   
.   
.
```

# PCAP Flow Generation

One of the massive benefits of a full line rate PCAP replay feature is, you can generate PCAPs at any speed, upload them to the FMAD Packet Replay Device and then replay them at any speed you required. There are many ways to generate PCAP files for replay, we will use our builtin utility pcap\_genflow ( [https://github.com/fmadio/pcap\\_genflow](https://github.com/fmadio/pcap_genflow) however many other tools such as tcpreplay, iperf3 and others which can output to a PCAP file for upload.

- is to generate the PCAP using various toolchains
- Upload the PCAP into the FMADIO Capture System
- Replay the capture at any speed

## PCAP Generation

Some example using pcap\_genflow as follows

100Gbps @ 64B Packets, TCP , 1 Billion Unique Flows

Generate 1 billion packets, with 1M unique TCP flows using 64B packets @ 100Gbps

```
$ ./pcap_genflow --pktcnt 1e9 --pktsize 64 --flowcnt 1e6 --bps 100e9 > flow_1M_64B_100G.pcap
```

100Gbps @ 1500B packets, TCP, 1 Million Unique Flows

Generate 100M packets, with 1M unique TCP flows using 1500B packets @ 100Gbps

```
$ ./pcap_genflow --pktcnt 100e6 --pktsize 1500 --flowcnt 1e6 --bps 100e9 > flow_1M_1500B_100G.pcap
```

100Gbps @ IMIX Packet size distribution, TCP, 1 Million Unique Flows

Generate 100M packets, with 1M unique TCP flows using IMIX packet size distribution @ 100Gbps

```
$ ./pcap_genflow --pktcnt 100e6 --imix --flowcnt 1e6 --bps 100e9 > flow_1M_IMIX_100G.pcap
```

## Uploading PCAP into FMADIO

Typically when generating PCAP files the output is written to a linux pipe, the FMAD PCAP File upload function always reads PCAPs from stdin. The tool to upload PCAP into the FMAD capture system is stream\_upload. The syntax is



```

fmadio@fmadio100v2-228:$ sudo ./stream_upload --help
Stream Upload V3: Oct  4 2019 20:00:10
Stream Uploader V3
-----
uploader always reads from stdin

stream_upload

--append-fcs          : this appends an FCS to all packets
--name <stream_name> : set the uploaded capture name to stream_name
--verbose            : prints basic status as upload in progress
--slice <slice amount> : emulate packet slicing
--time-compress <scale> : how much to scale the input PCAPS timestamp

fmadio@fmadio100v2-228:/mnt/store0/develop_20191003_rc2/stream_upload$

```

Example of uploading the previously generated file flow\_1M\_IMIX\_100G.pcap. PCAP can be scp to the device, or streamed over an SSH connection

```
$ cat flow_1M_IMIX_100G.pcap | sudo stream_upload --name flow_1M_IMIX_100G
```

Or this can be generated via on the FMADIO device itself, by piping the output of pcap\_genflow directly to stream\_upload

```
$ ./pcap_genflow --pktcnt 100e6 --imix --flowcnt 1e6 --bps 100e9 | sudo stream_upload --name
```

## Replay PCAP File

Once the PCAP has been uploaded into the capture system, it can be replayed with various options as discussed [here](#). It should be clear PCAP Generation + Upload + Replay is a powerful tool for any Network Engineer. If you have suggestions or questions feel free to contact us.

# CLI Reference

# fmadiocli

fmadiocli is a command line interface using a switch style interface to operate FMADIO Packet Capture devices.

Brief help description is provided using ? command

```
[Wed Apr 27 07:18:20 2022] ?
[Wed Apr 27 07:18:20 2022] config
[Wed Apr 27 07:18:20 2022] config analytics schedule del <schedule row name>
[Wed Apr 27 07:18:20 2022] config analytics schedule mod <schedule row name>
[Wed Apr 27 07:18:20 2022] config analytics schedule new
[Wed Apr 27 07:18:20 2022] config capture schedule del <schedule row name>
[Wed Apr 27 07:18:20 2022] config capture schedule mod <schedule row name>
[Wed Apr 27 07:18:20 2022] config capture schedule new
[Wed Apr 27 07:18:20 2022] config capture start <capture name>
[Wed Apr 27 07:18:20 2022] config capture stop
[Wed Apr 27 07:18:20 2022] config interface dns <interface name> <dns>
[Wed Apr 27 07:18:20 2022] config interface gateway <interface name> <gateway>
[Wed Apr 27 07:18:20 2022] config interface ip <interface name> <ip address>
[Wed Apr 27 07:18:20 2022] config interface mode <interface name> <disabled|static>
[Wed Apr 27 07:18:20 2022] config interface netmask <interface name> <netmask>
.
.
.
```

Verbose help description is provided using ??? command

```

[Wed Apr 27 07:18:56 2022] > ???
[Wed Apr 27 07:18:56 2022] =====
[Wed Apr 27 07:18:56 2022] ?
[Wed Apr 27 07:18:56 2022] -----
[Wed Apr 27 07:18:56 2022] Command Line Help Info
[Wed Apr 27 07:18:56 2022]
[Wed Apr 27 07:18:56 2022] =====
[Wed Apr 27 07:18:56 2022] config
[Wed Apr 27 07:18:56 2022] -----
[Wed Apr 27 07:18:56 2022] enter configuration mode
[Wed Apr 27 07:18:56 2022]
[Wed Apr 27 07:18:56 2022] =====
[Wed Apr 27 07:18:56 2022] config analytics schedule del
[Wed Apr 27 07:18:56 2022] -----
[Wed Apr 27 07:18:56 2022] Deletes a analytics scheduler row
[Wed Apr 27 07:18:56 2022]
[Wed Apr 27 07:18:56 2022] Example Usage:
[Wed Apr 27 07:18:56 2022] > config analytics schedule del <name>           : deletes
[Wed Apr 27 07:18:56 2022]
[Wed Apr 27 07:18:56 2022]
.
.
.

```

## Command History

Command history is stored in the file

```
/opt/fmadio/etc/fmadiocli.history
```

It can be deleted to clear the history

## Configure Interfaces

### show interface status

FW: 7856+

Shows the current state of the interfaces

```
show interface status
```

Example below shows port status on an FMADIO100Gv2 Analytics system

```
[Wed Apr 27 07:16:43 2022] > show interface status
[Wed Apr 27 07:16:43 2022] Port          Description Status          Speed          Transceiver
[Wed Apr 27 07:16:43 2022] -----
[Wed Apr 27 07:16:43 2022] man0                connected        1G
[Wed Apr 27 07:16:43 2022] man10               connected        40G           40G Base-SR4 0
[Wed Apr 27 07:16:43 2022] cap0                notconnected     100G          100G CR
[Wed Apr 27 07:16:43 2022] cap1                connected        100G          100G CR
[Wed Apr 27 07:16:43 2022] >
```

## show interface ip

FW: 8336+

Shows the currently configured IP address information for the management and BMC/IPMI/Capture ports

```
show interface ip
```

Example below shows the status on an FMADIO20Gv3 system

```
[Tue Dec 13 04:18:36 2022] > show interface ip
[Tue Dec 13 04:18:37 2022] Port          Mode          IP              Netmask         Gateway
[Tue Dec 13 04:18:37 2022] -----
[Tue Dec 13 04:18:38 2022] bmc          static        192.168.187.2   255.255.255.0   192.168.187
[Tue Dec 13 04:18:38 2022] man0         static        192.168.187.10 255.255.255.0   192.168.187
[Tue Dec 13 04:18:38 2022] man1         disabled     192.168.1.2     255.255.255.0   192.168.1.1
[Tue Dec 13 04:18:38 2022] man10        static        192.168.91.50   255.255.255.0
[Tue Dec 13 04:18:38 2022] cap0
[Tue Dec 13 04:18:38 2022] cap1
[Tue Dec 13 04:18:38 2022] -----
[Tue Dec 13 04:18:38 2022] >
```

## show interface counter

Shows RMON1 counter information on each capture port.

```
show interface counter
```

Example below is FMADIO20Gv3 system output

```

[Tue Dec 13 04:40:01 2022] > show interface counter
[Tue Dec 13 04:40:01 2022] Port                Packet                Byte
[Tue Dec 13 04:40:01 2022] -----+-----
[Tue Dec 13 04:40:01 2022]   cap0      Total :                0                0
[Tue Dec 13 04:40:01 2022]                Under :                0                0
[Tue Dec 13 04:40:01 2022]                64 :                0                0
[Tue Dec 13 04:40:01 2022]                65-127 :            0                0
[Tue Dec 13 04:40:01 2022]                128-255 :           0                0
[Tue Dec 13 04:40:01 2022]                256-511 :           0                0
[Tue Dec 13 04:40:01 2022]                512-1023 :          0                0
[Tue Dec 13 04:40:01 2022]                1024-1518 :         0                0
[Tue Dec 13 04:40:01 2022]                1024-2047 :         0                0
[Tue Dec 13 04:40:01 2022]                2048_4095 :         0                0
[Tue Dec 13 04:40:01 2022]                4096_8191 :         0                0
[Tue Dec 13 04:40:01 2022]                8192_9216 :         0                0
[Tue Dec 13 04:40:01 2022]                Over :                0                0
[Tue Dec 13 04:40:01 2022] -----+-----
[Tue Dec 13 04:40:01 2022]   cap1      Total :            303349            125359723
[Tue Dec 13 04:40:01 2022]                Under :                0                0
[Tue Dec 13 04:40:01 2022]                64 :                58765            0
[Tue Dec 13 04:40:01 2022]                65-127 :            146476            0
[Tue Dec 13 04:40:01 2022]                128-255 :            11891            0
[Tue Dec 13 04:40:01 2022]                256-511 :             8283            0
[Tue Dec 13 04:40:01 2022]                512-1023 :            15793            0
[Tue Dec 13 04:40:01 2022]                1024-1518 :           62140            0
[Tue Dec 13 04:40:01 2022]                1024-2047 :           62141            0
[Tue Dec 13 04:40:01 2022]                2048_4095 :                0                0
[Tue Dec 13 04:40:01 2022]                4096_8191 :                0                0
[Tue Dec 13 04:40:01 2022]                8192_9216 :                0                0
[Tue Dec 13 04:40:01 2022]                Over :                0                0
[Tue Dec 13 04:40:01 2022] -----+-----
[Tue Dec 13 04:40:01 2022] >

```

Example below shows FMADIO100v2 in 8x10G port output

[Tue Dec 13 04:40:57 2022] > show interface counter

[Tue Dec 13 04:40:58 2022]	Port		Packet	Byte
[Tue Dec 13 04:40:58 2022]	-----+			
[Tue Dec 13 04:40:58 2022]	cap0	Total :	0	0
[Tue Dec 13 04:40:58 2022]		Under :	0	0
[Tue Dec 13 04:40:58 2022]		64 :	0	0
[Tue Dec 13 04:40:58 2022]		65-127 :	0	0
[Tue Dec 13 04:40:58 2022]		128-255 :	0	0
[Tue Dec 13 04:40:58 2022]		256-511 :	0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :	0	0
[Tue Dec 13 04:40:58 2022]		1024-1518 :	0	0
[Tue Dec 13 04:40:58 2022]		1024-2047 :	0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :	0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :	0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :	0	0
[Tue Dec 13 04:40:58 2022]		Over :	0	0
[Tue Dec 13 04:40:58 2022]	-----+			
[Tue Dec 13 04:40:58 2022]	cap1	Total :	0	0
[Tue Dec 13 04:40:58 2022]		Under :	0	0
[Tue Dec 13 04:40:58 2022]		64 :	0	0
[Tue Dec 13 04:40:58 2022]		65-127 :	0	0
[Tue Dec 13 04:40:58 2022]		128-255 :	0	0
[Tue Dec 13 04:40:58 2022]		256-511 :	0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :	0	0
[Tue Dec 13 04:40:58 2022]		1024-1518 :	0	0
[Tue Dec 13 04:40:58 2022]		1024-2047 :	0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :	0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :	0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :	0	0
[Tue Dec 13 04:40:58 2022]		Over :	0	0
[Tue Dec 13 04:40:58 2022]	-----+			
[Tue Dec 13 04:40:58 2022]	cap2	Total :	0	0
[Tue Dec 13 04:40:58 2022]		Under :	0	0
[Tue Dec 13 04:40:58 2022]		64 :	0	0
[Tue Dec 13 04:40:58 2022]		65-127 :	0	0
[Tue Dec 13 04:40:58 2022]		128-255 :	0	0
[Tue Dec 13 04:40:58 2022]		256-511 :	0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :	0	0
[Tue Dec 13 04:40:58 2022]		1024-1518 :	0	0config captur
[Tue Dec 13 04:40:58 2022]		1024-2047 :	0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :	0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :	0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :	0	0
[Tue Dec 13 04:40:58 2022]		Over :	0	0
[Tue Dec 13 04:40:58 2022]	-----+			
[Tue Dec 13 04:40:58 2022]	cap3	Total :	0	0
[Tue Dec 13 04:40:58 2022]		Under :	0	0
[Tue Dec 13 04:40:58 2022]		64 :	0	0
[Tue Dec 13 04:40:58 2022]		65-127 :	0	0
[Tue Dec 13 04:40:58 2022]		128-255 :	0	0
[Tue Dec 13 04:40:58 2022]		256-511 :	0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :	0	0

[Tue Dec 13 04:40:58 2022]		1024-1518 :	conf	0	0
[Tue Dec 13 04:40:58 2022]		1024-2047 :		0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :		0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :		0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :		0	0
[Tue Dec 13 04:40:58 2022]		Over :		0	0
[Tue Dec 13 04:40:58 2022]		-----+			
[Tue Dec 13 04:40:58 2022]	cap4	Total :		0	0
[Tue Dec 13 04:40:58 2022]		Under :		0	0
[Tue Dec 13 04:40:58 2022]		64 :		0	0
[Tue Dec 13 04:40:58 2022]		65-127 :		0	0
[Tue Dec 13 04:40:58 2022]		128-255 :		0	0
[Tue Dec 13 04:40:58 2022]		256-511 :		0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :		0	0
[Tue Dec 13 04:40:58 2022]		1024-1518 :		0	0
[Tue Dec 13 04:40:58 2022]		1024-2047 :		0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :		0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :		0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :		0	0
[Tue Dec 13 04:40:58 2022]		Over :		0	0
[Tue Dec 13 04:40:58 2022]		-----+			
[Tue Dec 13 04:40:58 2022]	cap5	Total :		0	0
[Tue Dec 13 04:40:58 2022]		Under :		0	0
[Tue Dec 13 04:40:58 2022]		64 :		0	0
[Tue Dec 13 04:40:58 2022]		65-127 :		0	0
[Tue Dec 13 04:40:58 2022]		128-255 :		0	0
[Tue Dec 13 04:40:58 2022]		256-511 :		0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :		0	0
[Tue Dec 13 04:40:58 2022]		1024-1518 :		0	0
[Tue Dec 13 04:40:58 2022]		1024-2047 :		0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :		0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :		0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :		0	0
[Tue Dec 13 04:40:58 2022]		Over :		0	0
[Tue Dec 13 04:40:58 2022]		-----+			
[Tue Dec 13 04:40:58 2022]	cap6	Total :		0	0
[Tue Dec 13 04:40:58 2022]		Under :		0	0
[Tue Dec 13 04:40:58 2022]		64 :		0	0
[Tue Dec 13 04:40:58 2022]		65-127 :		0	0
[Tue Dec 13 04:40:58 2022]		128-255 :		0	0
[Tue Dec 13 04:40:58 2022]		256-511 :		0	0
[Tue Dec 13 04:40:58 2022]		512-1023 :		0	0
[Tue Dec 13 04:40:58 2022]		1024-1518 :		0	0
[Tue Dec 13 04:40:58 2022]		1024-2047 :		0	0
[Tue Dec 13 04:40:58 2022]		2048_4095 :		0	0
[Tue Dec 13 04:40:58 2022]		4096_8191 :		0	0
[Tue Dec 13 04:40:58 2022]		8192_9216 :		0	0
[Tue Dec 13 04:40:58 2022]		Over :		0	0
[Tue Dec 13 04:40:58 2022]		-----+			
[Tue Dec 13 04:40:58 2022]	cap7	Total :		0	0
[Tue Dec 13 04:40:58 2022]		Under :		0	0
[Tue Dec 13 04:40:58 2022]		64 :		0	0
[Tue Dec 13 04:40:58 2022]		65-127 :		0	0



```

[Tue Dec 13 04:40:58 2022]          128-255 :          0          0
[Tue Dec 13 04:40:58 2022]          256-511 :          0          0
[Tue Dec 13 04:40:58 2022]          512-1023 :          0          0
[Tue Dec 13 04:40:58 2022]         1024-1518 :          0          0
[Tue Dec 13 04:40:58 2022]         1024-2047 :          0          0
[Tue Dec 13 04:40:58 2022]         2048_4095 :          0          0
[Tue Dec 13 04:40:58 2022]         4096_8191 :          0          0
[Tue Dec 13 04:40:58 2022]         8192_9216 :          0          0
[Tue Dec 13 04:40:58 2022]           Over :          0          0
[Tue Dec 13 04:40:58 2022] -----+-----

```

## config interface shutdown

FW: 7856+ support for 100Gv2 2x100G 2x40G

This shuts down a specific capture interface as specified, usually this is cap0 or cap1 and depends on the SKU and Port configuration on which ports can be shutdown

```
config interface shutdown <interface>
```

```

[Wed Apr 27 07:13:15 2022] > config interface shutdown cap0
[Wed Apr 27 07:13:15 2022]     Disable cycle calibration
[Wed Apr 27 07:13:15 2022]     PCIVersion: 50434930 50434930
[Wed Apr 27 07:13:15 2022]     PortMask:1 0 0 0 0 0 0 0
[Wed Apr 27 07:13:15 2022]     2x100G pcs shutdown 1
[Wed Apr 27 07:13:15 2022] set interface [cap0] mode () -> (disabled)
[Wed Apr 27 07:13:15 2022]
[Wed Apr 27 07:13:21 2022] >

```

## config interface no shutdown

FW: 7856+ support for 100Gv2 2x100G 2x40G

Re-enables the specified capture interface from shutdown status. Depending on the link peer, the link peer might need to be bounced as it may be in a shutdown error state.

```
config interface no shutdown <interface>
```

```
[Wed Apr 27 07:13:21 2022] > config interface no shutdown cap0
[Wed Apr 27 07:13:21 2022]
[Wed Apr 27 07:13:21 2022]      Disable cycle calibration
[Wed Apr 27 07:13:21 2022]      PCIVersion: 50434930 50434930
[Wed Apr 27 07:13:21 2022]      PortMask:1 0 0 0 0 0 0 0
[Wed Apr 27 07:13:21 2022]      2x100G pcs shutdown 0
[Wed Apr 27 07:13:21 2022] set interface [cap0] mode (disabled) -> ()
[Wed Apr 27 07:13:21 2022]
[Wed Apr 27 07:13:21 2022] >
```

### config interface fec

FW: 8224+ ( 2x100G only)

This forces FEC on the specific capture port. FEC Autoneg is disabled. This setting is persistent across reboots.

```
config interface fec <interface>
```

```
Wed Aug 24 05:48:53 2022] > config interface fec cap0
[Wed Aug 24 05:48:53 2022]      Disable cycle calibration
[Wed Aug 24 05:48:53 2022]      FEC Force
[Wed Aug 24 05:48:53 2022]      FECENable: 1 PortMask:0001
[Wed Aug 24 05:48:53 2022]      [0] FECEnable: 1 FECForce:1
[Wed Aug 24 05:48:53 2022]      [1] FECEnable: 1 FECForce:1
[Wed Aug 24 05:48:53 2022] set interface [cap0] fec (true) -> (true)
[Wed Aug 24 05:48:53 2022]
[Wed Aug 24 05:48:53 2022] >
```

### config interface no fec

FW: 8224+ ( 2x100G only)

This disables the forced FEC setting where the system will try autoneg if FEC is enabled or not. Setting is persistent across reboots.

```
config interface no fec <interface>
```

```
[Wed Aug 24 05:50:39 2022] > config interface no fec cap0
[Wed Aug 24 05:50:39 2022]     Disable cycle calibration
[Wed Aug 24 05:50:39 2022]     no FEC Force
[Wed Aug 24 05:50:39 2022]     FECENable: 0 PortMask:0001
[Wed Aug 24 05:50:39 2022]     [0] FECENable: 1 FECForce:0
[Wed Aug 24 05:50:39 2022]     [1] FECENable: 1 FECForce:1

[Wed Aug 24 05:50:39 2022] set interface [cap0] fec (true) -> (false)
[Wed Aug 24 05:50:39 2022]
[Wed Aug 24 05:50:39 2022] >
```

## config interface ip

Configures the IP address of the specified network port. Typically this is used for setting the management/BMC IP address of the system

```
config interface ip <interface name> <IPv4 address>
```

Example below sets the man0 port to IP address 192.168.187.10. The exact output may vary between SKUs

```
config interface ip man0 192.168.187.10
```

```
[Tue Dec 13 04:22:47 2022] > config interface ip man0 192.168.187.10
[Tue Dec 13 04:22:48 2022] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocall
.
.
<snip>
.
.
[Tue Dec 13 04:22:51 2022] set interface [man0] ip (192.168.187.10) -> (192.168.187.10)
[Tue Dec 13 04:22:51 2022]
[Tue Dec 13 04:22:51 2022] Port          Mode          IP              Netmask         Gateway
[Tue Dec 13 04:22:51 2022] -----
[Tue Dec 13 04:22:53 2022] bmc          static        192.168.187.2   255.255.255.0   192.168.187
[Tue Dec 13 04:22:53 2022] man0         static        192.168.187.10 255.255.255.0   192.168.187
[Tue Dec 13 04:22:53 2022] man1         disabled      192.168.1.2     255.255.255.0   192.168.1.1
[Tue Dec 13 04:22:53 2022] man10        static        192.168.91.50   255.255.255.0
[Tue Dec 13 04:22:53 2022] cap0
[Tue Dec 13 04:22:53 2022] cap1
[Tue Dec 13 04:22:53 2022] -----
```

Example below shows setting the IP address of the BMC/IPMI port

```
config interface ip bmc 192.168.187.2
```

```
[Tue Dec 13 04:24:49 2022] > config interface ip bmc 192.168.187.2
[Tue Dec 13 04:24:50 2022] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocall --update
.
.
<snip>
.
.
[Tue Dec 13 04:26:00 2022] set interface [bmc] ip (192.168.187.2) -> (192.168.187.2)
[Tue Dec 13 04:26:00 2022]
[Tue Dec 13 04:26:00 2022] Port          Mode          IP              Netmask         Gateway
[Tue Dec 13 04:26:00 2022] -----
[Tue Dec 13 04:26:04 2022] bmc          static        192.168.187.2   255.255.255.0   192.168.187
[Tue Dec 13 04:26:04 2022] man0         static        192.168.187.10  255.255.255.0   192.168.187
[Tue Dec 13 04:26:04 2022] man1         disabled      192.168.1.2     255.255.255.0   192.168.1.1
[Tue Dec 13 04:26:04 2022] man10        static        192.168.91.50   255.255.255.0
[Tue Dec 13 04:26:04 2022] cap0
[Tue Dec 13 04:26:04 2022] cap1
[Tue Dec 13 04:26:04 2022] -----
[Tue Dec 13 04:26:04 2022] >
```

## config interface netmask

Sets the netmask of the specified interface

```
config interface netmask man0 255.255.255.0
```

```
[Tue Dec 13 04:28:02 2022] > config interface netmask man0 255.255.255.0
[Tue Dec 13 04:28:03 2022] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocall
.
.
<snip>
.
.
[Tue Dec 13 04:28:06 2022] set interface [man0] netmask (255.255.255.0) -> (255.255.255.0)
[Tue Dec 13 04:28:06 2022]
[Tue Dec 13 04:28:06 2022] Port          Mode          IP              Netmask         Gateway
[Tue Dec 13 04:28:06 2022] -----
[Tue Dec 13 04:28:07 2022] bmc          static        192.168.187.2   255.255.255.0   192.168.187
[Tue Dec 13 04:28:07 2022] man0         static        192.168.187.10  255.255.255.0   192.168.187
[Tue Dec 13 04:28:07 2022] man1         disabled      192.168.1.2     255.255.255.0   192.168.1.1
[Tue Dec 13 04:28:07 2022] man10        static        192.168.91.50   255.255.255.0
[Tue Dec 13 04:28:07 2022] cap0
[Tue Dec 13 04:28:07 2022] cap1
[Tue Dec 13 04:28:07 2022] -----
[Tue Dec 13 04:28:07 2022] >
```

## config interface gateway

Sets the default gateway for the specified interface

```
config interface gateway <interface> <IPv4 gateway address>
```

Example below sets the man0 management interfaces default gateway address

```
[Tue Dec 13 04:33:11 2022] > config interface gateway man0 192.168.187.30
[Tue Dec 13 04:33:11 2022] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocall
.
.
<snip>
.
.
[Tue Dec 13 04:33:14 2022] set interface [man0] Gateway (192.168.187.30) -> (192.168.187.30)
[Tue Dec 13 04:33:14 2022]
[Tue Dec 13 04:33:14 2022] Port          Mode      IP          Netmask     Gateway
[Tue Dec 13 04:33:14 2022] -----
[Tue Dec 13 04:33:17 2022] bmc      static    192.168.187.2  255.255.255.0  192.168.187
[Tue Dec 13 04:33:17 2022] man0     static    192.168.187.10 255.255.255.0  192.168.187
[Tue Dec 13 04:33:17 2022] man1     disabled  192.168.1.2    255.255.255.0  192.168.1.1
[Tue Dec 13 04:33:17 2022] man10    static    192.168.91.50  255.255.255.0
[Tue Dec 13 04:33:17 2022] cap0
[Tue Dec 13 04:33:17 2022] cap1
[Tue Dec 13 04:33:17 2022] -----
[Tue Dec 13 04:33:17 2022] >
```

## config interface dns

Sets the DNS server for the specified interface

```
config interface <interface> <IPv4 DNS address>
```

Example below sets the DNS server for man0 interface to be 1.1.1.1

```
config interface dns man0 1.1.1.1
```

```

[Tue Dec 13 04:35:33 2022] > config interface dns man0 1.1.1.1
[Tue Dec 13 04:35:34 2022] UPDATING: sudo /opt/fmadio/bin/setup_network.lua --nocall
.
.
<snip>
.
.
[Tue Dec 13 04:35:37 2022] set interface [man0] DNS (nil) -> (1.1.1.1)
[Tue Dec 13 04:35:37 2022]
[Tue Dec 13 04:35:37 2022] Port          Mode          IP              Netmask         Gateway
[Tue Dec 13 04:35:37 2022] -----+-----
[Tue Dec 13 04:35:40 2022] bmc          static        192.168.187.2   255.255.255.0   192.168.187
[Tue Dec 13 04:35:40 2022] man0         static        192.168.187.10 255.255.255.0   192.168.187
[Tue Dec 13 04:35:40 2022] man1         disabled     192.168.1.2     255.255.255.0   192.168.1.1
[Tue Dec 13 04:35:40 2022] man10        static        192.168.91.50  255.255.255.0
[Tue Dec 13 04:35:40 2022] cap0
[Tue Dec 13 04:35:40 2022] cap1
[Tue Dec 13 04:35:40 2022] -----+-----
[Tue Dec 13 04:35:40 2022] >

```

## Configure Capture

### show capture status

Shows the current capture status

```

Sun Jan 15 11:21:17 2023] > show capture status
[Sun Jan 15 11:21:17 2023]
[Sun Jan 15 11:21:17 2023] Current Capture Status
[Sun Jan 15 11:21:17 2023] -----+-----
[Sun Jan 15 11:21:17 2023] Capture Running      | true
[Sun Jan 15 11:21:17 2023] Capture Name         | asdf_20230115_1041
[Sun Jan 15 11:21:17 2023] Capture Bytes        | 0
[Sun Jan 15 11:21:17 2023] Capture Packets      | 0
[Sun Jan 15 11:21:17 2023] Capture Drop         | 0
[Sun Jan 15 11:21:17 2023] Capture FCS Error    | 0
[Sun Jan 15 11:21:17 2023] Capture Rate         | 0.000000 Gbps
[Sun Jan 15 11:21:17 2023]                      | 0.000000 MPps
[Sun Jan 15 11:21:17 2023] Capture Start        |
[Sun Jan 15 11:21:17 2023] Capture Duration     |
[Sun Jan 15 11:21:17 2023] -----+-----
[Sun Jan 15 11:21:17 2023] >

```

### show capture schedule

Shows the current capture schedule

```
[Sun Jan 15 11:22:07 2023] > show capture schedule
[Sun Jan 15 11:22:07 2023]
[Sun Jan 15 11:22:07 2023] Scheduled Capture Status
[Sun Jan 15 11:22:07 2023]
[Sun Jan 15 11:22:07 2023] | 24/7 | Start |
[Sun Jan 15 11:22:07 2023] -----+-----+-----+-----+
[Sun Jan 15 11:22:07 2023] wan_colo0 | false | 00:00:00 | 24
[Sun Jan 15 11:22:07 2023] -----+-----+-----+-----+
[Sun Jan 15 11:22:07 2023] >
```

## show capture list

Displays list of all captures on the system

```
[Sun Jan 15 11:23:21 2023] > show capture list
[Sun Jan 15 11:23:21 2023] Showing captures
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221230_0000] 1310720 B (Wed . 18:04:08 . 28
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221228_0000] 29613621248 B (Wed . 18:03:58 . 28
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221227_0000] 3442999296 B (Tue . 23:59:35 . 27
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221226_0000] 16513236992 B (Mon . 23:59:38 . 26
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221225_0000] 1869873152 B (Sun . 23:59:42 . 25
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221224_0000] 6031671296 B (Sat . 23:59:39 . 24
[Sun Jan 15 11:23:21 2023] [wan_colo0_20221223_0000] 33109311488 B (Fri . 23:59:36 . 23
[Sun Jan 15 11:23:21 2023] >
```

## show capture roll

FW: 8367+

Shows the current capture roll setting

```
[Sun Jan 15 11:25:18 2023] > show capture roll
[Sun Jan 15 11:25:18 2023] Capture Roll Setting:
[Sun Jan 15 11:25:18 2023] Roll every 1.00 Hour
[Sun Jan 15 11:25:18 2023] >
```

## show capture flush

FW: 8367+

Shows the current capture flushing behaviour

```
[Sun Jan 15 11:25:55 2023] > show capture flush
[Sun Jan 15 11:25:56 2023] Capture Flush Setting:
[Sun Jan 15 11:25:56 2023] Periodic Flush: 10 Sec
[Sun Jan 15 11:25:56 2023] >
```

## config capture start <name>

Starts a capture with the specified name

```
[Sat Jan 21 01:39:12 2023] > config capture start test-capture
[Sat Jan 21 01:39:12 2023]
[Sat Jan 21 01:39:12 2023] Starting Quick Capture [test-capture]
[Sat Jan 21 01:39:12 2023] OK: [Sat Jan 21 01:39:12 2023] successfully started capture [test-capture]
[Sat Jan 21 01:39:12 2023] >
```

Use `config capture status` to verify the current state

## config capture stop

Stops the currently active capture

NOTE: This will only stop captures manually started, for scheduled captures please disable the schedule entry to stop the capture

```
[Sat Jan 21 01:40:33 2023] > config capture stop
[Sat Jan 21 01:40:33 2023]
[Sat Jan 21 01:40:33 2023] Stopping Capture
[Sat Jan 21 01:40:33 2023] OK: [Sat Jan 21 01:40:33 2023] successfully stopped capture [test-capture]
[Sat Jan 21 01:40:33 2023] >
```

Use `config capture status` to verify the current state

## config capture flush

FW: 8367+

Sets the capture flushing behavior.

Default setting is flush 1sec after capture is idle

Flush always every 1 second. NOTE: 1sec is very aggressive mode and will consume additional storage. However it does provide low latency when watching low bandwidth captures.

```
// Some code
[Sun Jan 15 11:27:34 2023] > config capture flush period 1
[Sun Jan 15 11:27:35 2023] Setting Flush Mode[period] Timeout 1 sec
[Sun Jan 15 11:27:35 2023]
[Sun Jan 15 11:27:35 2023] **** requires restarting of capture to take effect ****
[Sun Jan 15 11:27:35 2023]
```



Flush when capture is idle for >= 1sec

```
[Sun Jan 15 11:27:44 2023] > config capture flush idle 10
[Sun Jan 15 11:27:44 2023] Setting Flush Mode[idle] Timeout 10 sec
[Sun Jan 15 11:27:44 2023]
[Sun Jan 15 11:27:44 2023] **** requires restarting of capture to take effect ****
[Sun Jan 15 11:27:44 2023]
[Sun Jan 15 11:27:44 2023] >
```

## config capture roll

FW: 8367+

Configures the capture rolling behavior. Default (0) is roll at midnight.

Example configures capture to roll every 1 hour.

```
[Sun Jan 15 11:31:46 2023] > config capture roll 1
[Sun Jan 15 11:31:47 2023] Setting Capture Roll Every 1 Hour
[Sun Jan 15 11:31:47 2023]
[Sun Jan 15 11:31:47 2023] **** requires restarting of capture to take effect ****
[Sun Jan 15 11:31:47 2023]
[Sun Jan 15 11:31:47 2023] >
```

## Configure PCAP Download

### show pcap timestamp

Shows the current PCAP timestamp mode. e.g. from the FMADIO FPGA or extract timing information from a packet broker

Example below shows the PCAP timestamp uses the Arista 7130 (Metamako) footer timestamp

```
show pcap timestamp
```

```
[Mon Jun 12 16:48:09 2023] > show pcap timestamp
[Mon Jun 12 16:48:10 2023] TimeStamp Mode: arista7130 : (Arista 7130 (Metamako))
[Mon Jun 12 16:48:10 2023]
[Mon Jun 12 16:48:10 2023] >
```

### config pcap timestamp <tsmode>

Configures the default PCAP timestamp mode when downloading PCAP data. this value can be overridden by URI option TSMODE.

## Supported Timestamp Modes

- nic - FMADIO FPGA timestamp
- arista7130 - Arista 7130 (Metamako) Footer
- arista7150\_overwrite - Arista 7150 Overwrite FCS + Keyframes
- arista7150\_insert - Arista 7150 Insert 32bit + Keyframes
- arista7280\_mac48 - Arista 7280 Source MAC 48bit Overwrite
- arista7280\_eth64 - Arista 7280 Ethernet Insert 64bit
- erspanv3 - Cisco ERSPANv3 Encapsulation
- cisco3550 - Cisco 3550 (Exablaze) Footer

## Help command

```
[Mon Jun 12 16:50:30 2023] > config pcap timestamp
[Mon Jun 12 16:50:31 2023] Example Usage:
[Mon Jun 12 16:50:31 2023] > config pcap timestamp <mode>           : config
[Mon Jun 12 16:50:31 2023]                                     arista7280_mac48       : Arista
[Mon Jun 12 16:50:31 2023]                                     arista7130           : Arista
[Mon Jun 12 16:50:31 2023]                                     arista7280_eth64    : Arista
[Mon Jun 12 16:50:31 2023]                                     arista7150_overwrite : Arista
[Mon Jun 12 16:50:31 2023]                                     erspanv3             : CISCO
[Mon Jun 12 16:50:31 2023]                                     arista7150_insert   : Arista
[Mon Jun 12 16:50:31 2023]                                     nic                   : Timest
[Mon Jun 12 16:50:31 2023]                                     cisco3550           : CISCO
[Mon Jun 12 16:50:31 2023]
[Mon Jun 12 16:50:31 2023] ERROR: Unknown Command [config pcap timestamp]
[Mon Jun 12 16:50:31 2023] >
```

Example to set the default behavior to use Arista 7130 footer. It takes 60sec to restart the processes after the setting.

```
config pcap timestamp arista7130
```

```

[Mon Jun 12 16:54:04 2023] > config pcap timestamp arista7130
[Mon Jun 12 16:54:04 2023] TimeStamp Mode set to [arista7130] : (Arista 7130 (Metamako))
[Mon Jun 12 16:54:04 2023]
[Mon Jun 12 16:54:04 2023] Restarting Processes
[Mon Jun 12 16:54:14 2023] Wait 60sec for processes to restart
[Mon Jun 12 16:54:14 2023]          0/60
[Mon Jun 12 16:54:15 2023]          1/60
[Mon Jun 12 16:54:16 2023]          2/60
[Mon Jun 12 16:54:17 2023]          3/60
[Mon Jun 12 16:54:18 2023]          4/60
.
.

[Mon Jun 12 16:55:07 2023]          53/60
[Mon Jun 12 16:55:08 2023]          54/60
[Mon Jun 12 16:55:09 2023]          55/60
[Mon Jun 12 16:55:10 2023]          56/60
[Mon Jun 12 16:55:11 2023]          57/60
[Mon Jun 12 16:55:12 2023]          58/60
[Mon Jun 12 16:55:13 2023]          59/60
[Mon Jun 12 16:55:14 2023]          60/60
[Mon Jun 12 16:55:15 2023] done
[Mon Jun 12 16:55:15 2023] >

```

## LXC Container Management

Following provides commands for configuration and managing LXC containers on the system

### show lxc status

Shows the current LXC container status of the system

```
show lxc status
```

Example shows 2 containers "suricata" and "centos7"

Suricata is enabled to start at boot time.

```

[Sat Jun 24 15:39:35 2023] > show lxc status
[Sat Jun 24 15:39:35 2023]
[Sat Jun 24 15:39:35 2023] Name                OnBoot    Install   State     Desc
[Sat Jun 24 15:39:35 2023] -----
[Sat Jun 24 15:39:35 2023] suricata          true      yes      STOPPED  suricata id
[Sat Jun 24 15:39:35 2023] centos7           false     yes      STOPPED
[Sat Jun 24 15:39:35 2023] -----
[Sat Jun 24 15:39:35 2023] >

```

### **config lxc add <lxc name>**

Adds an already installed container to the configuration file

```
config lxc add <lxc name>
```

In the example below adding an already installed container named "ubuntu20" to the system

```
[Sat Jun 24 15:45:56 2023] > config lxc add ubuntu20  
[Sat Jun 24 15:45:57 2023] Added container [ubuntu20] to the configuration  
[Sat Jun 24 15:45:57 2023] >
```

### **config lxc del <lxc name>**

Removes the specified container from the configuration

NOTE it does not delete the container on disk. Only removes it from the configuration files

```
config lxc del <lxc name>
```

Example deletes the container "ubuntu20" from the configuration files

```
[Sat Jun 24 15:48:24 2023] > config lxc del ubuntu20  
[Sat Jun 24 15:48:25 2023] Removed container [ubuntu20] to the configuration  
[Sat Jun 24 15:48:25 2023] >
```

### **config lxc desc <lxc name> "<description>"**

Adds a human description to the lxc to provide context

```
config lxc desc <lxc name> "<description>"
```

Example set a human readable description for the container "ubuntu20" this is visible when using the show lxc command

```
[Sat Jun 24 15:51:01 2023] > config lxc desc ubuntu20 "general purpose ubuntu20 container"  
[Sat Jun 24 15:51:01 2023] Set container [ubuntu20] to description to [general purpose ubuntu20]  
[Sat Jun 24 15:51:02 2023] >
```

### **config lxc boot <lxc name>**

Sets the specified container to boot on startup

```
config lxc boot <lxc name>
```

Example set the "ubuntu20" container to boot on system startup

```
[Sat Jun 24 15:53:06 2023] > config lxc boot ubuntu20
[Sat Jun 24 15:53:07 2023] Set container [ubuntu20] to boot on system start
[Sat Jun 24 15:54:11 2023] >
```

### **config lxc no-boot <lxc name>**

Sets the specified LXC container to not boot at startup

```
config lxc no-boot <lxc name>
```

Example sets the LXC container "ubuntu20" to not boot on startup

```
[Sat Jun 24 15:57:50 2023] > config lxc no-boot ubuntu20
[Sat Jun 24 15:57:50 2023] Set container [ubuntu20] to NOT boot on system start
[Sat Jun 24 15:57:50 2023] >
```

### **config lxc start <lxc name>**

Starts the specified container named <lxc name> if the container starts successfully system will return back to the prompt

```
config lxc start <lxc name>
```

Example starts the fshark2 container on the system

```
[Sat Jun 24 16:03:05 2023] > config lxc start fshark2
[Sat Jun 24 16:03:05 2023] sudo lxc-start -n fshark2 --logfile /tmp/lxc_fshark2_168759378584
[Sat Jun 24 16:03:07 2023]
[Sat Jun 24 16:03:07 2023] use the following on a shell to attach to the conatiners console
[Sat Jun 24 16:03:07 2023] sudo lxc-attach -n fshark2
[Sat Jun 24 16:03:07 2023]
[Sat Jun 24 16:03:07 2023] >
```

When a container fails to start the output will look similar to below.

```
[Sat Jun 24 16:04:24 2023] > config lxc start ubuntu20
[Sat Jun 24 16:04:24 2023] sudo lxc-start -n ubuntu20 --logfile /tmp/lxc_ubuntu20_1687593864
[Sat Jun 24 16:04:24 2023] lxc: lxc-start: ubuntu20: tools/lxc_start.c: main: 322 Executing
[Sat Jun 24 16:04:24 2023] lxc:          lxc-start ubuntu20 20230624080424.927 ERROR    lxc_sta
[Sat Jun 24 16:04:24 2023]
[Sat Jun 24 16:04:24 2023] >
```

### **config lxc stop <lxc name>**

Stops the specified container from running

```
config lxc stop <lxc name>
```

Example stops the fshark2 container running

```
[Sat Jun 24 16:06:49 2023] > config lxc stop fshark2
[Sat Jun 24 16:06:50 2023] sudo lxc-stop -n fshark2 --logfile /tmp/lxc_fshark2_1687594010169
[Sat Jun 24 16:06:50 2023]
```

### **config lxc list**

**TBD** lists all available containers in the public repo

contact [support@fmad.io](mailto:support@fmad.io) for more info

### **config lxc install**

**TBD** installs the specified container from the public repo

contact [support@fmad.io](mailto:support@fmad.io) for more info

### **config lxc uninstall**

**TBD** removes the specified container from the configuration and disk

contact [support@fmad.io](mailto:support@fmad.io) for more info

## **Automatic Push PCAP**

Configure and monitor the automatic push generation of PCAPs to storage locations.

### **show push pcap status**

FW: 7963+

Shows the currently configured automatic push pcaps

```

[Thu Jun 9 08:30:42 2022] > show push pcap status
[Thu Jun 9 08:30:42 2022] -----
[Thu Jun 9 08:30:42 2022] FollowStart : true
[Thu Jun 9 08:30:42 2022] Decap      : true
[Thu Jun 9 08:30:42 2022] -----
[Thu Jun 9 08:30:42 2022] [tcp] #1
[Thu Jun 9 08:30:42 2022] Mode      : File
[Thu Jun 9 08:30:42 2022] Path      : /mnt/remote0/push/ttest/
[Thu Jun 9 08:30:42 2022] Split     : --split-time 3600e9
[Thu Jun 9 08:30:42 2022] FileName  : --filename-tstr-HHMMSS
[Thu Jun 9 08:30:42 2022] FilterBPF : host 192.168.1.1
[Thu Jun 9 08:30:42 2022] FilterFrame :
[Thu Jun 9 08:30:42 2022] -----
[Thu Jun 9 08:30:42 2022] >

```

### **config push pcap add <push target>**

FW: 7963+

Creates a new push pcap target called <push target>

NOTE: all target names should be unique

```

[Thu Jun 9 08:32:13 2022] > config push pcap add udp-all
[Thu Jun 9 08:32:13 2022] Add Push PCAP target [udp-all]
[Thu Jun 9 08:32:13 2022] >

```

### **config push pcap del <push target>**

Deletes the current push pcap entry name <push target>

```

[Thu Jun 9 08:33:17 2022] > config push pcap del udp-all
[Thu Jun 9 08:33:18 2022] deleting: [udp-all] row 2
[Thu Jun 9 08:33:18 2022] >

```

### **config push pcap name <push target> <new name>**

Renames the specified <push target> entry with an updated one <new name>

```

[Thu Jun 9 08:36:21 2022] > config push pcap name udp-all udp-port-1900
[Thu Jun 9 08:36:23 2022] Rename [udp-all] -> [udp-port-1900]
[Thu Jun 9 08:36:23 2022] >

```

### config push pcap path <push target> <new write path>

Updates the push write path to the specified <new write path>. Typically this is the NFS remote path or rclone write path.

```
[Thu Jun 9 08:41:41 2022] > config push pcap path udp-port-1900 /mnt/remote0/push/  
[Thu Jun 9 08:41:44 2022] Set Path [] -> [/mnt/remote0/push/]  
[Thu Jun 9 08:41:44 2022] >
```

### config push pcap split-time <push target> <value>

Configure PCAPs to be split by the specified time value. By default <value> is scientific notation in nanoseconds. In addition s (seconds) m (minutes) h (hours) suffix can be used also

Text	Description
1e9	1 second in scientific notation
60s	60 seconds
1m	1 minute
1h	1 hour

Example configure to split every 1 minute

```
[Thu Jun 9 08:48:05 2022] > config push pcap split-time udp-port-1900 1m  
[Thu Jun 9 08:48:06 2022] Set Split to [--split-time 3600e9] -> [--split-time 600000000000]  
[Thu Jun 9 08:48:06 2022] >
```

### config push pcap split-size <push target> <value>

Configure PCAPs to be split by total byte size <value>

Text	Description
1e9	1 Gigabyte specified in scientific notation
100M	100 Megabyte
5G	5 Gigabyte

Example below shows splitting on 1GB boundaries



```
[Thu Jun 9 08:53:32 2022] > config push pcap split-size udp-port-1900 1G
[Thu Jun 9 08:53:33 2022] Set Split to [--split-time 60000000000] -> [--split-byte 1000000000]
[Thu Jun 9 08:53:33 2022] >
```

### config push pcap filename <push target> <value>

Specifies the filename format for each individual split PCAP

Value	Example	Description
epoch-sec	_1654610221.pcap	Second Epoch
epoch-sec-startend	_1654610221-1654620221.pcap	Epoch start and End
epoch-msec	_1654610221012.pcap	Epoch in msec
epoch-usec	_1654610221012345.pcap	Epoch is micro sec
epoch-nsec	_1654610221012345678.pcap	Epoch in Nano sec
HHMM	_20200101_1201.pcap	Hour Min
HHMMSS	_20200101_120159.pcap	Hour Min Sec
HHMMSS_TZ	2020-01-01_12:01:59+09:00.pcap	House Min Sec + Timezone
HHMMSS_NS	_20200101_120159.012345678.pcap	House Min Sec Nan

Example uses a simple Hour Min Sec format

```
[Thu Jun 9 09:02:52 2022] > config push pcap filename udp-port-1900 HHMMSS
[Thu Jun 9 09:02:53 2022] Set Filename to [--filename-tstr-HHMMSS_TZ] -> [--filename-tstr-2020-01-01_12:01:59+09:00]
[Thu Jun 9 09:02:53 2022] >
```

### config push pcap filter-bpf <push target> "<bpf filter>"

Sets the specified push with a BPF filter.

**NOTE: the BPF filter must be enclosed in double quotes**

Example sets for udp and port 1900

```
[Thu Jun 9 09:07:05 2022] > config push pcap filter-bpf udp-port-1900 "udp and port 1900"  
[Thu Jun 9 09:07:05 2022] Set FilterBPF [] -> [udp and port 1900]  
[Thu Jun 9 09:07:05 2022] >
```

## config push pcap restart

Shutdown the current push processes and restarts them

```
config push pcap restart
```

Example output

```
[Thu Sep 28 12:18:53 2023] > config push pcap restart  
Killing 78246 {push_pcap.lua}  
Killing 78403 /opt/fmadio/bin/stream_cat  
Killing 78406 /opt/fmadio/bin/pcap_split  
Killing 78412 /opt/fmadio/bin/pcap_split  
wait for respawn 0/120  
wait for respawn 1/120  
wait for respawn 2/120  
wait for respawn 3/120  
wait for respawn 4/120  
wait for respawn 5/120  
wait for respawn 6/120  
wait for respawn 7/120  
wait for respawn 8/120  
wait for respawn 9/120  
wait for respawn 10/120  
process respawned  
[Thu Sep 28 12:19:05 2023] >
```

## Automatic Push to LXC (Container)

The system can push automatically into a lxc\_ring enabling a container to consume the data. These functions are to add/delete/modify these push functions.

NOTE this requires the push\_lxc analytics script to be running

### show push lxc

Shows the current push lxc targets configured on the system

```
show push lxc
```

Example shown below, indicates a single suricata ring is enabled with a BPF filter to remove all traffic from subnet 192.168.100.0/24

```
[Sat Jun 24 14:22:12 2023] > show push lxc
[Sat Jun 24 14:22:12 2023]
[Sat Jun 24 14:22:12 2023] Ring name                               : Enable :
[Sat Jun 24 14:22:12 2023] -----+-----+-----
[Sat Jun 24 14:22:12 2023] /opt/fmadio/queue/lxc_ring_suricata           :   true :   s
[Sat Jun 24 14:22:12 2023] -----+-----+-----
[Sat Jun 24 14:22:12 2023] >
```

### **config push lxc add <ring name>**

This adds a new LXC push to the ring named <ring name>.

By default the push is disabled when created.

```
config push lxc add <ring name>
```

Example below shows adding a push to the ring named "general"

```
[Sat Jun 24 14:25:26 2023] > config push lxc add general
[Sat Jun 24 14:25:26 2023] New Push LXC target [/opt/fmadio/queue/lxc_ring_general]
[Sat Jun 24 14:25:26 2023] >
```

NOTE this does not create the ring, it only creates the push to the specified ring

### **config push lxc del <ring name>**

Removes the specified LXC push target

```
config push lxc del <ring name>
```

Example removes the push lxc target named "general"

```
[Sat Jun 24 14:32:31 2023] > config push lxc del general
[Sat Jun 24 14:32:32 2023] Delete Push LXC target [/opt/fmadio/queue/lxc_ring_general]
[Sat Jun 24 14:32:32 2023] >
```

### **config push lxc enable <ring name>**

Enables the specified lxc ring push target. By default when adding a new target the state is disabled.

```
config push lxc enable <ring name>
```

Example enables the push lxc ring target named "general"

```
[Sat Jun 24 14:36:29 2023] > config push lxc enable general
```

```
[Sat Jun 24 14:36:29 2023] Set LXC target [/opt/fmadio/queue/lxc_ring_general] Enable
```

```
[Sat Jun 24 14:36:29 2023] >
```

### **config push lxc disable <ring name>**

Disables the specified lxc push <ring name>

```
config push lxc disable <ring name>
```

Example disables the lxc ring named "general"

```
[Sat Jun 24 14:39:44 2023] > config push lxc disable general
```

```
[Sat Jun 24 14:39:45 2023] Set LXC target [/opt/fmadio/queue/lxc_ring_general] to Disable
```

```
[Sat Jun 24 14:39:45 2023] >
```

### **config push lxc filter-bpf <ring name> "<filter bpf>"**

Adds the specified BPF filter to the LXC push to the ring.

NOTE The filter must be enclosed in double quotes ""

```
config push lxc filter-bpf <ring name> "<filter bpf>"
```

Example adds a subnet "192.168.0.0/24" filter to the ring named "general"

```
[Sat Jun 24 14:44:11 2023] > config push lxc filter-bpf general "net 192.168.0.0/24"
```

```
[Sat Jun 24 14:44:12 2023] Set LXC target [/opt/fmadio/queue/lxc_ring_general] filter bpf to
```

```
[Sat Jun 24 14:44:12 2023] >
```

### **config push lxc filter-frame <ring name> "<filter frame>"**

Adds the specified frame filter to the lxc push to the ring.

NOTE the filter must be enclosed in double quotes ""

```
config push lxc filter-frame <ring name> "<filter frame>"
```

Example add a Frame filter of capture port 0 only to the ring named "general"

```
[Sat Jun 24 14:46:56 2023] > config push lxc filter-frame general "capture.port=0"  
[Sat Jun 24 14:46:56 2023] Set LXC target [/opt/fmadio/queue/lxc_ring_general] filter frame  
[Sat Jun 24 14:46:56 2023] >
```

### **config push lxc from-now <ring name>**

Sets the push to start from the current capture position into the lxc ring.

This is the default behaviour

```
config push lxc from-now <ring name>
```

Example sets the ring "general" to push data into the ring from now.

```
[Sat Jun 24 14:50:03 2023] > config push lxc from-now general  
[Sat Jun 24 14:50:04 2023] Set LXC target [/opt/fmadio/queue/lxc_ring_general] fetch from cu  
[Sat Jun 24 14:50:04 2023] >
```

### **config push lxc from-start <ring name>**

Sets the push to start from the beginning of the capture.

```
config push lxc from-start <ring name>
```

Example sets the ring "general" to start from the beginning of the capture

```
[Sat Jun 24 14:51:48 2023] > config push lxc from-start general  
[Sat Jun 24 14:51:48 2023] Set LXC target [/opt/fmadio/queue/lxc_ring_general] fetch from st  
[Sat Jun 24 14:51:48 2023] >
```

### **config push lxc restart**

This shutdowns and then restarts the push lxc processes.

```
config push lxc restart
```

Example output

```
[Thu Sep 28 12:14:27 2023] > config push lxc restart
Killing 78263 {push_lxc.lua}
Killing 78376 /opt/fmadio/bin/stream_cat
wait for respawn 0/120
wait for respawn 1/120
wait for respawn 2/120
wait for respawn 3/120

wait for respawn 4/120
wait for respawn 5/120
wait for respawn 6/120
.
.
wait for respawn 29/120
wait for respawn 30/120
wait for respawn 31/120
wait for respawn 32/120
wait for respawn 33/120
wait for respawn 34/120
wait for respawn 35/120
wait for respawn 36/120
process respawned
[Thu Sep 28 12:15:05 2023] >
```

## Ring management

Various functions for monitoring the ring status both push pcap and push lxc

### show ring status

Shows all rings status information

```
show ring status
```

Example, this can be helpful for monitoring data is being produced and consumed correctly.

```

> show ring status
Name                               : Path                               : St
-----+-----+-----+
lxc_ring_suricata                  : /opt/fmadio/queue/lxc_ring_suricata : on
pcap_ring_all                      : /opt/fmadio/queue/pcap_ring_all     : on
pcap_ring_aws-sg2                  : /opt/fmadio/queue/pcap_ring_aws-sg2 : on
pcap_ring_dns                      : /opt/fmadio/queue/pcap_ring_dns     : on
pcap_ring_icmp                    : /opt/fmadio/queue/pcap_ring_icmp    : on
pcap_ring_ssh                     : /opt/fmadio/queue/pcap_ring_ssh     : on
pcap_ring_vlan-4091               : /opt/fmadio/queue/pcap_ring_vlan-4091 : on
pcap_ring_voip-hk                  : /opt/fmadio/queue/pcap_ring_voip-hk  : on
-----+-----+-----+
>

```

## Time

Various functions for configuration and monitoring time

### show timezone

Shows the current timezone the system is configured

```
show timezone
```

Example showing the current timezone

```

[Sat Jun 24 15:25:29 2023] > show timezone
[Sat Jun 24 15:25:30 2023] Timezone: Asia/Singapore
[Sat Jun 24 15:25:30 2023]      UTC +08:00 (SGT)
[Sat Jun 24 15:25:32 2023] >

```

### config timezone "<city>"

Configures the timezone by searching the timezone list for the location named "<city>"

System uses the first found match

For cities with spaces in the name, ensure to use double quotes around the city name

```
config timezone "<city>"
```

Example set the timezone to New York

```
[Sat Jun 24 15:32:00 2023] > config timezone "New York"
[Sat Jun 24 15:32:01 2023] setting timezone to [/usr/share/zoneinfo/America/New_York]
[Sat Jun 24 15:32:01 2023]      UTC -04:00 M (EDT)
[Sat Jun 24 15:32:01 2023] *** System requires a reboot to take effect ***

[Sat Jun 24 15:32:01 2023] >
```

**NOTE** change only takes effect on next reboot

## User Management

**FW: 8336+**

FMADIO Web GUI supports multiple users with 2 levels of access

Permission	Description
full	Provides full admin level access to all functions
user	User level only, no ability to modify config and start/stop captures

Using `fmadiocli` to setup and configure is shown below

### **show userlist**

This shows the currently configured list of users on the system

```
show userlist
```

Example output, it shows 2 users `fmadio` (full access), `bob` (user access)



```
[Tue Dec 13 04:05:54 2022] > show userlist
[Tue Dec 13 04:05:55 2022] Showing User List
[Tue Dec 13 04:05:55 2022]
[Tue Dec 13 04:05:55 2022] UserList Enable: true
[Tue Dec 13 04:05:55 2022]
[Tue Dec 13 04:05:55 2022] -----
[Tue Dec 13 04:05:55 2022]
[Tue Dec 13 04:05:55 2022]   UserName   : fmadio
[Tue Dec 13 04:05:55 2022]   Permission : full
[Tue Dec 13 04:05:55 2022]   SecBPF     :
[Tue Dec 13 04:05:55 2022]
[Tue Dec 13 04:05:55 2022]   UserName   : bob
[Tue Dec 13 04:05:55 2022]   Permission : user
[Tue Dec 13 04:05:55 2022]   SecBPF     :
[Tue Dec 13 04:05:55 2022]
[Tue Dec 13 04:05:55 2022] -----
[Tue Dec 13 04:05:55 2022] >
```

### config userlist add

Adds a new user with default permissions and no password

```
config userlist add <username>
```

Example adds the username "bob" to the system

```
[Tue Dec 13 04:08:14 2022] > config userlist add bob
[Tue Dec 13 04:08:17 2022] Created new User [bob]
[Tue Dec 13 04:08:17 2022] >
```

### config userlist del

Deletes the specified username

```
config userlist del <username>
```

Example below deletes the username "bob"

```
[Tue Dec 13 04:09:15 2022] > config userlist del bob
[Tue Dec 13 04:09:16 2022] deleted username [bob]
[Tue Dec 13 04:09:16 2022] >
```

### config userlist password

Sets the WEB user password. This has no effect on SSH access to the system

```
config userlist password <username>
```

Example below sets the web password for user "bob"

```
[Tue Dec 13 04:12:19 2022] > config userlist password bob
[Tue Dec 13 04:12:20 2022] New Password      : *****
[Tue Dec 13 04:12:22 2022] Re-enter Password: *****
[Tue Dec 13 04:12:24 2022] web password for username [bob] set
[Tue Dec 13 04:12:24 2022] >
```

### config userlist permission

Sets the userlevel permission for the specified username

```
config userlist permission <username> <level>
```

Level types are 2

- full - provides full unrestricted GUI access
- user - provides download and analysis only access (no configuration or capture state change)

Example below shows setting the username "bob" to be a "user" level (e.g. can not change system configuration or capture states)

```
[Tue Dec 13 04:13:55 2022] > config userlist permission bob user
[Tue Dec 13 04:13:55 2022] modified username [bob] to permission level [user]
[Tue Dec 13 04:13:55 2022] >
```

Example below shows setting username "bob" to be a full access user (e.g. can change any configuration using the GUI)

```
[Tue Dec 13 04:14:42 2022] > config userlist permission bob full
[Tue Dec 13 04:14:42 2022] modified username [bob] to permission level [full]
[Tue Dec 13 04:14:42 2022] >
```

## Security Management

Various commands to set and modify the security settings of the system

### show security

Shows the current security settings

```
show security
```

```
[Wed May 17 13:56:44 2023] > show security
[Wed May 17 13:56:44 2023] Authentication: PAM-LDAP
[Wed May 17 13:56:44 2023] HTTP Access    : enabled
[Wed May 17 13:56:44 2023] Timeout SSH   : 0.166667min (idle)
[Wed May 17 13:56:44 2023] Timeout WWW   : 1.000000min (session)
[Wed May 17 13:56:44 2023] >
```

### config security auth

This sets the authentication method of the system. Number of options as follows

- BASIC - this is basic authentication, low security level
- OAUTH - OAUTH 2.0 including Active Directory, Google, Ping Identity
- RADIUS - Use Radius based authentication
- PAM-LDAP - Use the linux PAM system with an LDAP authentication mode

Example to set PAM-LDAP as follows

```
config security auth pam-ldap
```

Output as follows

```
[Fri Mar 3 14:24:34 2023] > config security auth pam-ldap
[Fri Mar 3 14:24:35 2023] Authentication [BASIC] -> [PAM-LDAP]
[Fri Mar 3 14:24:35 2023] rebooting the system may be required
[Fri Mar 3 14:24:35 2023] >
```

For some authentication methods it requires a system reboot. In this case a reboot is required as the system needs to start LDAP client daemons.

### config security http

This enables/disables HTTP as a mode of access to the device. HTTP is plain clear text transport protocol, meaning all private data such as username and password are sent in the clear.

For private and secure networks this is ok(ish) for most situations HTTP should be disabled, allowing only HTTPS as the mode of access.

To disable HTTP access (HTTPS only)

```
config security http false
```

Example output

```
[Fri Mar 3 14:27:34 2023] > config security http false  
[Fri Mar 3 14:27:34 2023] HTTP Access [enable] -> [false]  
[Fri Mar 3 14:27:35 2023] please wait 60sec for web access to restart  
[Fri Mar 3 14:27:35 2023] >
```

### **config security timeoutSSH**

This sets the SSH idle timeout value. Use "show security" to validate the value is correct.

Time units supported are

```
s - second  
m - minute  
h - hour  
disable - disable timeout
```

An example of setting a 30 sec idle timeout as follows

```
config security timeoutSSH 30s
```

With the following output

```
[Wed May 17 13:57:23 2023] > config security timeoutSSH 30s  
[Wed May 17 13:57:24 2023] SSH Timeout [10000000000] -> [30000000000]  
[Wed May 17 13:57:24 2023] please reboot for new setting to take effect  
[Wed May 17 13:57:24 2023] >
```

NOTE: the system requires a reboot for the changes to take effect.

### **config security timeoutWWW**

This sets the WWW session timeout value. This is the maximum session duration. Once the session duration is reached the web interface will require a re-login.

Time units supported are

```
// Some code  
s - second  
m - minute  
h - hour  
disable - disable timeout
```

An example setting a 1 hour maximum session timeout

```
config security timeoutWWW 1h
```

Example output

```
[Wed May 17 14:00:35 2023] > config security timeoutWWW 1h
[Wed May 17 14:00:35 2023] WWW Timeout [600000000000] -> [36000000000000]
[Wed May 17 14:00:35 2023] please reboot for new setting to take effect
[Wed May 17 14:00:35 2023] >
```

NOTE: a the system requires a reboot for the changes to take effect.

## Disk Management

Configuration and status information for disks and disk encryption

### show disk status

Shows the current disk status information

```
show disk status
```

Example below shows a fully setup 100Gp3 system with PSID and Encryption enabled

```
[Sat Jun 24 17:48:45 2023] > show disk status
[Sat Jun 24 17:48:45 2023] SSD Cache
[Sat Jun 24 17:48:46 2023] Disk :          Serial :    Size : Temp  : Used : Error
[Sat Jun 24 17:48:46 2023] -----+-----+-----+-----+-----+-----
[Sat Jun 24 17:48:46 2023] os0   :      50026B7685513F33 : 0.00 TB :  0 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] par0  :      22443E9D2087 : 0.00 TB : 30 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd0  :      22443E9D204F : 0.15 TB : 29 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd1  :      22223AD5BFC3 : 0.15 TB : 32 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd2  :      22443E9D3AFF : 0.15 TB : 28 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd3  :      22443E9DC543 : 0.15 TB : 30 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd4  :      22443E9D2076 : 0.15 TB : 29 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd5  :      22443E9D3B41 : 0.15 TB : 29 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd6  :      22443E9D3B65 : 0.15 TB : 30 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd7  :      22443E9D20A4 : 0.15 TB : 28 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] ssd8  :      22443E9DC54E : 0.15 TB : 31 C :  0 % :    0
[Sat Jun 24 17:48:46 2023] -----+-----+-----+-----+-----+-----
[Sat Jun 24 17:48:46 2023] >
```

### config disk sanitize

Using TCG OPAL2 sedutils the system will factory reset the device using the PSID values, initialize the drives for encryption and set a default password.

When complete the drives data is encrypted with a default password to access a randomly generated AES256 encryption key.

When complete the drives are in the unlocked state. To enable locking use the `config disk lock` comand

```
config disk sanitize
```

Example shows a partial log of the 100G systems sanitize operation. Entire operation takes about 60 seconds

```
[Sat Jun 24 17:53:32 2023] > config disk sanitize
[Sat Jun 24 17:53:32 2023] -----
[Sat Jun 24 17:53:32 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme9n1
[Sat Jun 24 17:53:32 2023] [par0] /dev/nvme9n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:53:32 2023]
[Sat Jun 24 17:53:32 2023] [par0] factory reset 22443E9D2087
[Sat Jun 24 17:53:32 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --PSIDrevert 11EDDF5767CA
[Sat Jun 24 17:53:35 2023] [par0] revertTper completed successfully
[Sat Jun 24 17:53:35 2023]
[Sat Jun 24 17:53:35 2023] [par0] set default password
[Sat Jun 24 17:53:35 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --initialSetup ***** /dev
[Sat Jun 24 17:53:35 2023] [par0] SID password changed
[Sat Jun 24 17:53:36 2023] [par0] takeOwnership complete
[Sat Jun 24 17:53:37 2023] [par0] Locking SP Activate Complete
[Sat Jun 24 17:53:38 2023] [par0] LockingRange0 disabled
[Sat Jun 24 17:53:38 2023] [par0] LockingRange0 set to RW
[Sat Jun 24 17:53:39 2023] [par0] MBRDone set on
[Sat Jun 24 17:53:40 2023] [par0] MBREnable set on
[Sat Jun 24 17:53:40 2023] [par0] Initial setup of TPer complete on /dev/nvme9n1
[Sat Jun 24 17:53:40 2023]
[Sat Jun 24 17:53:40 2023] [par0] disable MBR
[Sat Jun 24 17:53:40 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --setMBREnable off *****
[Sat Jun 24 17:53:40 2023] [par0] MBRDone set on
[Sat Jun 24 17:53:41 2023] [par0] MBREnable set off
[Sat Jun 24 17:53:41 2023] -----
[Sat Jun 24 17:53:41 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme10n
[Sat Jun 24 17:53:41 2023] [ssd0] /dev/nvme10n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:53:41 2023]
[Sat Jun 24 17:53:41 2023] [ssd0] factory reset 22443E9D204F
[Sat Jun 24 17:53:41 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --PSIDrevert 7306CD8CE0BE
[Sat Jun 24 17:53:43 2023] [ssd0] revertTper completed successfully
[Sat Jun 24 17:53:43 2023]
[Sat Jun 24 17:53:43 2023] [ssd0] set default password
[Sat Jun 24 17:53:43 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --initialSetup ***** /dev
[Sat Jun 24 17:53:44 2023] [ssd0] SID password changed
[Sat Jun 24 17:53:44 2023] [ssd0] takeOwnership complete
[Sat Jun 24 17:53:46 2023] [ssd0] Locking SP Activate Complete
[Sat Jun 24 17:53:47 2023] [ssd0] LockingRange0 disabled
[Sat Jun 24 17:53:47 2023] [ssd0] LockingRange0 set to RW
[Sat Jun 24 17:53:48 2023] [ssd0] MBRDone set on
[Sat Jun 24 17:53:49 2023] [ssd0] MBREnable set on
[Sat Jun 24 17:53:49 2023] [ssd0] Initial setup of TPer complete on /dev/nvme10n1
[Sat Jun 24 17:53:49 2023]
[Sat Jun 24 17:53:49 2023] [ssd0] disable MBR
[Sat Jun 24 17:53:49 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --setMBREnable off *****
[Sat Jun 24 17:53:49 2023] [ssd0] MBRDone set on
[Sat Jun 24 17:53:50 2023] [ssd0] MBREnable set off
[Sat Jun 24 17:53:50 2023] -----
.
.
.
[Sat Jun 24 17:54:53 2023] -----
```

```
[Sat Jun 24 17:54:53 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme8n1
[Sat Jun 24 17:54:53 2023] [ssd8] /dev/nvme8n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:54:53 2023]
[Sat Jun 24 17:54:53 2023] [ssd8] factory reset 22443E9DC54E
[Sat Jun 24 17:54:53 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --PSIDrevert 8287AD621B44
[Sat Jun 24 17:54:56 2023] [ssd8] revertTper completed successfully
[Sat Jun 24 17:54:56 2023]
[Sat Jun 24 17:54:56 2023] [ssd8] set default password
[Sat Jun 24 17:54:56 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --initialSetup ***** /dev
[Sat Jun 24 17:54:57 2023] [ssd8] SID password changed
[Sat Jun 24 17:54:57 2023] [ssd8] takeOwnership complete
[Sat Jun 24 17:54:58 2023] [ssd8] Locking SP Activate Complete
[Sat Jun 24 17:54:59 2023] [ssd8] LockingRange0 disabled
[Sat Jun 24 17:54:59 2023] [ssd8] LockingRange0 set to RW
[Sat Jun 24 17:55:00 2023] [ssd8] MBRDone set on
[Sat Jun 24 17:55:01 2023] [ssd8] MBREnable set on
[Sat Jun 24 17:55:01 2023] [ssd8] Initial setup of TPer complete on /dev/nvme8n1
[Sat Jun 24 17:55:01 2023]
[Sat Jun 24 17:55:01 2023] [ssd8] disable MBR
[Sat Jun 24 17:55:01 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --setMBREnable off *****
[Sat Jun 24 17:55:01 2023] [ssd8] MBRDone set on
[Sat Jun 24 17:55:02 2023] [ssd8] MBREnable set off
[Sat Jun 24 17:55:02 2023] >
```

## config disk password

Changes the password used for all encryption related disk operations.

Enter Old Password may be ENTER/NULL in which case the default password will be used

Enter Old Password can be read without keyboard input from the file /tmp/disk-password-old if the file exists

Enter New Password can be read without keyboard input from the file /tmp/disk-password if the file exists

```
config disk password
```

Example of setting a new password from the default password



```
[Sat Jun 24 17:58:51 2023] > config disk password
[Sat Jun 24 17:58:53 2023] Enter Old Password:

[Sat Jun 24 17:58:56 2023] Enter New Password:
****
[Sat Jun 24 17:58:59 2023] Re-Enter New Password:
****
[Sat Jun 24 17:59:28 2023] -----
[Sat Jun 24 17:59:28 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme9n1
[Sat Jun 24 17:59:28 2023] [par0] /dev/nvme9n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:28 2023]
[Sat Jun 24 17:59:28 2023] [par0] set new password 22443E9D2087
[Sat Jun 24 17:59:28 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:29 2023] [par0] Admin1 password changed
[Sat Jun 24 17:59:29 2023] -----
[Sat Jun 24 17:59:29 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme10n
[Sat Jun 24 17:59:29 2023] [ssd0] /dev/nvme10n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:29 2023]
[Sat Jun 24 17:59:29 2023] [ssd0] set new password 22443E9D204F
[Sat Jun 24 17:59:29 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:30 2023] [ssd0] Admin1 password changed
[Sat Jun 24 17:59:30 2023] -----
[Sat Jun 24 17:59:30 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme1n1
[Sat Jun 24 17:59:30 2023] [ssd1] /dev/nvme1n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:30 2023]
[Sat Jun 24 17:59:30 2023] [ssd1] set new password 22223AD5BFC3
[Sat Jun 24 17:59:30 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:32 2023] [ssd1] Admin1 password changed
[Sat Jun 24 17:59:32 2023] -----
[Sat Jun 24 17:59:32 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme6n1
[Sat Jun 24 17:59:32 2023] [ssd2] /dev/nvme6n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:32 2023]
[Sat Jun 24 17:59:32 2023] [ssd2] set new password 22443E9D3AFF
[Sat Jun 24 17:59:32 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:33 2023] [ssd2] Admin1 password changed
[Sat Jun 24 17:59:33 2023] -----
[Sat Jun 24 17:59:33 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme5n1
[Sat Jun 24 17:59:33 2023] [ssd3] /dev/nvme5n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:33 2023]
[Sat Jun 24 17:59:33 2023] [ssd3] set new password 22443E9DC543
[Sat Jun 24 17:59:33 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:34 2023] [ssd3] Admin1 password changed
[Sat Jun 24 17:59:34 2023] -----
[Sat Jun 24 17:59:34 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme7n1
[Sat Jun 24 17:59:34 2023] [ssd4] /dev/nvme7n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:34 2023]
[Sat Jun 24 17:59:34 2023] [ssd4] set new password 22443E9D2076
[Sat Jun 24 17:59:34 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:36 2023] [ssd4] Admin1 password changed
[Sat Jun 24 17:59:36 2023] -----
[Sat Jun 24 17:59:36 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme2n1
[Sat Jun 24 17:59:36 2023] [ssd5] /dev/nvme2n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
```

```

[Sat Jun 24 17:59:36 2023]
[Sat Jun 24 17:59:36 2023] [ssd5] set new password 22443E9D3B41
[Sat Jun 24 17:59:36 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:37 2023] [ssd5] Admin1 password changed
[Sat Jun 24 17:59:37 2023] -----
[Sat Jun 24 17:59:37 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme3n1
[Sat Jun 24 17:59:37 2023] [ssd6] /dev/nvme3n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:37 2023]
[Sat Jun 24 17:59:37 2023] [ssd6] set new password 22443E9D3B65
[Sat Jun 24 17:59:37 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:39 2023] [ssd6] Admin1 password changed
[Sat Jun 24 17:59:39 2023] -----
[Sat Jun 24 17:59:39 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme4n1
[Sat Jun 24 17:59:39 2023] [ssd7] /dev/nvme4n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:39 2023]
[Sat Jun 24 17:59:39 2023] [ssd7] set new password 22443E9D20A4
[Sat Jun 24 17:59:39 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:40 2023] [ssd7] Admin1 password changed
[Sat Jun 24 17:59:40 2023] -----
[Sat Jun 24 17:59:40 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme8n1
[Sat Jun 24 17:59:40 2023] [ssd8] /dev/nvme8n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 17:59:40 2023]
[Sat Jun 24 17:59:40 2023] [ssd8] set new password 22443E9DC54E
[Sat Jun 24 17:59:40 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --setPassword ***** Admin
[Sat Jun 24 17:59:41 2023] [ssd8] Admin1 password changed
[Sat Jun 24 17:59:41 2023] >

```

## config disk lock

This sets the disks into the locked state requiring a password.

On any power cycle command (disks loose power) the disks will need to be unlocked using the `config disk unlock` command and a password

The password can be read without keyboard input from the file `/tmp/disk-password` if the file exists

```
config disk lock
```

Example locks all data disks

```
[Sat Jun 24 18:03:58 2023] > config disk lock
[Sat Jun 24 18:03:58 2023] Enter Password. or CR for default:
****
[Sat Jun 24 18:04:00 2023] -----
[Sat Jun 24 18:04:00 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme9n1
[Sat Jun 24 18:04:00 2023] [par0] /dev/nvme9n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:00 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:01 2023] [par0] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:01 2023]
[Sat Jun 24 18:04:01 2023] [par0] set to locked22443E9D2087
[Sat Jun 24 18:04:01 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:02 2023] [par0] LockingRange0 set to LK
[Sat Jun 24 18:04:02 2023] -----
[Sat Jun 24 18:04:02 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme10n
[Sat Jun 24 18:04:02 2023] [ssd0] /dev/nvme10n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:02 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:03 2023] [ssd0] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:03 2023]
[Sat Jun 24 18:04:03 2023] [ssd0] set to locked22443E9D204F
[Sat Jun 24 18:04:03 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:04 2023] [ssd0] LockingRange0 set to LK
[Sat Jun 24 18:04:04 2023] -----
[Sat Jun 24 18:04:04 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme1n1
[Sat Jun 24 18:04:04 2023] [ssd1] /dev/nvme1n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:04 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:05 2023] [ssd1] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:05 2023]
[Sat Jun 24 18:04:05 2023] [ssd1] set to locked22223AD5BFC3
[Sat Jun 24 18:04:05 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:06 2023] [ssd1] LockingRange0 set to LK
[Sat Jun 24 18:04:06 2023] -----
[Sat Jun 24 18:04:06 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme6n1
[Sat Jun 24 18:04:06 2023] [ssd2] /dev/nvme6n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:06 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:07 2023] [ssd2] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:07 2023]
[Sat Jun 24 18:04:07 2023] [ssd2] set to locked22443E9D3AFF
[Sat Jun 24 18:04:07 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:07 2023] [ssd2] LockingRange0 set to LK
[Sat Jun 24 18:04:07 2023] -----
[Sat Jun 24 18:04:07 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme5n1
[Sat Jun 24 18:04:07 2023] [ssd3] /dev/nvme5n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:07 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:08 2023] [ssd3] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:08 2023]
[Sat Jun 24 18:04:08 2023] [ssd3] set to locked22443E9DC543
[Sat Jun 24 18:04:08 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:09 2023] [ssd3] LockingRange0 set to LK
[Sat Jun 24 18:04:09 2023] -----
[Sat Jun 24 18:04:09 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme7n1
[Sat Jun 24 18:04:09 2023] [ssd4] /dev/nvme7n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:09 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
```

```

[Sat Jun 24 18:04:10 2023] [ssd4] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:10 2023]
[Sat Jun 24 18:04:10 2023] [ssd4] set to locked22443E9D2076
[Sat Jun 24 18:04:10 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:10 2023] [ssd4] LockingRange0 set to LK
[Sat Jun 24 18:04:10 2023] -----
[Sat Jun 24 18:04:10 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme2n1
[Sat Jun 24 18:04:10 2023] [ssd5] /dev/nvme2n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:10 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:11 2023] [ssd5] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:11 2023]
[Sat Jun 24 18:04:11 2023] [ssd5] set to locked22443E9D3B41
[Sat Jun 24 18:04:11 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:12 2023] [ssd5] LockingRange0 set to LK
[Sat Jun 24 18:04:12 2023] -----
[Sat Jun 24 18:04:12 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme3n1
[Sat Jun 24 18:04:12 2023] [ssd6] /dev/nvme3n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:12 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:13 2023] [ssd6] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:13 2023]
[Sat Jun 24 18:04:13 2023] [ssd6] set to locked22443E9D3B65
[Sat Jun 24 18:04:13 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:13 2023] [ssd6] LockingRange0 set to LK
[Sat Jun 24 18:04:13 2023] -----
[Sat Jun 24 18:04:13 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme4n1
[Sat Jun 24 18:04:13 2023] [ssd7] /dev/nvme4n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:13 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:14 2023] [ssd7] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:14 2023]
[Sat Jun 24 18:04:14 2023] [ssd7] set to locked22443E9D20A4
[Sat Jun 24 18:04:14 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:15 2023] [ssd7] LockingRange0 set to LK
[Sat Jun 24 18:04:15 2023] -----
[Sat Jun 24 18:04:15 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme8n1
[Sat Jun 24 18:04:15 2023] [ssd8] /dev/nvme8n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:04:15 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --enableLockingRange 0 **
[Sat Jun 24 18:04:16 2023] [ssd8] LockingRange0 enabled ReadLocking,WriteLocking
[Sat Jun 24 18:04:16 2023]
[Sat Jun 24 18:04:16 2023] [ssd8] set to locked22443E9DC54E
[Sat Jun 24 18:04:16 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 LK **
[Sat Jun 24 18:04:16 2023] [ssd8] LockingRange0 set to LK

```

## config disk no-lock

This removes the disk locking function of the drives.

```
config disk no-lock
```

Example below shows a disk no-lock operation

```
[Fri Jun 30 08:59:34 2023] > config disk no-lock
[Fri Jun 30 08:59:34 2023] Enter Password. or CR for default:
*****
[Fri Jun 30 08:59:37 2023] -----
[Fri Jun 30 08:59:37 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme8n1
[Fri Jun 30 08:59:37 2023] [par0] /dev/nvme8n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:37 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:38 2023] [par0] LockingRange0 disabled
[Fri Jun 30 08:59:38 2023] -----
[Fri Jun 30 08:59:38 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme2n1
[Fri Jun 30 08:59:38 2023] [ssd0] /dev/nvme2n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:38 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:39 2023] [ssd0] LockingRange0 disabled
[Fri Jun 30 08:59:39 2023] -----
[Fri Jun 30 08:59:39 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme9n1
[Fri Jun 30 08:59:39 2023] [ssd1] /dev/nvme9n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:39 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:40 2023] [ssd1] LockingRange0 disabled
[Fri Jun 30 08:59:40 2023] -----
[Fri Jun 30 08:59:40 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme7n1
[Fri Jun 30 08:59:40 2023] [ssd2] /dev/nvme7n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:40 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:42 2023] [ssd2] LockingRange0 disabled
[Fri Jun 30 08:59:42 2023] -----
[Fri Jun 30 08:59:42 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme6n1
[Fri Jun 30 08:59:42 2023] [ssd3] /dev/nvme6n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:42 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:43 2023] [ssd3] LockingRange0 disabled
[Fri Jun 30 08:59:43 2023] -----
[Fri Jun 30 08:59:43 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme4n1
[Fri Jun 30 08:59:43 2023] [ssd4] /dev/nvme4n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:43 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:44 2023] [ssd4] LockingRange0 disabled
[Fri Jun 30 08:59:44 2023] -----
[Fri Jun 30 08:59:44 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme5n1
[Fri Jun 30 08:59:44 2023] [ssd5] /dev/nvme5n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:44 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:45 2023] [ssd5] LockingRange0 disabled
[Fri Jun 30 08:59:45 2023] -----
[Fri Jun 30 08:59:45 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme3n1
[Fri Jun 30 08:59:45 2023] [ssd6] /dev/nvme3n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:45 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:46 2023] [ssd6] LockingRange0 disabled
[Fri Jun 30 08:59:46 2023] -----
[Fri Jun 30 08:59:46 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme1n1
[Fri Jun 30 08:59:46 2023] [ssd7] /dev/nvme1n1 SED -2----- SAMSUNG MZQL23T8HCLS-00A07
[Fri Jun 30 08:59:46 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --disableLockingRange 0 *
[Fri Jun 30 08:59:48 2023] [ssd7] LockingRange0 disabled
[Fri Jun 30 08:59:48 2023] >
```

## **config disk unlock**

Unlocks the drives using the specified password

The password can be read without keyboard input from the file `/tmp/disk-password` if the file exists

```
config disk unlock
```

Example of unlocking



```
[Sat Jun 24 18:07:22 2023] > config disk unlock
[Sat Jun 24 18:07:23 2023] Enter Password:
****
[Sat Jun 24 18:07:24 2023] -----
[Sat Jun 24 18:07:24 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme9n1
[Sat Jun 24 18:07:24 2023] [par0] /dev/nvme9n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:24 2023]
[Sat Jun 24 18:07:24 2023] [par0] set to locked 22443E9D2087
[Sat Jun 24 18:07:24 2023] [par0] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:25 2023] [par0] LockingRange0 set to RW
[Sat Jun 24 18:07:25 2023] -----
[Sat Jun 24 18:07:25 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme10n
[Sat Jun 24 18:07:25 2023] [ssd0] /dev/nvme10n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:25 2023]
[Sat Jun 24 18:07:25 2023] [ssd0] set to locked 22443E9D204F
[Sat Jun 24 18:07:25 2023] [ssd0] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:25 2023] [ssd0] LockingRange0 set to RW
[Sat Jun 24 18:07:25 2023] -----
[Sat Jun 24 18:07:25 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme1n1
[Sat Jun 24 18:07:25 2023] [ssd1] /dev/nvme1n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:25 2023]
[Sat Jun 24 18:07:25 2023] [ssd1] set to locked 22223AD5BFC3
[Sat Jun 24 18:07:25 2023] [ssd1] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:26 2023] [ssd1] LockingRange0 set to RW
[Sat Jun 24 18:07:26 2023] -----
[Sat Jun 24 18:07:26 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme6n1
[Sat Jun 24 18:07:26 2023] [ssd2] /dev/nvme6n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:26 2023]
[Sat Jun 24 18:07:26 2023] [ssd2] set to locked 22443E9D3AFF
[Sat Jun 24 18:07:26 2023] [ssd2] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:27 2023] [ssd2] LockingRange0 set to RW
[Sat Jun 24 18:07:27 2023] -----
[Sat Jun 24 18:07:27 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme5n1
[Sat Jun 24 18:07:27 2023] [ssd3] /dev/nvme5n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:27 2023]
[Sat Jun 24 18:07:27 2023] [ssd3] set to locked 22443E9DC543
[Sat Jun 24 18:07:27 2023] [ssd3] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:27 2023] [ssd3] LockingRange0 set to RW
[Sat Jun 24 18:07:27 2023] -----
[Sat Jun 24 18:07:27 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme7n1
[Sat Jun 24 18:07:27 2023] [ssd4] /dev/nvme7n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:27 2023]
[Sat Jun 24 18:07:27 2023] [ssd4] set to locked 22443E9D2076
[Sat Jun 24 18:07:27 2023] [ssd4] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:28 2023] [ssd4] LockingRange0 set to RW
[Sat Jun 24 18:07:28 2023] -----
[Sat Jun 24 18:07:28 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme2n1
[Sat Jun 24 18:07:28 2023] [ssd5] /dev/nvme2n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:28 2023]
[Sat Jun 24 18:07:28 2023] [ssd5] set to locked 22443E9D3B41
[Sat Jun 24 18:07:28 2023] [ssd5] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:29 2023] [ssd5] LockingRange0 set to RW
```

```

[Sat Jun 24 18:07:29 2023] -----
[Sat Jun 24 18:07:29 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme3n1
[Sat Jun 24 18:07:29 2023] [ssd6] /dev/nvme3n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:29 2023]
[Sat Jun 24 18:07:29 2023] [ssd6] set to locked 22443E9D3B65
[Sat Jun 24 18:07:29 2023] [ssd6] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:29 2023] [ssd6] LockingRange0 set to RW
[Sat Jun 24 18:07:29 2023] -----
[Sat Jun 24 18:07:29 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme4n1
[Sat Jun 24 18:07:29 2023] [ssd7] /dev/nvme4n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:29 2023]
[Sat Jun 24 18:07:29 2023] [ssd7] set to locked 22443E9D20A4
[Sat Jun 24 18:07:29 2023] [ssd7] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:30 2023] [ssd7] LockingRange0 set to RW
[Sat Jun 24 18:07:30 2023] -----
[Sat Jun 24 18:07:30 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --isValidSED /dev/nvme8n1
[Sat Jun 24 18:07:30 2023] [ssd8] /dev/nvme8n1 SED -2----- Micron_7450_MTFDKCC3T8TFR
[Sat Jun 24 18:07:30 2023]
[Sat Jun 24 18:07:30 2023] [ssd8] set to locked 22443E9DC54E
[Sat Jun 24 18:07:30 2023] [ssd8] sudo /usr/bin/sedutil-cli-sha512 --setLockingRange 0 RW **
[Sat Jun 24 18:07:31 2023] [ssd8] LockingRange0 set to RW
[Sat Jun 24 18:07:31 2023] >

```

## System Configuration

### config system fpga firmware

This configures the systems FPGA firmware, currently support modes

- capture-2x100G (2 x100G packet capture mode)
- capture-2x40G (2x40G packet capture mode)
- capture-8x10G (8x10G packet capture mode)

NOTE: this only sets up the system. **Reboot is required to start the configuration change**

Example:

```
config system fpga firmware capture-2x100G
```

Output



```
[Fri Aug 25 15:52:35 2023] > config system fpga firmware capture-2x100G
[Fri Aug 25 15:52:35 2023]     fmad fmadlua Aug 24 2023 (/usr/local/bin/fmadiolua /opt/fmadi
[Fri Aug 25 15:52:35 2023]     calibrating...
[Fri Aug 25 15:52:36 2023]     0 : 2095071170                2.0951 cycles/nsec offset:4.929 Mhz
[Fri Aug 25 15:52:36 2023]     Cycles/Sec 2095071170.0000 Std:          0 cycle std( 0.0000000
[Fri Aug 25 15:52:36 2023]     Change PortMode
[Fri Aug 25 15:52:36 2023]     Copy [cp /mnt/system/boot/bitstream.rom.2x100G /mnt/system/bc
[Fri Aug 25 15:52:36 2023]     os[sudo /opt/fmadio/bin/bitstream_update.lua --noreboot --wri
[Fri Aug 25 15:52:36 2023]     fmad fmadlua Aug 24 2023 (/opt/fmadio/bin/fmadiolua /opt/fmad
[Fri Aug 25 15:52:36 2023]     calibrating...
[Fri Aug 25 15:52:37 2023]     0 : 2095073220                2.0951 cycles/nsec offset:4.927 Mhz
[Fri Aug 25 15:52:37 2023]     Cycles/Sec 2095073220.0000 Std:          0 cycle std( 0.0000000
[Fri Aug 25 15:52:37 2023]     cp: '/mnt/system/boot/bitstream.rom' and '/mnt/system//boot/b
[Fri Aug 25 15:52:37 2023]     done 0.006711Sec 0.000112Min
[Fri Aug 25 15:52:37 2023]     Firmware Install Complete
[Fri Aug 25 15:52:37 2023]     done 1.083203Sec 0.018053Min
[Fri Aug 25 15:52:37 2023] set port mode from(2x40G) -> (2x100G)
[Fri Aug 25 15:52:37 2023]
[Fri Aug 25 15:52:37 2023] *****
[Fri Aug 25 15:52:37 2023] ***** REBOOT IS REQUIRED. *****
[Fri Aug 25 15:52:37 2023] *****
[Fri Aug 25 15:52:37 2023]
[Fri Aug 25 15:52:37 2023] Reboot -> Safe Mode -> FPGA Flash -> Reboot -> System ready (5-10
[Fri Aug 25 15:52:37 2023]
[Fri Aug 25 15:52:37 2023] >
```

# stream\_cat

stream\_cat is the core utility to extract data off the system. By default it outputs a standard nanosecond PCAP to stdout. This can be piped in multiple ways per the unix philosophy

```
$ sudo stream_cat --help
stream_cat -vf (stream name)

-v                : verbose print of status
--follow         : follow mode
--follow-start   : follow mode but start at the beginning of the file
--check-fcs     : check for FCS errors
--force-hdd     : force reading from HDD
--info          : dump info on the current capture
--cpu <cpu id>  : pin to a specific CPU
--io-priority <level> : sets IO priority (default 20)
-n N            : follow mode start N packets from the end
--bpf "bpf expression" : add a BPF filter expression
--chunked       : output chunks of packets
--decap        : de-encapsulate packets
--pktslice <bytes> : slice packets before sending down the pipe
--time-start <HH:MM:SS> : time start in local timezone hour:min:sec
--time-stop <HH:MM:SS> : time stop in local timezone hour:min:sec

--epoch-start <epoch time> : time start in epoch nano seconds
--epoch-stop <epoch time> : time stop in epoch nano seconds
--no-pcap-header           : remove pcap header from output stream

Example: 'sudo stream_cat -f -n 1000000 | tcpdump -r - -nn '
Example: 'sudo stream_cat --bpf "host 192.168.1.1" | tcpdump -r - -nn '

$
```

## Reference

CLI argument reference

### **--cpu <cpu number>**

Pins stream\_cat to a specific CPU number.

### **--ring <ring path> <bpf filter> <cpu number>**

Writes PCAP to the specified LXC <ring path> when the <bpf filter> matches.

Multiple rings can be specified

NOTE: if no BPF is used <bpf filter> needs to be ""

NOTE: If no CPU is specified, use setting of "0"

All fields must be populated.

Example:

```
sudo stream_cat --ring /opt/fmadio/queue/lxc_ring0 "net 192.168.0.0/24" 0
                --ring /opt/fmadio/queue/lxc_ring1 "net 192.168.1.0/24" 29
                --ring /opt/fmadio/queue/lxc_ring2 "net 192.168.2.0/24" 30
                --ring /opt/fmadio/queue/lxc_ring3 "net 192.168.3.0/24" 31
my_capture_20220325_000
```

### **--ring-reset**

Reset the LXC Ring read/write pointers

### **--ring-depth**

Adjusted the size of the ring FIFO depth. This value must be a Power of 2. Maximum value is 1024

Default value 8

### **--ring-timeout <timeout in nanoseconds>**

Default: 30e9 (30 seconds)

Sets the default LXC Ring timeout value in nanoseconds.

### **--delta-histo**

Used in combination with capinfos2 Generates a histogram of the time between packets displaying it in a vertical histogram form.

Default bin size is 1nsec

Default offset is 0nsec

Example output is shown below:

```
sudo stream_cat --bpf "vlan and dst 10.1.2.3"
                --epoch-start 1651131955760021825
                --epoch-stop 1651134322452715943
test_capture_20220201
| capinfos2 -v
--delta-histo
--delta-histo-offset 10e6
--delta-histo-bin 1e6
```

```

fmadio@fmadio100v2-228u:/mnt/store0/tmp2$ sudo stream_cat --bpf "vlan and dst 10.1.2.3"
--epoch-start 1651131955760021825
--epoch-stop 1651134322452715943
test_capture_20220201
| capinfos2 -v --delta-histo
--delta-histo-offset 10e6
--delta-histo-bin 1e6
0.03GB      0.185 Gbps      0.102 Mpps
Total Packets: 114698
Delta Time Histogram
TShisto    0 ns :      1744 (0.015) 0.000 : *****
TShisto   1000000 ns :         7 (0.000) 0.015 : *
TShisto   3000000 ns :         2 (0.000) 0.015 : *
TShisto   4000000 ns :        805 (0.007) 0.015 : ****
TShisto   5000000 ns :       1663 (0.014) 0.022 : *****
TShisto   6000000 ns :         11 (0.000) 0.037 : *
TShisto   7000000 ns :         2 (0.000) 0.037 : *
TShisto   8000000 ns :         29 (0.000) 0.037 : *
TShisto   9000000 ns :       3274 (0.029) 0.037 : *****
TShisto  10000000 ns :      20403 (0.178) 0.066 : *****
TShisto  11000000 ns :        307 (0.003) 0.244 : *
TShisto  12000000 ns :         7 (0.000) 0.246 : *
TShisto  13000000 ns :        175 (0.002) 0.246 : *
TShisto  14000000 ns :     11768 (0.103) 0.248 : *****
TShisto  15000000 ns :    12553 (0.109) 0.350 : *****
TShisto  16000000 ns :        118 (0.001) 0.460 : *
TShisto  17000000 ns :         4 (0.000) 0.461 : *
TShisto  18000000 ns :        115 (0.001) 0.461 : *
TShisto  19000000 ns :       7503 (0.065) 0.462 : *****
TShisto  20000000 ns :     22200 (0.194) 0.527 : *****
TShisto  21000000 ns :        266 (0.002) 0.721 : *
TShisto  22000000 ns :         8 (0.000) 0.723 : *
TShisto  23000000 ns :        150 (0.001) 0.723 : *
TShisto  24000000 ns :     8978 (0.078) 0.725 : *****
TShisto  25000000 ns :    13168 (0.115) 0.803 : *****
TShisto  26000000 ns :        143 (0.001) 0.918 : *
TShisto  27000000 ns :         4 (0.000) 0.919 : *
TShisto  28000000 ns :         46 (0.000) 0.919 : *****
TShisto  29000000 ns :       3015 (0.026) 0.919 : *****
TShisto  30000000 ns :       3868 (0.034) 0.946 : *****
TShisto  31000000 ns :         24 (0.000) 0.979 : *
TShisto  33000000 ns :         4 (0.000) 0.980 : *
TShisto  34000000 ns :        411 (0.004) 0.980 : *
TShisto  35000000 ns :        858 (0.007) 0.983 : ****
TShisto  36000000 ns :         2 (0.000) 0.991 : *
TShisto  38000000 ns :         1 (0.000) 0.991 : *
TShisto  39000000 ns :        128 (0.001) 0.991 : ****
TShisto  40000000 ns :        604 (0.005) 0.992 : ****
TShisto  41000000 ns :         2 (0.000) 0.997 : *
TShisto  44000000 ns :        16 (0.000) 0.997 : *
TShisto  45000000 ns :        80 (0.001) 0.997 : *
TShisto  49000000 ns :         4 (0.000) 0.998 : *
TShisto  50000000 ns :        53 (0.000) 0.998 : *
TShisto  54000000 ns :         1 (0.000) 0.998 : *
TShisto  55000000 ns :        20 (0.000) 0.998 : *
TShisto  59000000 ns :         2 (0.000) 0.999 : *
TShisto  60000000 ns :        14 (0.000) 0.999 : *
TShisto  65000000 ns :         6 (0.000) 0.999 : *
TShisto  69000000 ns :         1 (0.000) 0.999 : *
TShisto  70000000 ns :         6 (0.000) 0.999 : *
TShisto  75000000 ns :         6 (0.000) 0.999 : *
TShisto  79000000 ns :         2 (0.000) 0.999 : *
TShisto  80000000 ns :         8 (0.000) 0.999 : *
TShisto  85000000 ns :         6 (0.000) 0.999 : *
TShisto  90000000 ns :         9 (0.000) 0.999 : *
TShisto  91000000 ns :         1 (0.000) 0.999 : *
TShisto  95000000 ns :         4 (0.000) 0.999 : *
TShisto 100000000 ns :        11 (0.000) 0.999 : *
TShisto 105000000 ns :        17 (0.000) 0.999 : *
TShisto 110000000 ns :        23 (0.000) 0.999 : *
TShisto 111000000 ns :         1 (0.000) 1.000 : *
TShisto 115000000 ns :        13 (0.000) 1.000 : *
TShisto 116000000 ns :         1 (0.000) 1.000 : *
TShisto 119000000 ns :         1 (0.000) 1.000 : *
TShisto 120000000 ns :        11 (0.000) 1.000 : *
TShisto 125000000 ns :         3 (0.000) 1.000 : *
TShisto 130000000 ns :         2 (0.000) 1.000 : *
TShisto -484934592 ns :         1 (0.000) 1.000 : *
TShisto 1028065408 ns :         1 (0.000) 1.000 : *
TShisto 1409065408 ns :         4 (0.000) 1.000 : *
DeltaMean : 18.409467 ns
DeltaStdDev : 70.044783 ns
TotalBytes : 24315976
TotalPackets : 114698
PayloadCRC : 294607d5a42766
ErrorSeq : 0
ErrorPktsSize : 0
LastByte : 0x00000000
SeqStart : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
SeqEnd : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
PacketCnt : 0 0 0 0
TimeOrder : 0
CRCFail : 0
TotalPCAPTime : 2366668062775 ns
Bandwidth : 0.000 Gbps
Packet Rate : 0.000 Mpps
Complete
fmadio@fmadio100v2-228u:/mnt/store0/tmp2$

```

stream\_cat with delta histogram

Above example uses stream\_cat with an epoch and BPF filter to isolate the packet histogram deltas between packets. This is particularly useful for checking QoS SLAs

### --delta-histo-bin <nanos>

Used with capinfos2 it specifies the width of each timebin (e.g. the histogram resolution). By default it uses 1nsec. Example usage below, this uses a 1e6 (1 millisecond) time bin with a 10msec offset.

```
sudo stream_cat --bpf "vlan and dst 10.1.2.3"
                --epoch-start 1651131955760021825
                --epoch-stop 1651134322452715943
test_capture_20220201
| capinfos2 -v
--delta-histo
--delta-histo-offset 10e6
--delta-histo-bin 1e6
```

### **--delta-histo-offset <nanos>**

As the number of timebins is limited, it may be necessary to offset the histogram to where the data is. The example below offsets it by 10msec with a time bin of 1msec.

```
sudo stream_cat --bpf "vlan and dst 10.1.2.3"
                --epoch-start 1651131955760021825
                --epoch-stop 1651134322452715943
test_capture_20220201
| capinfos2 -v
--delta-histo
--delta-histo-offset 10e6
--delta-histo-bin 1e6
```

### **--epoch-start <nanosecond epoch>**

Filters the specified capture using start time specified argument epoch time value.

Value of 0 means filter is disabled

NOTE: typically --epoch-start and --epoch-stop are used together

Example: filter from epoch 1497015595000000000. This uses capinfos2 to verify the first packet (Time First) is as specified in the filter

```

fmadio@fmadio100v2-228U:~$ sudo stream_cat --epoch-start 1497015595000000000 interop17_20
Epoch Start 13:39:55.000.000.000 1497015595000000000
Epoch found start Chunks:26491 Bytes 6.944 GB skipped
StartChunkID: 3737931
StartChunk: 3737931 Offset: 0 Stride: 1
StartChunk: 3737931
PCAP nano
0.00GB    0.000 Gbps    0.000 Mpps
0.40GB    3.191 Gbps    2.063 Mpps
.
.

9.22GB    3.202 Gbps    2.067 Mpps
9.60GB    3.074 Gbps    1.992 Mpps
packet stream end
20220511_131303 25.643s : SUCCESS
Total Packets: 50746286
TotalBytes      : 57674285331
TotalPackets    : 50746286
PayloadCRC      : 33ad29d7358038fd
ErrorSeq        : 0
ErrorPktSize    : 0
LastByte        : 0x00000000
SeqStart        : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
SeqEnd          : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
PacketCnt       : 0 0 0 0
TimeOrder       : 0
CRCFail         : 0
Time First      : 20170609_133955 13:39:55.000.000.000 (1497015595.000000000)
Time Last       : 20170609_133956 13:39:56.807.825.664 (1497015596.807825664)
TotalPCAPTime   : 1807825664 ns
Bandwidth       : 255.221 Gbps
Packet Rate     : 28.070 Mpps

Complete
fmadio@fmadio100v2-228U:~$

```

### **--epoch-stop <nanosecond epoch>**

Filters the specified capture using and end time specified argument epoch time value.

Value of 0 means filter is disabled

NOTE: typically --epoch-start and --epoch-stop are used together

Example: filter up to epoch time 1497015594000000000. This example uses capinfos2 to verify the last packet (Time Last) meets the specified filter value.

```

fmadio@fmadio100v2-228U:~$ sudo stream_cat --epoch-stop 1497015594000000000 interop17_202
Epoch Stop 13:39:54.000.000.000 1497015594000000000
StartChunkID: 3711440
StartChunk: 3711440 Offset: 0 Stride: 1
StartChunk: 3711440
PCAP nano
0.00GB    0.000 Gbps    0.000 Mpps
0.48GB    3.804 Gbps    2.460 Mpps
0.89GB    3.269 Gbps    2.106 Mpps
1.29GB    3.270 Gbps    2.113 Mpps
TimeStop reached
20220511_131655 4.686s : SUCCESS
Total Packets: 7839777
TotalBytes      : 9003464361
TotalPackets    : 7839777
PayloadCRC      : 81ff1b0c04cdca1
ErrorSeq        : 0
ErrorPktSize    : 0
LastByte        : 0x00000000
SeqStart        : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
SeqEnd          : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
PacketCnt       : 0 0 0 0
TimeOrder       : 0
CRCFail         : 0
Time First      : 20170609_133953 13:39:53.717.953.280 (1497015593.717953280)
Time Last       : 20170609_133953 13:39:53.999.999.744 (1497015593.999999744)
TotalPCAPTime   : 282046464 ns
Bandwidth       : 255.375 Gbps
Packet Rate     : 27.796 Mpps

Complete
fmadio@fmadio100v2-228U:~$

```

## Example Usage

The following section shows how to use `stream_cat` on the command line in various different ways.

Where **test\_capture** is used, replace with a stream capture name from your `fmadio` system.

Where **sample\_file.pcap** is used, replace with your own pcap filename.

### Whole file

To create a whole pcap of an entire `fmadio` system capture use the following:

```
stream_cat -v test_capture > sample_file.pcap
```

## Time Selection

To choose a selection of time for a pcap on the fmadio system the following can be used. The following example selects a time period using epoch nano seconds. 1000 nanoseconds of capture time will be extracted - assuming the stream was captured during this epoch period.

```
stream_cat -v --epoch-start 1622000000000000000 --epoch-stop 1622000000000001000 test_captur
```

## Packet Filters

Stream\_cat can be executed with packet filtering commands. These are similar to the filter methods used by [wireshark filtering](#). Example filters are also available in the [fmadio user guide](#).

The examples here show some simple filter examples.

Stream\_cat with a IP and UDP filter:

```
stream_cat -v --bpf "ip and udp" test_capture > sample_file.pcap
```

Stream\_cat with a UDP port 80 filter:

```
stream_cat -v --bpf "udp port 80" test_capture > sample_file.pcap
```

Stream\_cat with a complex filter - select port 80 packets with tcp range selectors :

```
stream_cat -v --bpf "port 80 and tcp[((tcp[12:1] & 0xf0) >> 2):4]" test_capture > sample_fil
```

## Piping

Stream\_cat is very useful for piping output to other programs to process the data. Examples are shown in the **stream\_cat --help**. The example here shows stream\_cat used with gzip to compress the output pcap into a smaller sized file.

```
stream_cat -v -bpf "udp port 80" test_capture | gzip --fast > sample_file.pcap.gz
```

## LXC Ring

*streamcat can write directly to an lxc ring buffer located in /opt/fmadio/queue/lxc\_ring0*

It can also write to multiple lxcring buffers from a single stream\_cat instance by issuing the --ring command line multiple times.

### LXC Ring - One Ring - No Filter



This example writes all captured data to a single LXC Ring with no BPF filter applied

```
$ sudo stream_cat --ring /opt/fmadio/queue/lxc_ring0 "" my_captuyre_20220325_000
```

NOTE: the "" arguments are required. This indicates a NULL BPF filter

### LXC Ring - One Ring - BPF Filter

The following example writes to a single LXC Ring with a simple "tcp" BPF filter

```
$ sudo stream_cat --ring /opt/fmadio/queue/lxc_ring0 "tcp" my_captuyre_20220325_000
```

### LXC Ring - 4 Ring - BPF Filter

Example below shows a single stream\_cat instance writing to 4 separate LXC Ring rings each with a different BPF Filter

```
$ sudo stream_cat --ring /opt/fmadio/queue/lxc_ring0 "net 192.168.0.0/24"  
--ring /opt/fmadio/queue/lxc_ring1 "net 192.168.1.0/24"  
--ring /opt/fmadio/queue/lxc_ring2 "net 192.168.2.0/24"  
--ring /opt/fmadio/queue/lxc_ring3 "net 192.168.3.0/24"  
my_captuyre_20220325_000
```

NOTE: above should be a single line

# fnic\_test

fnic\_test is a general purpose utility to configure and run settings specific to FMADIO Packet Capture NIC card

```
$ sudo fnic_test --help
fnic_test <options>
Options:
--trans      : Set Transceiver tuning options
--loopback   : set loopback mode
--dump       : write crash dump
--cap-enable <0 0> : set capture port enable/disables
--fec-force   : force FEC enabled
--no-fec-force : unforce FEC enabled
$
```

API

# Summary

A summary for developers to access a FMADIO device using the API.

## FMADIO API

The FMADIO API is simple and designed for easy scripting integration.

**Note:** Replace the IP 127.0.0.1 with the host IP of your FMADIO device.

## Device Operation

∨ **GET** /sysmaster/capture\_start?StreamName=<capture name>

### Capture Start

This Command starts a capture running on the device.

### Parameters

#### Query

StreamName string

Stream capture name

### Responses

● 200

>

✓ GET /sysmaster/capture\_stop

### Capture Stop

Stops any currently capturing process.  
NOTE: this does NOT stop scheduled captures.

#### Parameters

No parameters

#### Responses

● 200 >

✓ GET /sysmaster/capture\_status

### Capture Status JSON format

Returns status of the capture in JSON format

#### Parameters

No parameters

#### Responses

● 200: OK >

✓ GET /sysmaster/status

### General System Status (CSV)

Returns Capture status of currently active capture.

### Parameters

No parameters

### Responses

● 200

>

## Device Management

GET /sysmaster/stats\_summary

## System Status

Get system status information. Partial example output in parsed JSON format below

```
bytes_disk: 15063434096
bytes_over: average_compression
bytes_pending: 160033931264
cap0_link: 1
cap0_link_uptime: "1D 15H 58M"
cap0_mode: " 10G SR"
cap0_powerrx: " 14.080 mW"
cap0_powertx: " 4.710 mA"
cap0_speed: 10
cap0_temperature: " 42.226 C"
cap0_vendor: " FS"
cap0_vendorprn: " SFP-10GSR-85"
cap0_voltage: " 3.119 V"
cap0_wavelength: " 850nm"
cap1_link: 1
cap1_link_uptime: "1D 15H 58M"
cap1_mode: " 10G SR"
cap1_powerrx: " 11.627 mW"
cap1_powertx: " 4.764 mA"
cap1_speed: 10
cap1_temperature: " 45.944 C"
cap1_vendor: " FS"
cap1_vendorprn: " SFP-10GSR-85"
cap1_voltage: " 3.228 V"
cap1_wavelength: " 850nm"
capture_bytes: 25612800000
capture_days: "249D 22H 28M"
capture_dropped: 0
capture_errors: 0
capture_received: 4002000000
chunk_errors: 0
clock_gm_mac: ""
clock_gm_offset_mean: 1.76
clock_gm_offset_stddev: 25.006847062355
clock_gm_sync: 1
clock_gm_uptime: "1D 15H 57M"
clock_isys_offset_mean: -1.5383714798126
clock_isys_offset_stddev: 305.82805634024
clock_isys_sync: 1
clock_isys_uptime: "1D 15H 57M"
clock_ntp_master: ""
clock_ntp_offset_mean: 0
clock_ntp_offset_stddev: 0
clock_ntp_sync: 0
clock_ntp_uptime: "0D 0H 0M"
clock_pps_offset_mean: 0
clock_pps_offset_stddev: 0
clock_pps_sync: 1
clock_pps_uptime: "1D 15H 57M"
clock_sys_offset_mean: 1
clock_sys_offset_stddev: 34.965697476241
clock_sys_sync: 1
clock_sys_uptime: "1D 15H 57M"
```

### Parameters

No parameters

### Responses

● 200

>

∨ **GET** /pcap/del?StreamName=<full capture name>

### Delete Capture

Deletes capture off the system

### Parameters

#### Path

StreamName string

full name of the capture file to be deleted

### Responses

● 200

>

## V1 API

The FMADIO V1 API uses endpoints with parameters. All V1 versions of the API endpoints shall have the format:

```
/api/v1/<operation>/<task>?<params=x>
```

The V1 API has advantages of the previous API. These are:

- Multiple downloads can occur at the same time on the same device (up to 4 concurrently).
- Improved download performance
- Ability to use compressed and bpf filter parameters on most downloads.



Note: All original API url's shall be available as well as the new V1 endpoints.

## **Downloading PCAP**

GET /api/v1/pcap/single

## Single PCAP Download

Download entire capture as a single file. Piping to a file or any other analysis tools is possible.

### Parameters

#### Query

**Compression** string

Compress the returned stream with gzip.

'fast' Fastest compression but not smallest.

'best' Slowest compression smallest size.

1-9 The range from 'fast' to 'best'

**StreamName\*** string

Stream capture name.

**FilterBPF** string

BPF Filter to be applied to the stream.

**FilterBPFDecap** string

**true** : Run BPF filter on a De-Encapsulated version of the packet

**false** : Run BPF on raw packet

Default: false

#### FilterFrame

Filter on the Packet Frame

a7130.device==<device id>

a7130.srcport==<port id>

c3550.srcport==<portid>

capture.port==<portid>

#### TSMode

Sets the Timestamp of the PCAP

nic - FMADIO FPGA timestamp

arista7130 - Arista 7130 (Metamako)

arista7150\_overwrite - Arista 7150 FCS Overwrite

arista7150\_insert - Arista 7150 Insert 32bit

arista7280\_eth64 - Arista 7280 Ethernet 64bit header

arista7280\_mac48 - Arista 7280 SrcMAC 48bit Overwrite

cisco\_erspan3 - Cisco ERPSANv3

cisco3550 - Cisco 3550 (Exablaze)

### Responses



GET /api/v1/pcap/splittime

## Get Specific PCAP timerange (optional BPF filter)

Gets PCAP from the specified StreamName with Start/Stop EPOCH time with an optional BPF filter

### Parameters

#### Query

##### StreamName\*

Capture name to fetch from

##### Start\*

EPOCH Nanosecond start time

##### Stop\*

EPOCH Nanosecond stop time

##### FilterBPF

Escape Encoded BPF filter

##### FilterFrame

Filter on the Packet Frame

a7130.device==<device id>

a7130.srcport==<port id>

c3550.srcport==<portid>

capture.port==<portid>

##### TSMode

Sets the Timestamp of the PCAP

nic - FMADIO FPGA timestamp

arista7130 - Arista 7130 (Metamako)

arista7150\_overwrite - Arista 7150 FCS Overwrite

arista7150\_insert - Arista 7150 Insert 32bit

arista7280\_eth64 - Arista 7280 Ethernet 64bit header

arista7280\_mac48 - Arista 7280 SrcMAC 48bit Overwrite

cisco\_erspan3 - Cisco ERPSANv3

cisco3550 - Cisco 3550 (Exablaze)

### Responses

● 200: OK

>

✓ GET /api/v1/pcap/timerange

## TimeRange PCAP Download (optional BPF and Frame Filter)

Download a timerange of pcap data that can cross over a multiple pcap files.

The timerange results may be a portion of a single pcap stream, or a portion of multiple streams that share a connected time series.

### Examples

<https://docs.fmad.io/fmadio-documentation/api/usage-guide#timerange-1>

#### TSBegin and TSEnd

```
curl -u fmadio:xxx "http://127.0.0.1/api/v1/pcap/timerange?
TSBegin=1621772572136996000&TSEnd=1621774913584264000"
```

#### TSBegin, TSEnd and FilterBPF

```
curl -u fmadio:xxx "http://127.0.0.1/api/v1/pcap/timerange?
TSBegin=1621772572136996000&TSEnd=1621774913584264000" -G --data-
urlencode "FilterBPF=tcp"
```

#### TSBegin, TSEnd, FilterBPF and Compression

```
curl -u fmadio:xxx "http://127.0.0.1/api/v1/pcap/timerange?
TSBegin=1621772572136996000&TSEnd=1621774913584264000&Compression=fas
t" -G --data-urlencode "FilterBPF=tcp"
```

### Parameters

#### Query

**Compression** string

Compress the returned stream with gzip.

'fast' Fastest compression but not smallest.

'best' Slowest compression smallest size.

1-9 The range from 'fast' to 'best'

**TSMAX** integer

Maximum nanosecond of packets to download.

**TSUnit** string

Time Range mode to use for time selection

nanos : Nanoseconds (default)

msecs : Milliseconds

sec : Seconds

YYYYMMDD\_HHMMSS: this year year month day hour min second time format

---

**TSBegin\*** integer

Start time in nanoseconds epoch.

---

**TSEnd\*** integer

Stop time in nanoseconds epoch.

---

**FilterBPF** string

---

**FilterBPFDecap** string

true : Run BPF filter on a De-Encapsulated version of the packet

false : Run BPF on raw packet

Default: false

---

### FilterFrame

Filtering based on frame parameters

capture.port==0,1

(fetch data for capture ports 0 and 1 only)

capture.port!=0

(fetch data for capture ports NOT equal to 0)

a7130.srcdevice!=0

(fetch data for anything that has a valid Arista 7130 device id)

a7130.srcdevice==0

(fetch anything that does NOT have a valid Arista 7130 device id)

a7130.srcdevice=54931 and a7130.srcport=1

(fetch data only for Arista 7130 device id 54931 and Arista 7130 port id 1)

a7130.srcdevice==54931 and a7130.srcport=1,2,3,4

(fetch data for Arista 7130 device id 54931 and Arista 7130 ports 1, 2, 3, 4)

a7130.srcdevice==54931 and a7130.srcport!=1

(fetch data for Arista 7130 device id 54931 and all ports except port 1)

---

### TSMode

Sets the Timestamp of the PCAP

nic - FMADIO FPGA timestamp

arista7130 - Arista 7130 (Metamako)

arista7150\_overwrite - Arista 7150 FCS Overwrite

arista7150\_insert - Arista 7150 Insert 32bit

arista7280\_eth64 - Arista 7280 Ethernet 64bit header

arista7280\_mac48 - Arista 7280 SrcMAC 48bit Overwrite

cisco\_erspan3 - Cisco ERPSANv3  
cisco3550 - Cisco 3550 (Exablaze)

## Responses

● 200

>

## System

∨ **GET** /api/v1/system/time\_current

### Read the current FPGA System Time

Returns the current time on the fpga in epoch nanoseconds. This time/clock is used directly to timestamp packets on the FPGA.

#### Parameters

No parameters

#### Responses

● 200: OK

>

∨ **GET** /api/v1/system/version

### Get current system version information

Returns the current system version.

Example output:

```
fmadio@fmadio100v2-228U:$ curl http://127.0.0.1/api/v1/system/version  
{ "version": "9120", "device": "fmadio100v2", "build": "Wed Sep 27 03:08:27  
2023" } fmadio@fmadio100v2-228U:$
```

#### Parameters

No parameters

#### Responses

GET /api/v1/system/port\_stats

### Get RMON1 capture port stats

returns RMON1 capture port statistics

Example:

```
fmadio@fmadio100v2-228U:$ curl -s  
http://127.0.0.1/api/v1/system/port_stats | jq
```

```
{
```

```
"cap0":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,  
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"cap1":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,  
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"cap2":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,  
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"cap3":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,  
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"cap4":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,  
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"cap5":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,
```



```
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"cap6":
```

```
{ "Pkt": 7237, "Byte": 1936022, "Pkt_RUNT": 0, "Pkt_64": 55,  
"Pkt_65_127": 559, "Pkt_128_255": 257, "Pkt_256_511": 6136,  
"Pkt_512_1023": 180, "Pkt_1024_1518": 50, "Pkt_1024_2047": 50,  
"Pkt_2048_4095": 0, "Pkt_4096_8191": 0, "Pkt_8192_9216": 0,  
"Pkt_OVER": 0 },
```

```
"cap7":
```

```
{ "Pkt": 0, "Byte": 0, "Pkt_RUNT": 0, "Pkt_64": 0, "Pkt_65_127": 0,  
"Pkt_128_255": 0, "Pkt_256_511": 0, "Pkt_512_1023": 0,  
"Pkt_1024_1518": 0, "Pkt_1024_2047": 0, "Pkt_2048_4095": 0,  
"Pkt_4096_8191": 0, "Pkt_8192_9216": 0, "Pkt_OVER": 0 },
```

```
"port_config": "8x10G"
```

```
}
```

```
fmadio@fmadio100v2-228U:$
```

## Parameters

No parameters

## Responses

✓ GET /api/v1/system/io\_stats

### Get IO Statistics on the system

Example:

```
fmadio@fmadio100v2-228U:$ curl -s  
http://127.0.0.1/api/v1/system/io_stats | jq  
{  
  "timestamp": "1696925340903886080",  
  "ioqueue_active": "1"  
}  
fmadio@fmadio100v2-228U:$
```

### Parameters

No parameters

### Responses

GET /api/v1/system/status

## Get system status

Returns the system status, this is identical to the Telemetry data

<https://docs.fmad.io/fmadio-documentation/monitoring/syslog-fmadio100gv2>

Example output:

```
fmadio@fmadio100v2-228U:~$ curl -s http://127.0.0.1/api/v1/system/status
```

Resulting JSON blob

```
{
  "timestamp":1695785446,"ver":"9120","temperature":
  {
    "module":"system","subsystem":"temperature","timestamp":1695785446,"ver":"9120","Temperature_CPU0":55.00,"Temperature_CPU1":70.00,"Temperature_PCH":46.00,"Temperature_SYS":42.00,"Temperature_PER":24.00,"Temperature_NIC":49.00,"Temperature_AirIn":24.00,"Temperature_AirOut":0.00,"Temperature_Transceiver0":42.00,"Temperature_Transceiver1":42.00},
    "fan":
    {
      "module":"system","subsystem":"fan","timestamp":1695785446,"ver":"9120","Fan_SYS0":21450,"Fan_SYS1":21450,"Fan_SYS2":21300,"Fan_SYS3":21450,"Fan_SYS4":21450,"Fan_SYS5":21450,"Fan_SYS6":21600,"Fan_SYS7":21450},
      "disk":
      {
        "module":"system","subsystem":"disk","timestamp":1695785446,"ver":"9120","FreeGB_System":8.977,"FreeGB_Store0":4720.779,"FreeGB_Store1":0.000,"FreeGB_Remote0":46349.530,"FreeGB_Remote1":46349.530,"DiskPresent_os0":true,"DiskTemperature_os0":40,"DiskSMART_os0":0,"DiskPresent_ssd0":true,"DiskTemperature_ssd0":36,"DiskSMART_ssd0":0,"DiskPresent_ssd1":true,"DiskTemperature_ssd1":34,"DiskSMART_ssd1":0,"DiskPresent_ssd2":true,"DiskTemperature_ssd2":35,"DiskSMART_ssd2":0,"DiskPresent_ssd3":true,"DiskTemperature_ssd3":34,"DiskSMART_ssd3":0,"DiskPresent_ssd4":true,"DiskTemperature_ssd4":35,"DiskSMART_ssd4":0,"DiskPresent_ssd5":true,"DiskTemperature_ssd5":40,"DiskSMART_ssd5":0,"DiskPresent_ssd6":true,"DiskTemperature_ssd6":36,"DiskSMART_ssd6":0,"DiskPresent_ssd7":true,"DiskTemperature_ssd7":37,"DiskSMART_ssd7":0,"DiskPresent_par0":true,"DiskTemperature_par0":34,"DiskSMART_par0":0},
        "link":
        {
          "module":"system","subsystem":"link","timestamp":1695785446,"ver":"9120","cap0_link":true,"cap1_link":true,"cap2_link":true,"cap3_link":true,"cap4_link":true,"cap5_link":true,"cap6_link":true,"cap7_link":true,"man0_link":true,"man10_link":true},
          "io":
          {
            "module":"system","subsystem":"io","timestamp":1695785446,"ver":"9120","DiskRdGbps":0.40,"DiskWrGbps":0.25},
            "capture":
            {
              "module":"system","subsystem":"capture","timestamp":1695785446,"ver":"9120","CaptureEnb":true,"CapturePkt":10497009,"CaptureByte":14691374307,"CaptureDrop":0,"CaptureFCSError":0,"CaptureRateGbps":0.237402,"CaptureRateMpps":0.020893,"CaptureName":"wan_colo0_20230927_0320","CapturePort0_Byte":0,"CapturePort0_Pkt":0,"CapturePort1_Byte":14734141814,"CapturePort1_Pkt":10527209,"CapturePort2_Byte":0,"CapturePort2_Pkt":0,"CapturePort3_Byte":0,"CapturePort3_Pkt":0,"CapturePort4_Byte":0,"CapturePort4_Pkt":0,"CapturePort5_Byte":0,"CapturePort5_Pkt":0,"CapturePort6_Byte":0,"CapturePort6_Pkt":0,"CapturePort7_Byte":0,"CapturePort7_Pkt":0},
              "power":
              {
                "module":"system","subsystem":"power","timestamp":1695785446,"ver":"9120","PSU0_Status":false,"PSU1_Status":true,"PSU_PowerWatt":370},
                "other":
                {
                  "module":"system","subsystem":"other","timestamp":1695785446,"ver":"9120","UptimeHour":0.33,
```

"MemFree":352248299520,"MemErrorECC":0,"MemCached":17432817664,"MemMapped":5480136704,"MemBuffer":57147392,"MemDirty":0,"PageInByte":0,"FDCnt":1198,"WritebackB":0,"WritebackPct":0.000000,"WritebackDropTotalG":0.00,"WritebackDropG":0.00,"CacheSize":3072604713792,"StoreSize":30725994708992,"CPULoad":5.10,"SerialNo":"undef-undef-e0d55e5d2150","PortConfig":"8x10G","Version":"fmadio100v2:9120pcap2json:715"},"cat":{"module":"system","subsystem":"cat","timestamp":1695785446,"ver":"9120","cat\_0\_Enable":true,"cat\_0\_Mode":"FMADRing","cat\_0\_CPUMain":0,"cat\_0\_TSPCAP":1695785445,"cat\_0\_ReadPkt":21098,"cat\_0\_ReadByte":30071136,"cat\_0\_ReadTotalPkt":10510527,"cat\_0\_ReadTotalByte":14779806736,"cat\_0\_ReadGbps":0.231257,"cat\_0\_ReadMpps":0.020281,"cat\_0\_WritePkt":42196,"cat\_0\_WriteByte":59997961,"cat\_0\_WriteTotalPkt":10510527,"cat\_0\_WriteTotalByte":14779806736,"cat\_0\_WriteGbps":0.461405,"cat\_0\_WriteMpps":0.040563,"cat\_0\_PendingByte":30670848,"cat\_0\_PktDiscard":0,"cat\_0\_PktDiscardTotal":0,"cat\_0\_PktSlice":0,"cat\_0\_IOPriority":20,"cat\_0\_ChunkID":10996718,"cat\_0\_CmdLine":"/opt/fmadio/bin/stream\_cat--uidpush\_pcap\_1695784865262465024--follow-start--nop-truncate--ring-eof--ring/opt/fmadio/queue/pcap\_ring\_all1sec--ring-cpu/opt/fmadio/queue/pcap\_ring\_all1sec23--ring-filter-bpf/opt/fmadio/queue/pcap\_ring\_all1sec--ring-filter-frame/opt/fmadio/queue/pcap\_ring\_all1sec--ring/opt/fmadio/queue/pcap\_ring\_icmp--ring-cpu/opt/fmadio/queue/pcap\_ring\_icmp23--ring-filter-bpf/opt/fmadio/queue/pcap\_ring\_icmpicmp--ring-filter-frame/opt/fmadio/queue/pcap\_ring\_icmp","cat\_0\_StreamName":"wan\_colo0\_20230927\_0320","cat\_0\_FilterBPF":"","cat\_0\_CPUIdle":0.7622,"cat\_0\_CPUFetch":0.0154,"cat\_0\_CUPProcess":0.0154,"cat\_0\_CPUSend":0.0000,"cat\_1\_Enable":true,"cat\_1\_Mode":"FMADRing","cat\_1\_CPUMain":0,"cat\_1\_TSPCAP":1695785443,"cat\_1\_ReadPkt":199179,"cat\_1\_ReadByte":283074384,"cat\_1\_ReadTotalPkt":9743856,"cat\_1\_ReadTotalByte":13763158272,"cat\_1\_ReadGbps":0.226272,"cat\_1\_ReadMpps":0.019901,"cat\_1\_WritePkt":199179,"cat\_1\_WriteByte":281761649,"cat\_1\_WriteTotalPkt":9743856,"cat\_1\_WriteTotalByte":13763158272,"cat\_1\_WriteGbps":0.225223,"cat\_1\_WriteMpps":0.019901,"cat\_1\_PendingByte":33030144,"cat\_1\_PktDiscard":0,"cat\_1\_PktDiscardTotal":0,"cat\_1\_PktSlice":0,"cat\_1\_IOPriority":20,"cat\_1\_ChunkID":10996554,"cat\_1\_CmdLine":"/opt/fmadio/bin/stream\_cat--uidpush\_lxc\_1695784924583593984--follow--ring/opt/fmadio/queue/lxc\_market2json\_euronext\_sbe--ring-filter-bpf/opt/fmadio/queue/lxc\_market2json\_euronext\_sbevlan177andnet224.0.208.0/24--ring-filter-frame/opt/fmadio/queue/lxc\_market2json\_euronext\_sbe--cpu23-v--print-period10e9","cat\_1\_StreamName":"wan\_colo0\_20230927\_0320","cat\_1\_FilterBPF":"","cat\_1\_CPUIdle":0.9921,"cat\_1\_CPUFetch":0.0071,"cat\_1\_CUPProcess":0.0071,"cat\_1\_CPUSend":0.0000,"cat\_2\_Enable":false,"cat\_2\_Mode":"","cat\_2\_CPUMain":0,"cat\_2\_TSPCAP":0,"cat\_2\_ReadPkt":0,"cat\_2\_ReadByte":0,"cat\_2\_ReadTotalPkt":0,"cat\_2\_ReadTotalByte":0,"cat\_2\_ReadGbps":0.000000,"cat\_2\_ReadMpps":0.000000,"cat\_2\_WritePkt":0,"cat\_2\_WriteByte":0,"cat\_2\_WriteTotalPkt":0,"cat\_2\_WriteTotalByte":0,"cat\_2\_WriteGbps":0.000000,"cat\_2\_WriteMpps":0.000000,"cat\_2\_PendingByte":0,"cat\_2\_PktDiscard":0,"cat\_2\_PktDiscardTotal":0,"cat\_2\_PktSlice":0,"cat\_2\_IOPriority":0,"cat\_2\_ChunkID":0,"cat\_2\_CmdLine":"","cat\_2\_StreamName":"","cat\_2\_FilterBPF":"","cat\_2\_CPUIdle":0.0000,"cat\_2\_CPUFetch":0.0000,"cat\_2\_CUPProcess":0.0000,"cat\_2\_CPUSend":0.0000,"cat\_3\_Enable":false,"cat\_3\_Mode":"","cat\_3\_CPUMain":0,"cat\_3\_TSPCAP":0,"cat\_3\_ReadPkt":0,"cat\_3\_ReadByte":0,"cat\_3\_ReadTotalPkt":0,"cat\_3\_ReadTotalByte":0,"cat\_3\_ReadGbps":0.000000,"cat\_3\_ReadMpps":0.000000,"cat\_3\_WritePkt":0,"cat\_3\_WriteByte":0,"cat\_3\_WriteTotalPkt":0,"cat\_3\_WriteTotalByte":0,"cat\_3\_WriteGbps":0.000000,"cat\_3\_WriteMpps":0.000000,"cat\_3\_PendingByte":0,"cat\_3\_PktDiscard":0,"cat\_3\_PktDiscardTotal":0,"cat\_3\_PktSlice":0,"cat\_3\_IOPriority":0,"cat\_3\_ChunkID":0,"cat\_3\_CmdLine":"","cat\_3\_StreamName":"","cat\_3\_FilterBPF":"","cat\_3\_CPUIdle":0.0000,"cat\_3\_CPUFetch":0.0000,"cat\_3\_CUPProcess":0.0000,"cat\_3\_CPUSend":0.0000,"c

```
at_4_Enable":false,"cat_4_Mode":"","cat_4_CPUMain":0,"cat_4_TSPCAP":0,"cat_4_ReadPkt":0,"
cat_4_ReadByte":0,"cat_4_ReadTotalPkt":0,"cat_4_ReadTotalByte":0,"cat_4_ReadGbps":0.00000
0,"cat_4_ReadMpps":0.000000,"cat_4_WritePkt":0,"cat_4_WriteByte":0,"cat_4_WriteTotalPkt":0,"c
at_4_WriteTotalByte":0,"cat_4_WriteGbps":0.000000,"cat_4_WriteMpps":0.000000,"cat_4_Pendin
gByte":0,"cat_4_PktDiscard":0,"cat_4_PktDiscardTotal":0,"cat_4_PktSlice":0,"cat_4_IOPriority":0,"
cat_4_ChunkID":0,"cat_4_CmdLine":"","cat_4_StreamName":"","cat_4_FilterBPF":"","cat_4_CPUI
dle":0.0000,"cat_4_CPUFetch":0.0000,"cat_4_CPUProcess":0.0000,"cat_4_CPUSend":0.0000,"c
at_5_Enable":false,"cat_5_Mode":"","cat_5_CPUMain":0,"cat_5_TSPCAP":0,"cat_5_ReadPkt":0,"
cat_5_ReadByte":0,"cat_5_ReadTotalPkt":0,"cat_5_ReadTotalByte":0,"cat_5_ReadGbps":0.00000
0,"cat_5_ReadMpps":0.000000,"cat_5_WritePkt":0,"cat_5_WriteByte":0,"cat_5_WriteTotalPkt":0,"c
at_5_WriteTotalByte":0,"cat_5_WriteGbps":0.000000,"cat_5_WriteMpps":0.000000,"cat_5_Pendin
gByte":0,"cat_5_PktDiscard":0,"cat_5_PktDiscardTotal":0,"cat_5_PktSlice":0,"cat_5_IOPriority":0,"
cat_5_ChunkID":0,"cat_5_CmdLine":"","cat_5_StreamName":"","cat_5_FilterBPF":"","cat_5_CPUI
dle":0.0000,"cat_5_CPUFetch":0.0000,"cat_5_CPUProcess":0.0000,"cat_5_CPUSend":0.0000,"c
at_6_Enable":false,"cat_6_Mode":"","cat_6_CPUMain":0,"cat_6_TSPCAP":0,"cat_6_ReadPkt":0,"
cat_6_ReadByte":0,"cat_6_ReadTotalPkt":0,"cat_6_ReadTotalByte":0,"cat_6_ReadGbps":0.00000
0,"cat_6_ReadMpps":0.000000,"cat_6_WritePkt":0,"cat_6_WriteByte":0,"cat_6_WriteTotalPkt":0,"c
at_6_WriteTotalByte":0,"cat_6_WriteGbps":0.000000,"cat_6_WriteMpps":0.000000,"cat_6_Pendin
gByte":0,"cat_6_PktDiscard":0,"cat_6_PktDiscardTotal":0,"cat_6_PktSlice":0,"cat_6_IOPriority":0,"
cat_6_ChunkID":0,"cat_6_CmdLine":"","cat_6_StreamName":"","cat_6_FilterBPF":"","cat_6_CPUI
dle":0.0000,"cat_6_CPUFetch":0.0000,"cat_6_CPUProcess":0.0000,"cat_6_CPUSend":0.0000,"c
at_7_Enable":false,"cat_7_Mode":"","cat_7_CPUMain":0,"cat_7_TSPCAP":0,"cat_7_ReadPkt":0,"
cat_7_ReadByte":0,"cat_7_ReadTotalPkt":0,"cat_7_ReadTotalByte":0,"cat_7_ReadGbps":0.00000
0,"cat_7_ReadMpps":0.000000,"cat_7_WritePkt":0,"cat_7_WriteByte":0,"cat_7_WriteTotalPkt":0,"c
at_7_WriteTotalByte":0,"cat_7_WriteGbps":0.000000,"cat_7_WriteMpps":0.000000,"cat_7_Pendin
gByte":0,"cat_7_PktDiscard":0,"cat_7_PktDiscardTotal":0,"cat_7_PktSlice":0,"cat_7_IOPriority":0,"
cat_7_ChunkID":0,"cat_7_CmdLine":"","cat_7_StreamName":"","cat_7_FilterBPF":"","cat_7_CPUI
dle":0.0000,"cat_7_CPUFetch":0.0000,"cat_7_CPUProcess":0.0000,"cat_7_CPUSend":0.0000,"c
at_EnableCnt":2,"cat_ReadPkt":220277,"cat_ReadByte":313145520,"cat_ReadTotalPkt":2025438
3,"cat_ReadTotalByte":28542965008,"cat_ReadGbps":0.457529,"cat_ReadMpps":0.457529,"cat_
WritePkt":241375,"cat_WriteByte":341759610,"cat_WriteTotalPkt":20254383,"cat_WriteTotalByte":
28542965008,"cat_WriteGbps":0.686627,"cat_WriteMpps":0.686627},"ptp":
{"module":"system","subsystem":"ptp","timestamp":1695785446,"ver":"9120","TimeFPGA":169578
5446482977592,"TimeSYS":1695785446286598912,"GMOffset":0.00,"GMSync":false,"GMMaster"
:"", "SysOffset":0.00,"SysSync":false,"iSysOffset":0.00,"iSysSync":true,"NTPOffset":0.00,"NTPSync
":false,"NTPMaster":"","GMUpTime":0,"SysUptime":0,"iSysUptime":1061,"PPSUptime":690,"NTPU
ptime":0,"PPSPeriod":6.399925850,"PPSOffset":-867,"PPSdPhase":2065,"PPSdZero":-867,"PPS
Cnt":1225,"Clk156Period":0.000000,"Clk156Offset":0,"Clk250Period":0.000000,"Clk250Offset":0,"
Clk322Period":0.000000,"Clk322Offset":0}}
```

Pretty print output

## Parameters

No parameters

## Responses

### Legacy Download

NOTE: These interfaces are legacy, recommend using the V1 API interfaces

✓ **GET** /stream/list

#### List All Captures

Lists all captures on the device.

#### Parameters

No parameters

#### Responses

● 200

>

✓ GET /stream/ssize?StreamName=<capture sname>&StreamView=<split mode>

## Split Capture by filesize

Lists splits for a specific capture based on file size.

Usually this is a 2 step process of

- 1) get the split list
- 2) download a specific split.

## Parameters

### Query

**StreamView** string

Stream time slice name

split\_10MB  
split\_100MB  
split\_250MB  
split\_1GB  
split\_2GB  
split\_5GB  
split\_10GB  
split\_100GB  
split\_1TB

---

**StreamName** string

Stream capture name

## Responses

● 200

>

✓ GET /stream/stime?StreamName=<capture sname>&StreamView=<split mode>

## Split Capture by time

Lists splits for a specific capture based on a time unit.

Usually this is a 2 step process of

- 1) get the split list
- 2) download a specific split

## Parameters

### Query

**StreamView** string

Split options for the time split

split\_1sec  
split\_10sec  
split\_1min  
split\_10min  
split\_15min  
split\_1hour  
split\_2hour  
split\_4hour  
split\_6hour  
split\_8hour  
split\_12hour

---

**StreamName** string

Stream capture name.

## Responses

● 200

>



✓ GET /pcap/single?StreamName=<capture name>&FilterRE=<string>

## Single PCAP Download

Download entire capture as a single file.  
Piping to a file or any other analysis tools is possible.

Compression example:

```
curl -u fmadio:100g "http://192.168.2.75/pcap/single?StreamName=TestCapture_20180702_1127&Compression=fast"
```

FilterBPF example:

```
curl -u fmadio:100g "http://192.168.2.75/pcap/single?StreamName=hitcon_20180702_1503_58&" -G --data-urlencode "FilterBPF=tcp"
```

## Parameters

### Query

**FilterRE** string

Download the capture with using a RegEx DPI filter.

**FilterBPF** string

BPF Filter to be applied to the stream

**Compression** string

Compress the returned stream with gzip.  
'fast' Fastest compression but not smallest  
'best' Slowest compression smallest size  
1-9 The range from 'fast' to 'best'

**StreamName** string

Stream capture name.

## Responses

● 200

>

GET /pcap/splittime?StreamName=<string>&Start=<int>&Stop=<int>&FilterBPF=<string>&FilterPort=<int>

## Split PCAP Time Download

Download the capture with a time filter.

Note: the nanosecond Epoch Start is 1530498788000000000.

Removing the nanosecond part convert epoch to date/time.

### Parameters

#### Query

**FilterPort** integer

Download the capture specifying the port capture number.

**FilterBPF** string

BPF Filter to be applied to the stream.

**StreamName\*** string

Stream capture name.

**Stop\*** integer

Stop time in nanoseconds epoch.

**Start\*** integer

Start time in nanoseconds epoch.

### Responses

● 200

PCAP Data stream. Usually used with tools like curl.

>

✓ GET /pcap/timerange?TSBegin=<epoch start>&TSEnd=<epoch stop>&TSMMax=<size>&TSMMode=<nanos or msec>

## TimeRange PCAP Download

Download a timerange of pcap data without any capture file referenced. The system will search all captures for the specified timerange.

At most it can cross two pcap files

### Parameters

#### Query

TSMMax integer

Maximum nanosecond of packets to download.

TSMMode string

Time Range mode to use for time:

nsec | epoch in nano seconds (default)

usec | epoch in micro seconds

msec | epoch in milli seconds

sec: | epoch in seconds

TSBegin\* integer

Start time in epoch (default nano seconds)

TSEnd\* integer

Stop time in epoch (default nano seconds).

### Responses

● 200

>

# Examples

A guide for developers to access a FMADIO using the web based API. Examples are provided for all endpoints.

## FMADIO API

The examples show how to use the different parameters for the uri endpoint.

**Note:** Replace the IP 127.0.0.1 with the host IP of your FMADIO device.

### Status

```
curl -u fmadio:100g "http://127.0.0.1/sysmaster/status"
```

### Device Status

```
curl -u fmadio:100g "http://127.0.0.1/sysmaster/stats_summary"
```

### CaptureList

```
curl -u fmadio:100g "http://127.0.0.1/stream/list"
```

### Capture Split By Filesize

```
curl -u fmadio:100g "http://127.0.0.1/stream/ssize?  
StreamName=stream_test_001&  
StreamView=split_1GB&"
```

### Capture Split By Time

```
curl -u fmadio:100g "http://127.0.0.1/stream/stime?  
StreamName=stream_test_001&  
StreamView=split_1sec&"
```

## Legacy Interface

### Single

**StreamName** only:

```
curl -u fmadio:100g "http://127.0.0.1/pcap/single?StreamName=stream_test_001"
```

**StreamName** and **FilterBPF**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/single?StreamName=stream_test_001&" -G --data-url
```

**StreamName** and **Compression**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/single?StreamName=stream_test_001&Compression=fas
```

**StreamName** and **FilterRE**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/single?StreamName=stream_test_001&" -G --data-url
```

**StreamName**, **Compression** and **FilterBPF**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/single?StreamName=stream_test_001&Compression=fas
```

**SplitTime**

**StreamName**, **Start** and **Stop**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/splittime?StreamName=stream_test_001&Start=153049
```

**StreamName**, **Start**, **Stop** and **FilterBPF**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/splittime?StreamName=stream_test_001&Start=153049
```

**StreamName**, **Start**, **Stop** and **FilterPort**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/splittime?StreamName=stream_test_001&Start=153049
```

**TimeRange**

**TSBegin** and **TSEnd**

```
curl -u fmadio:100g "http://127.0.0.1/pcap/timerange?TSBegin=1497329459948411420&TSEnd=15973
```

## TSBegin, TSEnd, TSMODE and TSMAX

```
curl -u fmadio:100g "http://127.0.0.1/pcap/timerange?TSBegin=1497329459948411420&TSEnd=15973
```

## V1 API

The examples show how to use the different parameters for the uri endpoint.

**Note:** Replace the IP 127.0.0.1 with the host IP of your FMADIO device.

### API v1 - Single

**StreamName only.**

```
curl -u fmadio:xxxxx "http://127.0.0.1/api/v1/pcap/single?
StreamName=stream_test"
```

**StreamName and FilterBPF**

```
curl -u fmadio:xxxxx "http://127.0.0.1/api/v1/pcap/single?
StreamName=stream_test_001"
-G --data-urlencode "FilterBPF=tcp"
```

**StreamName and Compression**

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/single?
StreamName=stream_test_001&
Compression=fast"
```

**StreamName, Compression and FilterBPF**

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/single?
StreamName=stream_test_001&
Compression=fast"
-G --data-urlencode "FilterBPF=tcp"
```

### API v1 - SplitTime

**StreamName, Start and Stop**

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/splittime?
StreamName=stream_test_001&
Start=1530498788000000000&
Stop=1530498789000000000"
```

### StreamName, Start, Stop and FilterBPF

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/splittime?
StreamName=stream_test_001&
Start=1530498788000000000&
Stop=1530498789000000000"
-G --data-urlencode "FilterBPF=tcp"
```

### StreamName, Start, Stop, FilterBPF and Compression

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/splittime?
StreamName=stream_test_001&
Start=1530498788000000000&
Stop=1530498789000000000&
Compression=fast"
-G --data-urlencode "FilterBPF=tcp"
```

### StreamName, Start, Stop and Compression

```
curl -u fmadio:xxxx"http://127.0.0.1/api/v1/pcap/splittime?
StreamName=stream_test_001&
Start=1530498788000000000&
Stop=1530498789000000000&
Compression=fast"
```

## API v1 - TimeRange

The Time Range function is very useful as the FMADIO system will work out which (or multiple) captures to check based on the Epoch Time stamp value.

### TSBegin and TSEnd

### Nanosecond Epoch selection

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
TSBegin=1621772572136996000&
TSEnd=1621774913584264000"
```

## Second Epoch time Selection

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSUnit=sec&
  TSBegin=1621772572&
  TSEnd=1621774913"
```

## Year Month Day Hour Min

Using the TSUnit option can use a more friendly time selection.

By default it uses the TimeZone configured on the system

Fetch PCAP from 3AM to 4AM on 2023 / 10 (October) / 1st

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSUnit=YYYYMMDD_HHMMSS&
  TSBegin=20231001_030000&
  TSEnd=20231001_040000"
```

## Year Month Dat Hour Min with Timezone

Same as above but specifying the timezone.

NOTE: if using the full `TSZone = Asia/Singapore` for example, CURL will append a ? to the URL. Its recommended to use the City name only to avoid confusing CURL.

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSUnit=YYYYMMDD_HHMMSS&
  TSZone=Singapore&
  TSBegin=20231001_030000&
  TSEnd=20231001_040000"
```

## Year Month Day Hour Min Timzone and BPF Filter with De-encapsulation

Specifying all of the above with a BPF Filter, with BPF De-encapsulation enabled

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSUnit=YYYYMMDD_HHMMSS&
  TSZone=Singapore&
  TSBegin=20231001_030000&
  TSEnd=20231001_040000&
  FilterBPFDecap=true&"
-G --data-urlencode "FilterBPF=icmp"
```



## TSBegin, TSEnd and TSMMax

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSBEGIN=1621772572136996000&
  TSEnd=1621774913584264000&
  TSMMax=100000"
```

## TSBegin, TSEnd and FilterBPF

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSBEGIN=1621772572136996000&
  TSEnd=1621774913584264000"
-G --data-urlencode "FilterBPF=tcp"
```

## TSBegin, TSEnd and Compression

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSBEGIN=1621772572136996000&
  TSEnd=1621774913584264000&
  Compression=fast"
```

## TSBegin, TSEnd, FilterBPF and Compression

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSBEGIN=1621772572136996000&
  TSEnd=1621774913584264000&
  Compression=fast"
-G --data-urlencode "FilterBPF=tcp"
```

## TSBegin, TSEnd, FilterBPF and FilterFrame

Frame Filters based on FMADIO Capture system

Filter based on FMADIO Capture port number

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBEGIN=1658744408270221800&
  TSEnd=1658744501189259300&"
-G --data-urlencode "FilterFrame=capture.port==0"
| tcpdump -r - -nn
| head
```

Filter based on multiple FMADIO Capture port numbers

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=capture.port==0,1,2,3"
  | tcpdump -r - -nn
  | head
```

Filter based on exclude FMADIO Capture port numbers

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=capture.port!=0"
  | tcpdump -r - -nn
  | head
```

Frame filters specific to 7130 (Metamako) hardware footer

Filter for a specific 7130 Device 54932 (any port)

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=a7130.srcdevice==54932"
  | tcpdump -r - -nn
  | head
```

Filter for everything except a specific 7130 Device (not device id 54932)

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=a7130.srcdevice!=54932"
  | tcpdump -r - -nn
  | head
```

Filter for a specific 7130 Port number 1

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=a7130.srcport==1"
  | tcpdump -r - -nn
  | head
```

Filter for multiple 7130 Port numbers 1, 2, 3, 5, 10

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=a7130.srcport==1,2,3,5,10"
  | tcpdump -r - -nn
  | head
```

Filter for everything except 7130 Port number 10

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&"
  -G --data-urlencode "FilterFrame=a7130.srcport!=10"
  | tcpdump -r - -nn
  | head
```

Filter on a specific 7130 Port number and use the 7130 Footer Timestamp as the PCAP timestamp.  
Overriding the current TimeStamp setting

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&TSMode=arista7130"
  -G --data-urlencode "FilterFrame=a7130.srcport!=10"
  | tcpdump -r - -nn
  | head
```

Frame filters specific to Cisco 3550 (Exablaze) hardware footer

Filter on a specific ingress port of the Cisco 3550, and use the Footer timestamp as the PCAP timestamp.

```
curl "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1658744408270221800&
  TSEnd=1658744501189259300&TSMode=cisco3550"
  -G --data-urlencode "FilterFrame=c3550.srcport==48"
  | tcpdump -r - -nn
  | head
```

## Miscellaneous Examples

### Encapsulation Debug

Many times the exact packet encapsulation is unclear, the following uses a Wireshark filter expression to extract and show the full encapsulation format of the packet. From this a high speed BPF filter can be used to process the data.

In the below example we are using the Wireshark filter "ip.addr == 192.168.1.1" on a historical capture.

```
curl -u fmadio:xxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1666706401000000000&
  TSEnd=1666706401010000000"
| tshark -r - -T fields -e frame.protocols -e ip.src -e ip.dst
-Y "ip.addr == 192.168.1.1"
```

Alternatively running on the currently running capture via SSH on the fmadio box looks like the following. This example filters on any UDP traffic.

```
sudo stream_cat
| tshark -r - -T fields -e frame.protocols -e ip.src -e ip.dst -Y "udp"
| head
```

The output looks like the following

```
eth:ethertype:vlan:ethertype:ip:udp:ntp 106.10.186.200 192.168.133.10
eth:ethertype:vlan:ethertype:ip:udp:ntp 106.10.186.201 192.168.133.10
eth:ethertype:vlan:ethertype:ip:udp:ntp 167.172.70.21 192.168.133.10
eth:ethertype:vlan:ethertype:ip:udp:ntp 106.10.186.200 192.168.133.10
eth:ethertype:vlan:ethertype:ip:udp:ntp 106.10.186.201 192.168.133.10
eth:ethertype:vlan:ethertype:ip:udp:ntp 167.172.70.21 192.168.133.10
```

The above output shows there is a single VLAN tag in the packet. Making the equivalent BPF filter

```
vlan and udp
```

With the final BPF filter using a CURL request

```
curl -u fmadio:xxxxx "http://127.0.0.1/api/v1/pcap/timerange?
  TSBegin=1671407102&
  TSEnd=1671407752&
  TSMODE=sec&"
-G --data-urlencode "FilterBPF=vlan and udp"
| tcpdump -r - -n
| head
```

Output per below

23:47:45.409489 IP 106.10.186.201.123 > 192.168.133.10.123: NTPv4, Server, length 48

23:52:14.407364 IP 167.172.70.21.123 > 192.168.133.10.123: NTPv4, Server, length 48

23:55:42.405072 IP 106.10.186.200.123 > 192.168.133.10.123: NTPv4, Server, length 48

# Monitoring

# Dashboard

The FMADIO 100G Packet capture dashboard provides a high level overview of the capture system. Example is shown below.

NOTE: Different hardware platforms 20G, 40G, 100G have slightly different dashboard settings.

SYSTEM STATUS							
CAPTURE RECEIVED	14,196,032	CAPTURE BYTES	92,500,608,500	CAPTURE ERRORS	0	CAPTURE DROPPED	0
GENERATE SENT	0	GENERATE BYTES	0	GENERATE ERRORS	0	STORAGE BYTES	2,240,494,698,496
SMART ERRORS	0	RAID STATUS	GOOD	SYSTEM WARNING	0	RAM ECC ERRORS	0
						UP TIME	00 04 41M
						CAPTURE DAYS	21D 22H 10M

FMADIO 100G Gen2 Dashboard Status

Going over the above in more detail as follows

## Capture Received

Total number of packets received and successfully stored on the capture device

## Capture Bytes

Total number of bytes successfully captured and stored on the device.

## Capture Errors

Errors with packet data seen on the wire. This includes Frame Check Sequence (FCS) errors.

## Capture Dropped

Total number of packets dropped / unable to make it to the capture storage system

## Capture Days

Total number of days worth of capture on the system. Calculated as difference between the oldest packet on the system vs the newest packet on the system

## Generate Sent

Total number of packets generated / transmitted on the device. This includes packet blaster and packet replay counters

## Generate Bytes

Total number of bytes generated / transmitted on the device.

## Generate Errors

Total number of errors occurred during packet generation

## Smart Errors

Total number of new SMART Disk errors

## RAID Status

Current RAID status, GOOD or DEGRADED or FAILED

## System Warning

Counter of system related warning or errors. Currently this is unused

## RAM ECC Errors

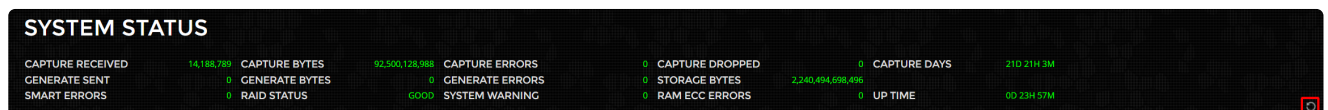
Total number of DDR4 System RAM ECC Errors

## Up Time

Total uptime of the system

## Counter Reset

Most of the counters can be reset by clicking on the small circle arrow highlighted in red below.



**SYSTEM STATUS**

CAPTURE RECEIVED	14,188,789	CAPTURE BYTES	92,500,128,988	CAPTURE ERRORS	0	CAPTURE DROPPED	0	CAPTURE DAYS	21D 21H 3M
GENERATE SENT	0	GENERATE BYTES	0	GENERATE ERRORS	0	STORAGE BYTES	2,240,494,698,496	UP TIME	0D 23H 57M
SMART ERRORS	0	RAID STATUS	GOOD	SYSTEM WARNING	0	RAM ECC ERRORS	0		

A small red square icon with a white circle and arrow is located in the bottom right corner of the dashboard.



# Alerts

FW Version: 7611+

Alerts can be generated by system automatically either

- EMail alerts
- Syslog alerts
- SNMP Traps

Alert configuration file is located in

```
/opt/fmadio/etc/alert.lua
```

By default all Alert triggers are disabled.

## Example Alert.lua

An example alert.lua file is shown below. If the file does not exist please create.

```
local L =
{
["AlertList"] =
{
    LinkState           = true,
    ByteCache           = 1e12,
    BytesOverflow       = true,
    PacketError         = true,
    PacketDrop          = true,
    CaptureState        = true,
    DiskSMART           = true,
    DiskFreeStore0      = 1e9,
    DiskFreeStore1      = 0,
    DiskFreeRemote0     = 0,
    CPUtemperature      = 80,
    FANAlert            = true,
    PSUAlert            = true,

    Sleep               = 60,           -- how long to sleep when an alert is tr
}
}
return L
```

## Triggers

System has can trigger an a small but well defined list of critical Events. The following is a description and example for each item. Triggers are enabled or disabled in the following part of the configuration file. Each line enables/disabled or puts a threshold on the trigger

```
["AlertList"] =
{
  LinkState           = true,
  CaptureState       = true,
  ByteCache          = 1e12,
  BytesOverflow      = true,
  PacketError        = true,
  PacketDrop         = true,
  DiskFreeStore0     = 1e9,
  DiskFreeStore1     = 0,
  DiskFreeRemote0    = 0,
  DiskSMART           = true,
  DiskError          = true,
  Sleep              = 60,
}
```

Each trigger is described below.

### LinkState (Capture Port State)

Monitoring the capture link status is critical to ensure no data is lost. Enabling this option will alert when a capture link goes up or down.

### Config

```
LinkState           = true,
```

### SYSLOG

```
2021.07.05-07:53:49.790767 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

### SNMP

```
fradioCapture0Link  
fradioCapture1Link  
fradioCapture2Link  
fradioCapture3Link  
fradioCapture4Link  
fradioCapture5Link  
fradioCapture6Link  
  
fradioCapture7Link
```

## CaptureState

Capture State shows the capture is active or in-active. When using in alert mode it will trigger anytime the capture state changes

### Config

```
CaptureState = true,
```

### SYSLOG

```
2021.12.25-14:33:24.849212 (+09:00) | fradio20v3-287 | local7.alert | fradio | Alert
```

### SNMP

```
fradioCaptureEnable
```

## Bytes Cached

Bytes Cached indicates how much capture data has been written to SSD, but not written back into long term storage yet. e.g. Its the delta between the capture SSD rate, and the HDD magnetic storage writeback. Trigger on for example 3TB here provides a good indication the HDD writeback process is running too slow for the sustained incoming capture rate.

### Config

(example trigger once Cache goes over 1TB)

```
ByteCache = 1e12,
```

### SYSLOG

```
2021.07.05-07:39:40.545686 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

SNMP

```
fmadioCaptureCache
```

### BytesOverflow (trigger)

Any time Bytes Over increases an alert is generated. This typically a symptom of capture rates being too high, or HDD writeback too slow (or failing)

### Config

```
BytesOverflow = true,
```

### SYSLOG

```
2021.07.05-08:08:01.038273 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

SNMP

```
fmadioCaptureOverflow
```

### PacketError

Counts FCS errors received on the interface. Any time packet error counts changes an alert is generated. Typically occurs when there are Layer1 link stability issues

### Config

```
PacketError = true,
```

### SYSLOG

```
2021.07.05-08:09:51.481888 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

SNMP

```
fradioCaptureError
```

## PacketDrop

Alerts generated when packets are dropped on the capture device.

## Config

```
PacketDrop = true,
```

## SYSLOG

```
2021.07.05-08:09:51.483071 (+09:00) | fradio20n40v3-363 | local7.alert | fradio | Aler
```

## SNMP

```
fradioCaptureDrop
```

## DiskFreeStore0

When space on /mnt/store0 partition is less than this amount (scientific notation) in bytes. Alerts are generated.

In the below example, an alert is generated when less than 4e9 (4GB) of space is free on /mnt/store0 partition

## Config

```
DiskFreeStore0 = 4e9,
```

## SYSLOG

```
2021.07.05-08:32:10.876238 (+09:00) | fradio20n40v3-363 | local7.alert | fradio | Aler
```

## SNMP

```
fradioDiskFreeStore0
```

## DiskFreeStore1

When space on /mnt/store1 (scratch analytics workspace) is less than this amount (scientific notation) in bytes an Alert is generated

### Config

```
DiskFreeStore1 = 10e9,
```

### SYSLOG

```
2021.07.05-08:32:10.876238 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

### SNMP

```
fmadioDiskFreeStore1
```

## DiskFreeRemote0

When space on the /mnt/remote0 (typically NFS mount partition) is less than this threshold an Alert is generated

### Config

```
DiskFreeRemote0 = 10e9,
```

### SYSLOG

```
2021.07.05-08:32:10.876238 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

### SNMP

```
fmadioDiskFreeRemote0
```

## DiskError

Alerts when there is a disk error or RAID error on the device. For example a disk has been lost or HDD RAID redundancy has been reduced.

## Config

```
DiskError = true,
```

## SYSLOG

```
2021.07.05-08:05:34.224665 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

## SNMP

```
fmadioDiskSMART
```

## DiskSMART

Alerts on the total number of disk SMART errors. The value is aggregated across all disks, please check the system log files for more details about which specific disk is having an issue.

## Config

```
DiskSMART = true,
```

## SYSLOG

```
2021.07.05-08:05:34.224665 (+09:00) | fmadio20n40v3-363 | local7.alert | fmadio | Aler
```

## SNMP

```
fmadioDiskSMART
```

## Sleep

Minimum number of seconds between alert generation. This is to prevent spamming of alerts due to unexpected system conditions.

## SYSLOG Alerts

Alert events are always output to SYSLOG regardless of the other transport modes (email/snmp etc)

SYSLOG logfile is found in

```
/mnt/store0/messages
```

An example syslog alerts as follows.

```
2021.05.24-17:44:20.001457 (+09:00) | fmadio20v3-287 | local7.alert | fmadio | Alert
2021.05.24-17:44:20.002903 (+09:00) | fmadio20v3-287 | local7.alert | fmadio | Alert
2021.05.24-17:44:25.444418 (+09:00) | fmadio20v3-287 | local7.alert | fmadio | Alert
2021.05.24-17:44:25.446463 (+09:00) | fmadio20v3-287 | local7.alert | fmadio | Alert
```

## EMAIL Alerts

Email alerts can be setup as the following, please add the ["Email"] section in the alert configuration file

```
/opt/fmadio/etc/alert.lua
```

An example that sends alerts to the address "alerts@fmad.io" is shown below.

```
local L =
{
  ["Email"] =
  {
    Enable = true,
    To      = "alerts@fmad.io",
    From    = "packet_capture@fmad.io",
  }
,
  ["AlertList"] =
  {
    BytesOverflow = true,
    PacketError   = true,
    PacketDrop    = true,
    DiskFreeStore0 = 4e9,
    Sleep         = 60,
  }
}
return L
```

In addition fmadio packet capture system uses msrtp as the email client, it requires smtp configuration file

```
/opt/fmadio/etc/msmtp.rc
```

Example configuration as follows. Please edit to match the email smtp provider



```
defaults
tls on
tls_certcheck off
logfile /mnt/store0/log/msmtp.log
tls_starttls on

account default
host mail.yourserver.com
port 587
auth on
user fmadio@yourserver.com
password <secrets>
```

## SNMP Broadcast

### FW: 7611+

FMADIO devices can operate in SNMP Broadcast mode. In this mode the system will periodically broadcast all SNMP counter values at a fixed time interval to an SNMP target.

### SNMP MIB

Latest MIB file is found (last updated 2021/12/25)

```
https://fmad.io/download/FMADIOv3-MIB.txt
```

### Config

The general configuration file is used for config

```
/opt/fmadio/etc/time.lua
```

Please edit the section titles ["SNMP"] as follows

```
["SNMP"] =
{
    ["Enable"]      = false,
    ["Trap"]        = false,
    ["Broadcast"]   = true,
    ["BroadcastPeriod"] = 60e9,
    ["Verbose"]     = false,
    ["Target"]      = "127.0.0.1",
    ["ComName"]    = "public",
},
```

The above config enables SNMP Broadcast mode only, while SNMP Trap(Alert) mode is disabled. Broadcast frequency is 60e9 nanoseconds, e.g. every 1 minute.

Broadcast and Trap mode can be use simultaneously if required.

Please update ["Target"] = setting to the correct SNMP collector address. Multiple SNMP targets can be specified separated by spaces. For example

```
["Target"]      = "127.0.0.1 127.0.0.2 127.0.0.3",
```

Example output in broadcast mode is as follows, from the /mnt/store0/log/monitor\_alert.cur logfile

```
SNMP Broadcast: true
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture0Link fmadioCapture0Link i 1
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture1Link fmadioCapture1Link i 1
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture2Link fmadioCapture2Link i 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture3Link fmadioCapture3Link i 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture4Link fmadioCapture4Link i 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture5Link fmadioCapture5Link i 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture6Link fmadioCapture6Link i 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapture7Link fmadioCapture7Link i 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureBps fmadioCaptureBps Integer64 401095.4375
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapturePps fmadioCapturePps Integer64 72.599212646484
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCapturePacket fmadioCapturePacket Counter64 550
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureByte fmadioCaptureByte Counter64 297930
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureCache fmadioCaptureCache Counter64 262144
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureDrop fmadioCaptureDrop Counter64 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureError fmadioCaptureError Counter64 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureOverflow fmadioCaptureOverflow Counter64 0
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioCaptureEnable fmadioCaptureEnable Integer32 1
/usr/bin/snmptrap -v 2c -c public -m /opt/fmadio/etc_ro/FMADIOv3-MIB.txt 127.0.0.1 '' FMADIOv3-MIB::fmadioDiskSMART fmadioDiskSMART Counter64 0
[15:48:49.000.000] Master: Get: 3121376. Diff: 3121377. (0)
```

This translates to

```
15:54:53.859848 IP 127.0.0.1.45808 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44410 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.1 .1.3.6.1.4.1.46891.4.1-1
15:54:53.863176 IP 127.0.0.1.40284 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44411 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.2 .1.3.6.1.4.1.46891.4.2-1
15:54:53.866477 IP 127.0.0.1.53786 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44411 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.3 .1.3.6.1.4.1.46891.4.3-0
15:54:53.869787 IP 127.0.0.1.44384 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44411 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.4 .1.3.6.1.4.1.46891.4.4-0
15:54:53.873113 IP 127.0.0.1.43882 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44412 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.5 .1.3.6.1.4.1.46891.4.5-0
15:54:53.876459 IP 127.0.0.1.34128 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44412 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.6 .1.3.6.1.4.1.46891.4.6-0
15:54:53.879744 IP 127.0.0.1.38479 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44412 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.7 .1.3.6.1.4.1.46891.4.7-0
15:54:53.882995 IP 127.0.0.1.42358 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44413 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.8 .1.3.6.1.4.1.46891.4.8-0
15:54:53.886269 IP 127.0.0.1.58714 > 127.0.0.1.162: V2Trap(77) .1.3.6.1.2.1.1.3.0=44413 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.16 .1.3.6.1.4.1.46891.4.16=[P/A/Opaque]_9f_7a_01_00
15:54:53.889345 IP 127.0.0.1.57167 > 127.0.0.1.162: V2Trap(77) .1.3.6.1.2.1.1.3.0=44413 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.17 .1.3.6.1.4.1.46891.4.17=[P/A/Opaque]_9f_7a_01_00
15:54:53.892938 IP 127.0.0.1.51872 > 127.0.0.1.162: V2Trap(75) .1.3.6.1.2.1.1.3.0=44414 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.10 .1.3.6.1.4.1.46891.4.10=20597
15:54:53.896467 IP 127.0.0.1.43259 > 127.0.0.1.162: V2Trap(77) .1.3.6.1.2.1.1.3.0=44414 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.1.1 .1.3.6.1.4.1.46891.1.1=12569119
15:54:53.899780 IP 127.0.0.1.50526 > 127.0.0.1.162: V2Trap(76) .1.3.6.1.2.1.1.3.0=44414 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.12 .1.3.6.1.4.1.46891.4.12=3670016
15:54:53.903056 IP 127.0.0.1.43678 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44415 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.13 .1.3.6.1.4.1.46891.4.13-0
15:54:53.906321 IP 127.0.0.1.46880 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44415 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.14 .1.3.6.1.4.1.46891.4.14-0
15:54:53.909609 IP 127.0.0.1.48572 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44415 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.15 .1.3.6.1.4.1.46891.4.15-0
15:54:53.912867 IP 127.0.0.1.50828 > 127.0.0.1.162: V2Trap(77) .1.3.6.1.2.1.1.3.0=44416 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.18 .1.3.6.1.4.1.46891.4.18=[P/A/Opaque]_9f_7a_01_01
15:54:53.916098 IP 127.0.0.1.53834 > 127.0.0.1.162: V2Trap(74) .1.3.6.1.2.1.1.3.0=44416 .1.3.6.1.6.3.1.1.4.1.0=.1.3.6.1.4.1.46891.4.26 .1.3.6.1.4.1.46891.4.26-0
```

Troubleshooting

Logfiles are found /mnt/store0/log/monitor\_alert.cur

Verbose mode above can be set to "true" to allow additional logging.

## SNMP Trap

## FW: 7611+

FMADIO Devices can send SNMP Traps based on the alert triggers described above. This may be preferable to email alerts for infrastructure management.

## SNMP MIB

Latest MIB file is found (last updated 2021/12/25)



public/snmp at master · fmadio/public  
GitHub

FMADIO GITHUB Public Repo

## Config

The general configuration file is used for config

```
/opt/fmadio/etc/time.lua
```

Please edit the section titles ["SNMP"] as follows

```
["SNMP"] =  
{  
  ["Enable"]      = false,  
  ["Trap"]        = true,  
  ["Broadcast"]   = false,  
  ["BroadcastPeriod"] = 60e9,  
  ["Verbose"]     = false,  
  ["Target"]      = "127.0.0.1",  
  ["ComName"]     = "public",  
},
```

The above config enables SNMP TRAP mode only, SNMP Broadcast mode is disabled. This configuration will only send SNMP TRAP events when a Trigger is alerted.

Please update ["Target"] = setting to the correct SNMP collector address.

## Troubleshooting

An easy way to trouble shoot traps is to set the DiskFreeStore0 threshold to a very large number. In this setup the SNMP TRAP event will be constantly generated (every 1 minute).

Logfiles are found in /mnt/store0/log/monitor\_alert.cur

# Telemetry Syslog

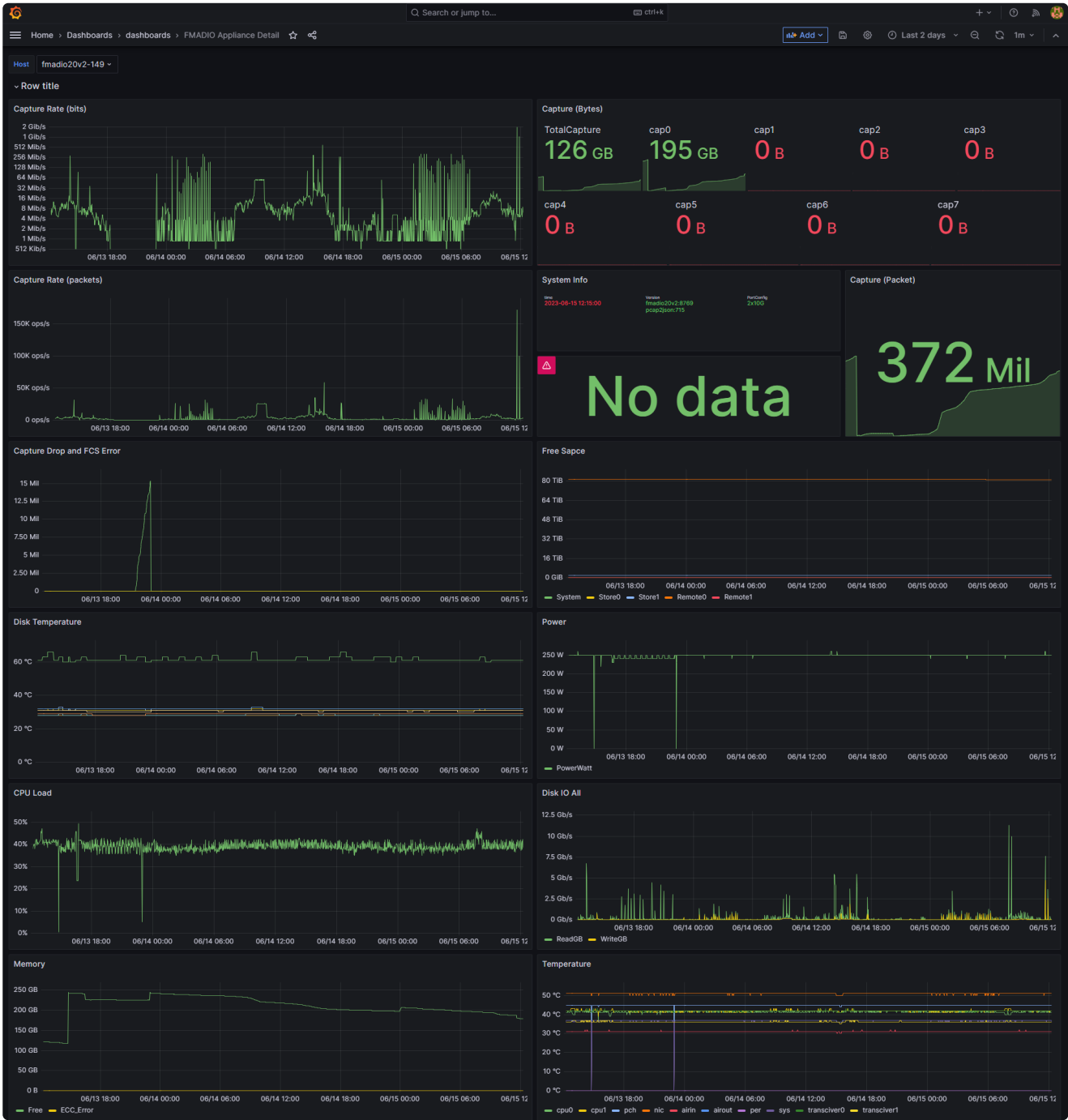
FMADIO Syslog provides rich telemetry stream for monitoring the health of the system.

Telemetry is provided in JSON format for easy ingestion in monitoring and observability infrastructure. Each syslog monitoring log event is described below, these are categorized into subsystems as follows

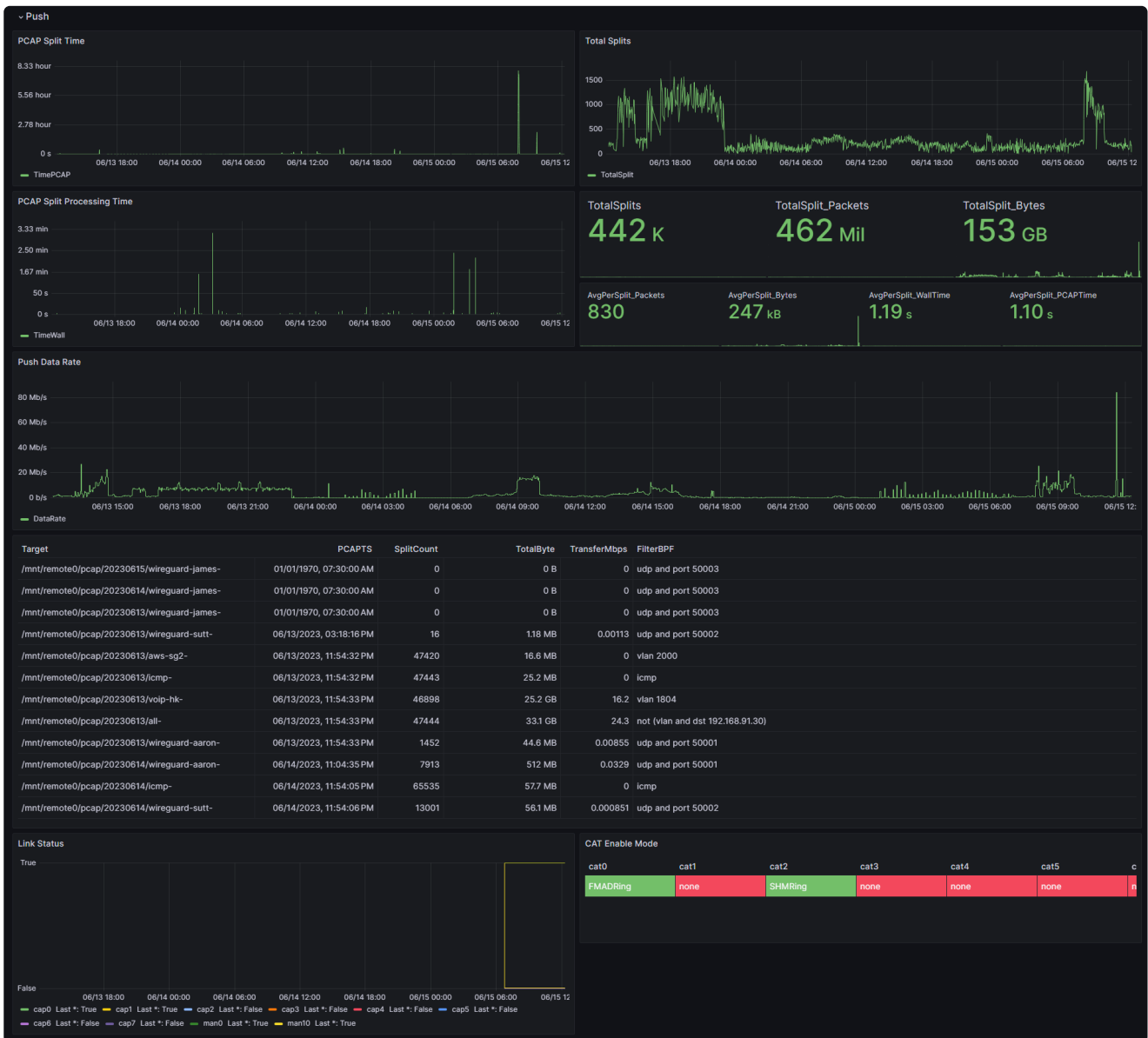
- Generic SYSLOG events
- FMADIO Temperature (Physical Thermal monitoring)
- FMADIO Time (NTP, PTPv2, PPS)
- FMADIO Other (uncategorized sensors)
- FMADIO Power (status of power supply and consumption)
- FMADIO Capture (monitoring state of the Packet Capture )
- FMADIO IO (metrics related to Disk IO)
- FMADIO Link (capture and management link status)
- FMADIO Fan (physical status of FAN and cooling)
- FMADIO Disk (status of both capture and OS disks)
- FMADIO Cat (status of the stream\_cat process used for extracting data off the disks)
- FMADIO Push PCAP (status of the PCAP Push process)
- FMADIO Push PCAP Split (event indicating each PCAP split completion)
- FMADIO Alert (Custom onbox system alerts)

## FMADIO Telemetry Service

FMADIO Telemetry service is included without charge on all support contracts. It provides a dashboard for each system, example screen shot shown below.







## JSON Header

All FMADIO JSON messages have the following header fields

```
{
  "module": "system",
  "subsystem": "alert",
  "timestamp": 1686730481,
  "ver": "8779",
  .
  .
}
```

## module

This provides a high level granularity on what the event/status is for, these map the sections below.

## subsystem

This provides more granular view on what kind of event/status this is for

## timestamp

This is the epoch time in seconds, where the FMADIO Monitoring system generated the event. Not the rsyslog time, although under normal situations they should be the same.

## ver

This is the Firmware Version, it allows easy backwards compatibility as the system updates and improves as data is ingested into the monitoring system.

## Generic SYSLOG

This includes generic syslog messages in free form plain text events. Example shown below

```
2023.06.14-15:48:02.323374 (+08:00) | fmadio100v2-228U | authpriv.info | sudo | pam_u
2023.06.14-15:48:02.500231 (+08:00) | fmadio100v2-228U | authpriv.info | sudo | pam_u
2023.06.14-15:48:02.541424 (+08:00) | fmadio100v2-228U | authpriv.info | sudo | pam_u
2023.06.14-15:48:03.359497 (+08:00) | fmadio100v2-228U | authpriv.info | sudo | pam_u
2023.06.14-15:48:03.538123 (+08:00) | fmadio100v2-228U | authpriv.info | sudo | pam_u
2023.06.14-15:48:03.566477 (+08:00) | fmadio100v2-228U | authpriv.info | sudo | pam_u
```

## FMADIO Temperature

This provides thermal monitoring of the system to ensure its running within operating ranges.

```
2023.06.14-15:51:48.439475 (+08:00) | fmadio100v2-228U | local7.info | fmadio | {"mod
```

Example JSON Pretty



```
{
  "module": "system",
  "subsystem": "temperature",
  "timestamp": 1686729129,
  "ver": "8779",
  "Temperature_CPU0": 39,
  "Temperature_CPU1": 44,
  "Temperature_PCH": 41,
  "Temperature_SYS": 37,
  "Temperature_PER": 22,
  "Temperature_NIC": 48,
  "Temperature_AirIn": 22,
  "Temperature_AirOut": 0,
  "Temperature_Transceiver0": 39,
  "Temperature_Transceiver1": 0
}
```

## FMADIO Time

Provides monitoring of time synchronization

```
2023.06.14-15:53:23.496058 (+08:00) | fmadio100v2-228U | local17.info | fmadio | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "ptp",
  "timestamp": 1686729331,
  "ver": "8779",
  "TimeFPGA": 1686729331198241000,
  "TimeSYS": 1686729331198588000,
  "GMOffset": 0,
  "GMSync": false,
  "GMMaster": "",
  "SysOffset": 0,
  "SysSync": false,
  "iSysOffset": 0,
  "iSysSync": true,
  "NTPOffset": -445000,
  "NTPSync": true,
  "NTPMaster": "*192.168.2.5",
  "GMUpTime": 0,
  "SysUptime": 0,
  "iSysUptime": 156675,
  "PPSUptime": 156124,
  "NTPUptime": 156224,
  "PPSPeriod": 3.103035483,

  "PPSOffset": 0,
  "PPSdPhase": -4,
  "PPSdZero": 0,
  "PPSCnt": 157316,
  "Clk156Period": 0,
  "Clk156Offset": 0,
  "Clk250Period": 0,
  "Clk250Offset": 0,
  "Clk322Period": 0,
  "Clk322Offset": 0
}
```

## FMADIO Other

Miscellaneous other fields

```
2023.06.14-15:54:27.535663 (+08:00) | fmadio100v2-228U | local7.info | fmadio | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "other",
  "timestamp": 1686729394,
  "ver": "8779",
  "UptimeHour": 43.55,

  "MemFree": 366355021824,
  "MemErrorECC": 0,
  "WritebackB": 0,
  "WritebackPct": 0,
  "WritebackDropTotalG": 0,
  "WritebackDropG": 0,
  "CacheSize": 30726047137792,
  "StoreSize": 30725994708992,
  "CPULoad": 0.39,
  "SerialNo": "undef-undef-e0d55e5d2150",
  "PortConfig": "2x100G",
  "Version": "fmadio100v2:8779 pcap2json:715"
}
```

## FMADIO Power

Provides power status and utilization information

```
2023.06.14-15:53:23.488485 (+08:00) | fmadio100v2-228U | local17.info | fmadio | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "power",
  "timestamp": 1686729447,
  "ver": "8779",
  "PSU0_Status": false,
  "PSU1_Status": true,
  "PSU_PowerWatt": 270
}
```

## FMADIO Capture

Provides status information around the capture process

```
2023.06.14-15:56:34.457781 (+08:00) | fmadio100v2-228U | local17.info | fmadio | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "capture",
  "timestamp": 1686673255,
  "ver": "8779",
  "CaptureEnb": false,
  "CapturePkt": 2795520,
  "CaptureByte": 190095360,
  "CaptureDrop": 952825443,
  "CaptureFCSError": 0,
  "CaptureRateGbps": 0,
  "CaptureRateMpps": 0,
  "CaptureName": "",
  "CapturePort0_Byte": 0,
  "CapturePort0_Pkt": 0,
  "CapturePort1_Byte": 68000000000,
  "CapturePort1_Pkt": 1000000000,
  "CapturePort2_Byte": 0,
  "CapturePort2_Pkt": 0,
  "CapturePort3_Byte": 0,
  "CapturePort3_Pkt": 0,
  "CapturePort4_Byte": 0,
  "CapturePort4_Pkt": 0,
  "CapturePort5_Byte": 0,
  "CapturePort5_Pkt": 0,
  "CapturePort6_Byte": 0,
  "CapturePort6_Pkt": 0,
  "CapturePort7_Byte": 0,
  "CapturePort7_Pkt": 0
}
```

## FMADIO IO

Provides status and performance of IO disk/network related systems

```
2023.06.14-15:54:27.528914 (+08:00) | fmadio100v2-228U | local17.info | fmadio | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "io",
  "timestamp": 1686729638,
  "ver": "8779",
  "DiskRdGbps": 0,
  "DiskWrGbps": 0
}
```

## FMADIO Link

Status information around port link status

```
{ "module": "system", "subsystem": "link" , "timestamp": 1686729256, "ver": "8779", "cap0_link"
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "link",
  "timestamp": 1686729744,
  "ver": "8779",
  "cap0_link": true,
  "cap1_link": true,
  "cap2_link": false,
  "cap3_link": false,
  "cap4_link": false,
  "cap5_link": false,
  "cap6_link": false,
  "cap7_link": false,
  "man0_link": true,
  "man10_link": true
}
```

## FMADIO Fan

Status information around fans and cooling

```
2023.06.14-15:53:02.396976 (+08:00) | fmadio100v2-228U | local17.info | fmadio | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "fan",
  "timestamp": 1686729819,
  "ver": "8779",
  "Fan_SYS0": 21600,
  "Fan_SYS1": 21450,
  "Fan_SYS2": 21450,
  "Fan_SYS3": 21450,
  "Fan_SYS4": 21450,
  "Fan_SYS5": 21450,
  "Fan_SYS6": 21750,
  "Fan_SYS7": 21600
}
```

## FMADIO Disk

Status around the capture and os disk

```
2023.06.14-15:59:56.186455 (+08:00) | fmadio100v2-228U | local7.info | fmadio | {"mod
```

Pretty JSON

```

{
  "module": "system",
  "subsystem": "disk",
  "timestamp": 1686729925,
  "ver": "8779",
  "FreeGB_System": 6.078,
  "FreeGB_Store0": 5528.059,
  "FreeGB_Store1": 0,
  "FreeGB_Remote0": 83294.126,
  "FreeGB_Remote1": 83294.126,
  "DiskPresent_os0": true,
  "DiskTemperature_os0": 37,
  "DiskSMART_os0": 0,
  "DiskPresent_ssd0": true,
  "DiskTemperature_ssd0": 31,
  "DiskSMART_ssd0": 0,
  "DiskPresent_ssd1": true,
  "DiskTemperature_ssd1": 29,
  "DiskSMART_ssd1": 0,
  "DiskPresent_ssd2": true,
  "DiskTemperature_ssd2": 30,
  "DiskSMART_ssd2": 0,
  "DiskPresent_ssd3": true,
  "DiskTemperature_ssd3": 29,
  "DiskSMART_ssd3": 0,
  "DiskPresent_ssd4": true,
  "DiskTemperature_ssd4": 30,
  "DiskSMART_ssd4": 0,
  "DiskPresent_ssd5": true,
  "DiskTemperature_ssd5": 31,
  "DiskSMART_ssd5": 0,
  "DiskPresent_ssd6": true,
  "DiskTemperature_ssd6": 31,
  "DiskSMART_ssd6": 0,
  "DiskPresent_ssd7": true,
  "DiskTemperature_ssd7": 32,
  "DiskSMART_ssd7": 0,
  "DiskPresent_par0": true,
  "DiskTemperature_par0": 30,
  "DiskSMART_par0": 0
}

```

## FMADIO Cat

stream\_cat is the core FMADIO utility for extracting packets off the storage system. Its used heavily for download, realtime processing and troubleshooting. This provides statistics and state of the currently running stream\_cat instances.

Up to 8 simultaneousle stream\_cat instances can be run at the same time. cat\_\*\_ is per instance.

2023.06.14-16:06:28.566556 (+08:00) | fmadio100v2-228U | local7.info | fmadio | {"mod

Pretty JSON



```
{
  "module": "system",
  "subsystem": "cat",
  "timestamp": 1686730073,
  "ver": "8779",
  "cat_0_Enable": false,
  "cat_0_Mode": "",
  "cat_0_CPUMain": 0,
  "cat_0_TSPCAP": 0,
  "cat_0_ReadPkt": 0,
  "cat_0_ReadByte": 0,
  "cat_0_ReadTotalPkt": 0,
  "cat_0_ReadTotalByte": 0,
  "cat_0_ReadGbps": 0,
  "cat_0_ReadMpps": 0,
  "cat_0_WritePkt": 0,
  "cat_0_WriteByte": 0,
  "cat_0_WriteTotalPkt": 0,
  "cat_0_WriteTotalByte": 0,
  "cat_0_WriteGbps": 0,
  "cat_0_WriteMpps": 0,
  "cat_0_PendingByte": 0,
  "cat_0_PktDiscard": 0,
  "cat_0_PktDiscardTotal": 0,
  "cat_0_PktSlice": 0,
  "cat_0_IOPriority": 0,
  "cat_0_ChunkID": 0,
  "cat_0_CmdLine": "",
  "cat_0_StreamName": "",
  "cat_0_FilterBPF": "",
  "cat_0_CPUIidle": 0,
  "cat_0_CPUFetch": 0,
  "cat_0_CPUProcess": 0,
  "cat_0_CPUSend": 0,

  "cat_1_Enable": false,
  "cat_1_Mode": "",
  "cat_1_CPUMain": 0,
  "cat_1_TSPCAP": 0,
  "cat_1_ReadPkt": 0,
  "cat_1_ReadByte": 0,
  "cat_1_ReadTotalPkt": 0,
  "cat_1_ReadTotalByte": 0,
  "cat_1_ReadGbps": 0,
  "cat_1_ReadMpps": 0,
  "cat_1_WritePkt": 0,
  "cat_1_WriteByte": 0,
  "cat_1_WriteTotalPkt": 0,
  "cat_1_WriteTotalByte": 0,
  "cat_1_WriteGbps": 0,
  "cat_1_WriteMpps": 0,
  "cat_1_PendingByte": 0,
```

```
"cat_1_PktDiscard": 0,
"cat_1_PktDiscardTotal": 0,
"cat_1_PktSlice": 0,
"cat_1_IOPriority": 0,
"cat_1_ChunkID": 0,
"cat_1_CmdLine": "",
"cat_1_StreamName": "",
"cat_1_FilterBPF": "",
"cat_1_CPUIidle": 0,
"cat_1_CPUFetch": 0,
"cat_1_CPUProcess": 0,
"cat_1_CPUSend": 0,

"cat_2_Enable": false,
"cat_2_Mode": "",
"cat_2_CPUMain": 0,
"cat_2_TSPCAP": 0,
"cat_2_ReadPkt": 0,
"cat_2_ReadByte": 0,
"cat_2_ReadTotalPkt": 0,
"cat_2_ReadTotalByte": 0,
"cat_2_ReadGbps": 0,
"cat_2_ReadMpps": 0,
"cat_2_WritePkt": 0,
"cat_2_WriteByte": 0,
"cat_2_WriteTotalPkt": 0,
"cat_2_WriteTotalByte": 0,
"cat_2_WriteGbps": 0,
"cat_2_WriteMpps": 0,
"cat_2_PendingByte": 0,
"cat_2_PktDiscard": 0,
"cat_2_PktDiscardTotal": 0,
"cat_2_PktSlice": 0,
"cat_2_IOPriority": 0,
"cat_2_ChunkID": 0,
"cat_2_CmdLine": "",
"cat_2_StreamName": "",
"cat_2_FilterBPF": "",
"cat_2_CPUIidle": 0,
"cat_2_CPUFetch": 0,
"cat_2_CPUProcess": 0,
"cat_2_CPUSend": 0,

"cat_3_Enable": false,
"cat_3_Mode": "",
"cat_3_CPUMain": 0,
"cat_3_TSPCAP": 0,
"cat_3_ReadPkt": 0,
"cat_3_ReadByte": 0,
"cat_3_ReadTotalPkt": 0,
"cat_3_ReadTotalByte": 0,
"cat_3_ReadGbps": 0,
```

```
"cat_3_ReadPkt": 0,  
"cat_3_WriteByte": 0,  
"cat_3_WriteTotalPkt": 0,  
"cat_3_WriteTotalByte": 0,  
"cat_3_WriteGbps": 0,  
"cat_3_WriteMpps": 0,  
"cat_3_PendingByte": 0,  
"cat_3_PktDiscard": 0,  
"cat_3_PktDiscardTotal": 0,  
"cat_3_PktSlice": 0,  
"cat_3_IOPriority": 0,  
"cat_3_ChunkID": 0,  
"cat_3_CmdLine": "",  
"cat_3_StreamName": "",  
"cat_3_FilterBPF": "",  
"cat_3_CPUIidle": 0,  
"cat_3_CPUFetch": 0,  
"cat_3_CPUProcess": 0,  
"cat_3_CPUSend": 0,  
  
"cat_4_Enable": false,  
"cat_4_Mode": "",  
"cat_4_CPUMain": 0,  
"cat_4_TSPCAP": 0,  
"cat_4_ReadPkt": 0,  
"cat_4_ReadByte": 0,  
"cat_4_ReadTotalPkt": 0,  
"cat_4_ReadTotalByte": 0,  
"cat_4_ReadGbps": 0,  
"cat_4_ReadMpps": 0,  
"cat_4_WritePkt": 0,  
"cat_4_WriteByte": 0,  
"cat_4_WriteTotalPkt": 0,  
"cat_4_WriteTotalByte": 0,  
"cat_4_WriteGbps": 0,  
"cat_4_WriteMpps": 0,  
"cat_4_PendingByte": 0,  
"cat_4_PktDiscard": 0,  
"cat_4_PktDiscardTotal": 0,  
"cat_4_PktSlice": 0,  
"cat_4_IOPriority": 0,  
"cat_4_ChunkID": 0,  
"cat_4_CmdLine": "",  
"cat_4_StreamName": "",  
"cat_4_FilterBPF": "",  
"cat_4_CPUIidle": 0,  
"cat_4_CPUFetch": 0,  
"cat_4_CPUProcess": 0,  
"cat_4_CPUSend": 0,  
  
"cat_5_Enable": false,  
"cat_5_Mode": "",
```

```
"cat_5_CPUMain": 0,
"cat_5_TSPCAP": 0,
"cat_5_ReadPkt": 0,
"cat_5_ReadByte": 0,
"cat_5_ReadTotalPkt": 0,
"cat_5_ReadTotalByte": 0,
"cat_5_ReadGbps": 0,
"cat_5_ReadMpps": 0,
"cat_5_WritePkt": 0,
"cat_5_WriteByte": 0,
"cat_5_WriteTotalPkt": 0,
"cat_5_WriteTotalByte": 0,
"cat_5_WriteGbps": 0,
"cat_5_WriteMpps": 0,
"cat_5_PendingByte": 0,
"cat_5_PktDiscard": 0,
"cat_5_PktDiscardTotal": 0,
"cat_5_PktSlice": 0,
"cat_5_IOPriority": 0,
"cat_5_ChunkID": 0,
"cat_5_CmdLine": "",
"cat_5_StreamName": "",
"cat_5_FilterBPF": "",
"cat_5_CPUIidle": 0,
"cat_5_CPUFetch": 0,
"cat_5_CPUProcess": 0,
"cat_5_CPUSend": 0,

"cat_6_Enable": false,
"cat_6_Mode": "",
"cat_6_CPUMain": 0,
"cat_6_TSPCAP": 0,
"cat_6_ReadPkt": 0,
"cat_6_ReadByte": 0,
"cat_6_ReadTotalPkt": 0,
"cat_6_ReadTotalByte": 0,
"cat_6_ReadGbps": 0,
"cat_6_ReadMpps": 0,
"cat_6_WritePkt": 0,
"cat_6_WriteByte": 0,
"cat_6_WriteTotalPkt": 0,
"cat_6_WriteTotalByte": 0,
"cat_6_WriteGbps": 0,
"cat_6_WriteMpps": 0,
"cat_6_PendingByte": 0,
"cat_6_PktDiscard": 0,
"cat_6_PktDiscardTotal": 0,
"cat_6_PktSlice": 0,
"cat_6_IOPriority": 0,

"cat_6_ChunkID": 0,
"cat_6_CmdLine": "",
"cat_6_StreamName": "",
"cat_6_FilterBPF": "",
```

```
"cat_6_CPUIdle": 0,  
"cat_6_CPUFetch": 0,  
"cat_6_CPUProcess": 0,  
"cat_6_CPUSend": 0,  
  
"cat_7_Enable": false,  
"cat_7_Mode": "",  
"cat_7_CPUMain": 0,  
"cat_7_TSPCAP": 0,  
"cat_7_ReadPkt": 0,  
"cat_7_ReadByte": 0,  
"cat_7_ReadTotalPkt": 0,  
"cat_7_ReadTotalByte": 0,  
"cat_7_ReadGbps": 0,  
"cat_7_ReadMpps": 0,  
"cat_7_WritePkt": 0,  
"cat_7_WriteByte": 0,  
"cat_7_WriteTotalPkt": 0,  
"cat_7_WriteTotalByte": 0,  
"cat_7_WriteGbps": 0,  
"cat_7_WriteMpps": 0,  
"cat_7_PendingByte": 0,  
"cat_7_PktDiscard": 0,  
"cat_7_PktDiscardTotal": 0,  
"cat_7_PktSlice": 0,  
"cat_7_IOPriority": 0,  
"cat_7_ChunkID": 0,  
"cat_7_CmdLine": "",  
"cat_7_StreamName": "",  
"cat_7_FilterBPF": "",  
"cat_7_CPUIdle": 0,  
"cat_7_CPUFetch": 0,  
"cat_7_CPUProcess": 0,  
"cat_7_CPUSend": 0,  
  
"cat_EnableCnt": 0,  
"cat_ReadPkt": 0,  
"cat_ReadByte": 0,  
"cat_ReadTotalPkt": 0,  
"cat_ReadTotalByte": 0,  
"cat_ReadGbps": 0,  
"cat_ReadMpps": 0,  
"cat_WritePkt": 0,  
"cat_WriteByte": 0,  
"cat_WriteTotalPkt": 0,  
"cat_WriteTotalByte": 0,  
"cat_WriteGbps": 0,  
"cat_WriteMpps": 0  
}
```

Provides status information of the currently active Push PCAP proceses

```
2023.06.14-16:11:44.342725 (+08:00) | fmadiao20v2-149 | local7.info | fmadiao | {"modul
```

Pretty JSON

```
{
  "module": "push_pcap",
  "subsystem": "status",
  "timestamp": 1686730363,
  "ver": "8769",
  "Process": "voip-hk",
  "IsUp": true,
  "Splits": 57343,
  "TotalByte": 63955460513,
  "TotalPkt": 228548682,
  "TransferMbps": 12,
  "PCAPTS": 1686730357291535600,
  "FilterBPF": "vlan 1804",
  "FilterFrame": "nil",
  "Target": "/mnt/remote0/pcap/20230614/voip-hk-"
}
```

## FMADIO Push PCAP File

Called after completing a PCAP file split event. Helpful to monitoring each and every PCAP split leaving the system

```
2023.06.14-16:14:43.131050 (+08:00) | fmadiao20v2-149 | user.notice | fmadiao | {"modul
```

Pretty JSON

```
{
  "module": "push_pcap",
  "subsystem": "split",
  "timestamp": 1686730521,
  "ver": "8769",
  "filename": "/mnt/remote0/pcap/20230614/icmp-20230614_161514.pcap.zstd",
  "Byte": 896,
  "Pkt": 8,
  "TimeWall": 0.930522,
  "TimePCAP": 0.988352,
  "PCATSStart": 1686730515000000000,
  "PCAPTEnd": 1686730514909767700
}
```

## FMADIO Alert

The system has capability to generate system alerts based on configuration files. These alerts can be sent to SYSLOG for ingestion by a monitoring system

```
2023.06.14-16:17:59.032528 (+08:00) | fmadiao100v2-228U | local17.info | fmadiao | {"mod
```

Pretty JSON

```
{
  "module": "system",
  "subsystem": "alert",
  "timestamp": 1686730679,
  "ver": "8779",
  "Subject": "CPU Temperature",
  "Message": "CPU Temperature Over CPU0:39 CPU1:43 > 20"
}
```

## Telemetry Service Setup

Setting up automatic telemetry as follows

### Step 1) Generate unique SSH key

FMADIO devices by default have a pre-installed ssh key. To correctly secure and uniquely identify the system generate your own SSH key as follows.

```
ssh-keygen -t rsa -b 3072
```

Using a password less key ensures the automatic setup requires no manual intervention.

Example output per below

```

fmadio@fmadio100p3-539:~$ ssh-keygen -t rsa -b 3072
Generating public/private rsa key pair.
Enter file in which to save the key (/home/fmadio/.ssh/id_rsa):
/home/fmadio/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/fmadio/.ssh/id_rsa
Your public key has been saved in /home/fmadio/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:G/***** fmadio@fmadio100p3-539
The key's randomart image is:
+---[RSA 3072]-----+

| Eo.oo=Bo . |
| o .o.. o.. + |
| . ..+ +.o o o |
|. . =..+ . ..|
| . +S o o |
| . ..+ o . |
| ..+.oo. o . |
| o. o= = o . |
| oo..o . .oo |
+----[SHA256]-----+
fmadio@fmadio100p3-539:~$

```

Send the above public key (below) to support@fmad.io

```
/opt/fmadio/etc/fmadio_id_rsa.pub
```

## Step 2) Copy to persistent Storage

The SSH public/private keys are on the volatile file system. Copy the keys to the persistent storage.

```
cp .ssh/id_rsa /opt/fmadio/etc/fmadio_id_rsa
```

```
cp .ssh/id_rsa.pub /opt/fmadio/etc/fmadio_id_rsa.pub
```

NOTE: the key is renamed with an fmadio\_\* prefix. The system copies the keys from this location and renames them in the .ssh/id\_rsa .ssh/id\_rsa.pub directory during the boot process.

## Step 3) Copy the reference boot script

There is a reference boot script located in



```
/opt/fmadio/etc_ro/boot.lua.telemetry_mon2
```

Copy this to to the /opt/fmadio/etc/boot.lua file to automatically establish ssh tunnel to the telemetry service.

```
cp /opt/fmadio/etc_ro/boot.lua.telemetry_mon2 /opt/fmadio/etc/boot.lua
```

After copying replace the "username" to the username provided by fmadio support and save the file.

```
fmadio@fmadio100p3-539:~$ cat /opt/fmadio/etc/boot.lua
local Cmd = "/usr/bin/autossh -f -N 50030 -N -o StrictHostKeyChecking=false -i /mnt/store0/etc/fmadio_id_rsa [redacted] usernames [redacted]"
print(Cmd)
os.execute(Cmd)
```

#### Step 4) Copy the reference rsyslog config

In addition to ssh tunnel setup, rsyslog configuration to forward syslog messages to the SSH tunnel.

Copy the reference config to /opt/fmadio/etc/ directory as follows

```
cp /opt/fmadio/etc_ro/syslogd.conf.tunnel /opt/fmadio/etc/syslogd.conf
```

No modifications are required.

#### Step 5) Reboot the system

Reboot the system to check all the above steps are executed correctly

#### Step 6) Validate

After rebooting log into the Grafana monitoring site with the assigned username and confirm data is being received.

Any problems please contact support@fmad.io

### Telemetry via API

In addition to syslog, the systems telemetry information snapshot can be fetched using the JSON API.

Example command:

```
curl -s http://127.0.0.1/api/v1/system/status | jq
```

With the pretty formatted output as follows. This output will always match the above syslog information.

```
{
  "timestamp": 1695785514,
  "ver": "9120",
  "temperature": {
    "module": "system",
    "subsystem": "temperature",
    "timestamp": 1695785514,
    "ver": "9120",
    "Temperature_CPU0": 55,
    "Temperature_CPU1": 70,
    "Temperature_PCH": 46,
    "Temperature_SYS": 42,
    "Temperature_PER": 23,
    "Temperature_NIC": 49,
    "Temperature_AirIn": 23,
    "Temperature_AirOut": 0,
    "Temperature_Transceiver0": 42,
    "Temperature_Transceiver1": 42
  },
  "fan": {
    "module": "system",
    "subsystem": "fan",
    "timestamp": 1695785514,
    "ver": "9120",
    "Fan_SYS0": 21450,
    "Fan_SYS1": 21450,
    "Fan_SYS2": 21300,
    "Fan_SYS3": 21450,
    "Fan_SYS4": 21450,
    "Fan_SYS5": 21450,
    "Fan_SYS6": 21450,
    "Fan_SYS7": 21600
  },
  "disk": {
    "module": "system",
    "subsystem": "disk",
    "timestamp": 1695785514,
    "ver": "9120",
    "FreeGB_System": 8.977,
    "FreeGB_Store0": 4720.779,
    "FreeGB_Store1": 0,
    "FreeGB_Remote0": 46349.53,
    "FreeGB_Remote1": 46349.53,
    "DiskPresent_os0": true,
    "DiskTemperature_os0": 40,
    "DiskSMART_os0": 0,
    "DiskPresent_ssd0": true,
    "DiskTemperature_ssd0": 36,
    "DiskSMART_ssd0": 0,
    "DiskPresent_ssd1": true,
    "DiskTemperature_ssd1": 34,
    "DiskSMART_ssd1": 0,
  }
}
```

```
"DiskPresent_ssd2": true,
"DiskTemperature_ssd2": 35,
"DiskSMART_ssd2": 0,
"DiskPresent_ssd3": true,
"DiskTemperature_ssd3": 34,
"DiskSMART_ssd3": 0,
"DiskPresent_ssd4": true,
"DiskTemperature_ssd4": 35,
"DiskSMART_ssd4": 0,
"DiskPresent_ssd5": true,
"DiskTemperature_ssd5": 40,
"DiskSMART_ssd5": 0,
"DiskPresent_ssd6": true,
"DiskTemperature_ssd6": 36,
"DiskSMART_ssd6": 0,
"DiskPresent_ssd7": true,
"DiskTemperature_ssd7": 37,
"DiskSMART_ssd7": 0,
"DiskPresent_par0": true,
"DiskTemperature_par0": 34,
"DiskSMART_par0": 0
},
"link": {
  "module": "system",
  "subsystem": "link",
  "timestamp": 1695785514,
  "ver": "9120",
  "cap0_link": true,
  "cap1_link": true,
  "cap2_link": true,
  "cap3_link": true,
  "cap4_link": true,
  "cap5_link": true,
  "cap6_link": true,
  "cap7_link": true,
  "man0_link": true,
  "man10_link": true
},
"io": {
  "module": "system",
  "subsystem": "io",
  "timestamp": 1695785514,
  "ver": "9120",
  "DiskRdGbps": 0.46,
  "DiskWrGbps": 0.19
},
"capture": {
  "module": "system",
  "subsystem": "capture",
  "timestamp": 1695785514,
  "ver": "9120",
  "CaptureEnb": true,
  "CapturePkt": 11751922,
```

```
"CaptureByte": 16465381240,
"CaptureDrop": 0,
"CaptureFCSError": 0,
"CaptureRateGbps": 0.214141,
"CaptureRateMpps": 0.0188,
"CaptureName": "wan_colo0_20230927_0320",
"CapturePort0_Byte": 0,
"CapturePort0_Pkt": 0,
"CapturePort1_Byte": 16482394142,
"CapturePort1_Pkt": 11763917,
"CapturePort2_Byte": 0,
"CapturePort2_Pkt": 0,
"CapturePort3_Byte": 0,
"CapturePort3_Pkt": 0,
"CapturePort4_Byte": 0,
"CapturePort4_Pkt": 0,
"CapturePort5_Byte": 0,
"CapturePort5_Pkt": 0,
"CapturePort6_Byte": 0,
"CapturePort6_Pkt": 0,
"CapturePort7_Byte": 0,
"CapturePort7_Pkt": 0
},
"power": {
  "module": "system",
  "subsystem": "power",
  "timestamp": 1695785514,
  "ver": "9120",
  "PSU0_Status": false,
  "PSU1_Status": true,
  "PSU_PowerWatt": 370
},
"other": {
  "module": "system",
  "subsystem": "other",
  "timestamp": 1695785514,
  "ver": "9120",
  "UptimeHour": 0.34,
  "MemFree": 350465687552,
  "MemErrorECC": 0,
  "MemCached": 19210211328,
  "MemMapped": 5480173568,
  "MemBuffer": 57253888,
  "MemDirty": 0,
  "PageInByte": 0,
  "FDCnt": 1188,
  "WritebackB": 0,
  "WritebackPct": 0,
  "WritebackDropTotalG": 0,
  "WritebackDropG": 0,
  "CacheSize": 30726047137792,
  "StoreSize": 30725994708992,
  "CPULoad": 5.1,
```

```
"SerialNo": "undef-undef-e0d55e5d2150",
"PortConfig": "8x10G",
"Version": "fmadio100v2:9120pcap2json:715"
},
"cat": {
  "module": "system",
  "subsystem": "cat",
  "timestamp": 1695785514,
  "ver": "9120",
  "cat_0_Enable": true,
  "cat_0_Mode": "FMADRing",
  "cat_0_CPUMain": 0,
  "cat_0_TSPCAP": 1695785512,
  "cat_0_ReadPkt": 20194,
  "cat_0_ReadByte": 28774880,
  "cat_0_ReadTotalPkt": 11727311,
  "cat_0_ReadTotalByte": 16508052240,
  "cat_0_ReadGbps": 0.221358,
  "cat_0_ReadMpps": 0.019418,
  "cat_0_WritePkt": 40388,
  "cat_0_WriteByte": 57412864,
  "cat_0_WriteTotalPkt": 11727311,
  "cat_0_WriteTotalByte": 16508052240,
  "cat_0_WriteGbps": 0.441662,
  "cat_0_WriteMpps": 0.038837,
  "cat_0_PendingByte": 30932992,
  "cat_0_PktDiscard": 0,
  "cat_0_PktDiscardTotal": 0,
  "cat_0_PktSlice": 0,
  "cat_0_IOPriority": 20,
  "cat_0_ChunkID": 11003385,
  "cat_0_CmdLine": "/opt/fmadio/bin/stream_cat--uidpush_pcap_1695784865262465024--follow-s",
  "cat_0_StreamName": "wan_colo0_20230927_0320",
  "cat_0_FilterBPF": "",
  "cat_0_CPUIdle": 0.9769,
  "cat_0_CPUFetch": 0.0148,
  "cat_0_CUPProcess": 0.0148,
  "cat_0_CPUSend": 0,
  "cat_1_Enable": true,
  "cat_1_Mode": "FMADRing",
  "cat_1_CPUMain": 0,
  "cat_1_TSPCAP": 1695785504,
  "cat_1_ReadPkt": 189594,
  "cat_1_ReadByte": 271169120,
  "cat_1_ReadTotalPkt": 10854873,
  "cat_1_ReadTotalByte": 15340528240,
  "cat_1_ReadGbps": 0.215211,
  "cat_1_ReadMpps": 0.018809,
  "cat_1_WritePkt": 189594,
  "cat_1_WriteByte": 269871369,
  "cat_1_WriteTotalPkt": 10854873,
  "cat_1_WriteTotalByte": 15340528240,
  "cat_1_WriteGbps": 0.214182,
```

```
"cat_1_WriteMpps": 0.018809,
"cat_1_PendingByte": 31719424,
"cat_1_PktDiscard": 0,
"cat_1_PktDiscardTotal": 0,
"cat_1_PktSlice": 0,
"cat_1_IOPriority": 20,
"cat_1_ChunkID": 11002639,
"cat_1_CmdLine": "/opt/fmadio/bin/stream_cat--uidpush_lxc_1695784924583593984--follow--r
"cat_1_StreamName": "wan_colo0_20230927_0320",
"cat_1_FilterBPF": "",
"cat_1_CPUIIdle": 0.9928,
"cat_1_CPUFetch": 0.0064,
"cat_1_CPUProcess": 0.0064,
"cat_1_CPUSend": 0,
"cat_2_Enable": false,
"cat_2_Mode": "",
"cat_2_CPUMain": 0,
"cat_2_TSPCAP": 0,
"cat_2_ReadPkt": 0,
"cat_2_ReadByte": 0,
"cat_2_ReadTotalPkt": 0,
"cat_2_ReadTotalByte": 0,
"cat_2_ReadGbps": 0,
"cat_2_ReadMpps": 0,
"cat_2_WritePkt": 0,
"cat_2_WriteByte": 0,
"cat_2_WriteTotalPkt": 0,
"cat_2_WriteTotalByte": 0,
"cat_2_WriteGbps": 0,
"cat_2_WriteMpps": 0,
"cat_2_PendingByte": 0,
"cat_2_PktDiscard": 0,
"cat_2_PktDiscardTotal": 0,
"cat_2_PktSlice": 0,
"cat_2_IOPriority": 0,
"cat_2_ChunkID": 0,
"cat_2_CmdLine": "",
"cat_2_StreamName": "",
"cat_2_FilterBPF": "",
"cat_2_CPUIIdle": 0,
"cat_2_CPUFetch": 0,
"cat_2_CPUProcess": 0,
"cat_2_CPUSend": 0,
"cat_3_Enable": false,
"cat_3_Mode": "",
"cat_3_CPUMain": 0,
"cat_3_TSPCAP": 0,
"cat_3_ReadPkt": 0,
"cat_3_ReadByte": 0,
"cat_3_ReadTotalPkt": 0,
"cat_3_ReadTotalByte": 0,
"cat_3_ReadGbps": 0,
"cat_3_ReadMpps": 0,
```

```
"cat_3_WritePkt": 0,  
"cat_3_WriteByte": 0,  
"cat_3_WriteTotalPkt": 0,  
"cat_3_WriteTotalByte": 0,  
"cat_3_WriteGbps": 0,  
"cat_3_WriteMpps": 0,  
"cat_3_PendingByte": 0,  
"cat_3_PktDiscard": 0,  
"cat_3_PktDiscardTotal": 0,  
"cat_3_PktSlice": 0,  
"cat_3_IOPriority": 0,  
"cat_3_ChunkID": 0,  
"cat_3_CmdLine": "",  
"cat_3_StreamName": "",  
"cat_3_FilterBPF": "",  
"cat_3_CPUIidle": 0,  
"cat_3_CPUFetch": 0,  
"cat_3_CPUProcess": 0,  
"cat_3_CPUSend": 0,  
"cat_4_Mode": "",  
"cat_4_CPUMain": 0,  
"cat_4_TSPCAP": 0,  
"cat_4_ReadPkt": 0,  
"cat_4_ReadByte": 0,  
"cat_4_ReadTotalPkt": 0,  
"cat_4_ReadTotalByte": 0,  
"cat_4_ReadGbps": 0,  
"cat_4_ReadMpps": 0,  
"cat_4_WritePkt": 0,  
"cat_4_WriteByte": 0,  
"cat_4_WriteTotalPkt": 0,  
"cat_4_WriteTotalByte": 0,  
"cat_4_WriteGbps": 0,  
"cat_4_WriteMpps": 0,  
"cat_4_PendingByte": 0,  
"cat_4_PktDiscard": 0,  
"cat_4_PktDiscardTotal": 0,  
"cat_4_PktSlice": 0,  
"cat_4_IOPriority": 0,  
"cat_4_ChunkID": 0,  
"cat_4_CmdLine": "",  
"cat_4_StreamName": "",  
"cat_4_FilterBPF": "",  
"cat_4_CPUIidle": 0,  
"cat_4_CPUFetch": 0,  
"cat_4_CPUProcess": 0,  
"cat_4_CPUSend": 0,  
"cat_5_Enable": false,  
"cat_5_Mode": "",  
"cat_5_CPUMain": 0,  
"cat_5_TSPCAP": 0,  
"cat_5_ReadPkt": 0,  
"cat_5_ReadByte": 0,
```

```
"cat_5_ReadTotalBkt": 0,
"cat_5_ReadGbps": 0,
"cat_5_ReadMpps": 0,
"cat_5_WritePkt": 0,
"cat_5_WriteByte": 0,
"cat_5_WriteTotalPkt": 0,
"cat_5_WriteTotalByte": 0,
"cat_5_WriteGbps": 0,
"cat_5_WriteMpps": 0,
"cat_5_PendingByte": 0,
"cat_5_PktDiscard": 0,
"cat_5_PktDiscardTotal": 0,
"cat_5_PktSlice": 0,
"cat_5_IOPriority": 0,
"cat_5_ChunkID": 0,
"cat_5_CmdLine": "",
"cat_5_StreamName": "",
"cat_5_FilterBPF": "",
"cat_5_CPUIidle": 0,
"cat_5_CPUFetch": 0,
"cat_5_CUPProcess": 0,
"cat_5_CPUSend": 0,
"cat_6_Enable": false,
"cat_6_Mode": "",
"cat_6_CPUMain": 0,
"cat_6_TSPCAP": 0,
"cat_6_ReadPkt": 0,
"cat_6_ReadByte": 0,
"cat_6_ReadTotalPkt": 0,
"cat_6_ReadTotalByte": 0,
"cat_6_ReadGbps": 0,
"cat_6_ReadMpps": 0,
"cat_6_WritePkt": 0,
"cat_6_WriteByte": 0,
"cat_6_WriteTotalPkt": 0,
"cat_6_WriteTotalByte": 0,
"cat_6_WriteGbps": 0,
"cat_6_WriteMpps": 0,
"cat_6_PendingByte": 0,
"cat_6_PktDiscard": 0,
"cat_6_PktDiscardTotal": 0,
"cat_6_PktSlice": 0,
"cat_6_IOPriority": 0,
"cat_6_ChunkID": 0,
"cat_6_CmdLine": "",
"cat_6_StreamName": "",
"cat_6_FilterBPF": "",
"cat_6_CPUIidle": 0,
"cat_6_CPUFetch": 0,
"cat_6_CUPProcess": 0,
"cat_6_CPUSend": 0,
"cat_7_Enable": false,
```



```
"cat_7_UDPMain": 0,
"cat_7_TSPCAP": 0,
"cat_7_ReadPkt": 0,
"cat_7_ReadByte": 0,
"cat_7_ReadTotalPkt": 0,
"cat_7_ReadTotalByte": 0,
"cat_7_ReadGbps": 0,
"cat_7_ReadMpps": 0,
"cat_7_WritePkt": 0,
"cat_7_WriteByte": 0,
"cat_7_WriteTotalPkt": 0,
"cat_7_WriteTotalByte": 0,
"cat_7_WriteGbps": 0,
"cat_7_WriteMpps": 0,
"cat_7_PendingByte": 0,
"cat_7_PktDiscard": 0,
"cat_7_PktDiscardTotal": 0,
"cat_7_PktSlice": 0,
"cat_7_IOPriority": 0,
"cat_7_ChunkID": 0,
"cat_7_CmdLine": "",
"cat_7_StreamName": "",
"cat_7_FilterBPF": "",
"cat_7_CPUIdle": 0,
"cat_7_CPUFetch": 0,
"cat_7_CPUProcess": 0,
"cat_7_CPUSend": 0,
"cat_EnableCnt": 2,
"cat_ReadPkt": 209788,
"cat_ReadByte": 299944000,
"cat_ReadTotalPkt": 22582184,
"cat_ReadTotalByte": 31848580480,
"cat_ReadGbps": 0.436569,
"cat_ReadMpps": 0.436569,
"cat_WritePkt": 229982,
"cat_WriteByte": 327284233,
"cat_WriteTotalPkt": 22582184,
"cat_WriteTotalByte": 31848580480,
"cat_WriteGbps": 0.655844,
"cat_WriteMpps": 0.655844
},
"ptp": {
  "module": "system",
  "subsystem": "ptp",
  "timestamp": 1695785514,
  "ver": "9120",
  "TimeFPGA": 1695785514554346200,
  "TimeSYS": 1695785514357678000,
  "GMOffset": 0,
  "GMSync": false,
  "GMMaster": "",
  "SysOffset": 0,
  "SysSync": false,
```

```
"iSysOffset": 0,  
"iSysSync": true,  
"NTPOffset": 0,  
"NTPSync": false,  
"NTPMaster": "",  
"GMUpTime": 0,  
"SysUptime": 0,  
"iSysUptime": 1131,  
"PPSUptime": 760,  
"NTPUptime": 0,  
"PPSPeriod": 6.39992575,  
"PPSOffset": -438,  
"PPSdPhase": 1929,  
"PPSdZero": -438,  
"PPSCnt": 1293,  
"Clk156Period": 0,  
"Clk156Offset": 0,  
"Clk250Period": 0,  
"Clk250Offset": 0,  
"Clk322Period": 0,  
"Clk322Offset": 0  
}  
}
```

As a raw JSON blob

```
{"timestamp":1695785768,"ver":"9120","temperature":{"module":"system","subsystem":"temperatu
```

# Linux Containers (LXC)

# Applications

Reference LX containers can be found here



public/container at master · fmadio/public  
GitHub

# Configuration

## Install

The following are generalized steps to install setup and start the specified container. Please consult any container specific documentation in addition to this process.

### Step 1)

Download the LXC into the directory below

```
/opt/fmadio/lxc/
```

Check the MD5 sum against the published value

Untar the tarball in that directory as root.

Example below shows the steps for the mdgap lxc container

```
fmadio@fmadio100v2-228U:/opt/fmadio/lxc$ sudo curl -O https://firmware.fmad.io/download/cont
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
100 824M  100 824M    0     0  14.2M      0  0:00:58  0:00:58  --:--:-- 14.9M
fmadio@fmadio100v2-228U:/opt/fmadio/lxc$

fmadio@fmadio100v2-228U:/opt/fmadio/lxc$ md5sum mdgap-20230512.tar.gz
5da383edd150b6ac9b5cedc8550fdca3  mdgap-20230512.tar.g

fmadio@fmadio100v2-228U:/opt/fmadio/lxc$ sudo tar xf mdgap-20230512.tar.gz
fmadio@fmadio100v2-228U:/opt/fmadio/lxc$
```

### Step 2)

Most LXCs have an install script in the root directory, it will configure the container requiring some information such as IP Netmask. Gateway etc.

```

fmadio@fmadio100v2-228U:/opt/fmadio/lxc$ sudo su
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc# cd mdgap-20230512
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512# ls -al
total 24
drwxrwx---    3 root    root          4096 May 12 08:16 ./
drwxr-xr-x   35 root    root          4096 Jun 24 05:41 ../
-rw-r--r--    1 root    root          1852 May 12 08:16 config
-rwxr-xr-x    1 root    root          5544 May 10 06:18 install.lua
drwxr-xr-x   28 root    root          4096 May  8 05:07 rootfs/
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512#

root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512# sudo ./install.lua
rm config
mkdir /mnt/store0/log/lxc/mdgap/clickhouse -p
mkdir /mnt/store0/log/lxc/mdgap/grafana -p
mkdir /mnt/store0/log/lxc/mdgap/remote -p
mkdir /mnt/store0/lxc/data/mdgap/clickhouse -p
mkdir /mnt/store0/lxc/data/mdgap/grafana -p
Container IP Address? (e.g. 192.168.1.100)
192.168.2.178
Container Netmask? (e.g. 24 for 255.255.255.0)
24
Container Gateway? (e.g. 192.168.1.1)
192.168.2.254
Container DNS? (e.g. 192.168.1.1)
1.1.1.1
-----
IP      : 192.168.2.178
CIDR    : 24
GW      : 192.168.2.254
DNS     : 1.1.1.1
echo nameserver 1.1.1.1 > rootfs/etc/resolv.conf

echo fmadio100v2-228U-mdgap > rootfs/etc/hostname
echo 127.0.0.1 fmadio100v2-228U-mdgap >> rootfs/etc/hosts
done 13.161981Sec 0.219366Min
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512#

```

### Step 3)

Modify the containers configuration. such as adding rings or shared mount points

```
/opt/fmadio/lxc/<container>/config
```

Example is enabling the Eurex lxc ring to the container by uncommenting in the configuration file, the line

```
lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_eurex          opt/fmadio/queue/lxc
```

In the configuration below

```
.
.
.
#set log mount
lxc.mount.entry = /mnt/store0/log/lxc/mdgap mnt/store0/log none bind,create=dir 0 0

#set data path
lxc.mount.entry = /mnt/store0/lxc/data/mdgap mnt/store0/mdgap none bind,create=dir 0 0

#set lxc ring mount
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring0                opt/fmadio/queue/lxc_ring0
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_cme        opt/fmadio/queue/lxc

lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_eurex      opt/fmadio/queue/lxc

#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_nasdaq     opt/fmadio/queue/lxc_ring_md
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_siac_cqs   opt/fmadio/queue/lxc_ring_md
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_siac_cts   opt/fmadio/queue/lxc_ring_md
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_siac_opra  opt/fmadio/queue/lxc_ring_md
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_lse_mitch  opt/fmadio/queue/lxc_ring_md
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_mdgap_euronext   opt/fmadio/queue/lxc_ring_
```

Ensure to save the changes

### Step 3a)

Generally containers require LXC Rings to feed data to. In this example we create an lxc ring named "mdgap\_eurex" as the container references this ring in step 3) it must be created for the container to start correctly.

```
fmadiocli "config push lxc add mdgap_eurex"
```

Example output as follows

```
/root@fmadiao100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512# fmadiocli "config push lxc ad
```

Disable cycle calibration

```
[Sat Jun 24 05:57:42 2023] -----
[Sat Jun 24 05:57:42 2023] _/ ____\_____ _____ __| _/|__| ____
[Sat Jun 24 05:57:42 2023] \  __\ / \__ \ / __ | | | / _ \
[Sat Jun 24 05:57:42 2023] | | | Y Y \ / __ \ / / | | | ( <_> )
[Sat Jun 24 05:57:42 2023] |__| |__| | /(\____ / \____ | |__| \____/
[Sat Jun 24 05:57:42 2023]          \ /      \ /      \ /
[Sat Jun 24 05:57:42 2023] =====
[Sat Jun 24 05:57:42 2023] +- Packets confiscated by Customs +-
[Sat Jun 24 05:57:42 2023]
[Sat Jun 24 05:57:42 2023] type '?' for command information
[Sat Jun 24 05:57:42 2023] type '???' for verbose information
[Sat Jun 24 05:57:42 2023]
[Sat Jun 24 05:57:42 2023] History: 1530
[Sat Jun 24 05:57:42 2023] CmdLine [config push lxc add mdgap_eurex]
[Sat Jun 24 05:57:42 2023] Cmd [config push lxc add mdgap_eurex]
[Sat Jun 24 05:57:42 2023] Created Ring named [mdgap_eurex]
cycle calibration disabled
null output
RING reset
RING file [/opt/fmadiao/queue/lxc_ring_mdgap_eurex]
RING[/opt/fmadiao/queue/lxc_ring_mdgap_eurex          ] Size mismatch 0 12595200
RING[/opt/fmadiao/queue/lxc_ring_mdgap_eurex          ] Size      : 12595200 16777216
RING[/opt/fmadiao/queue/lxc_ring_mdgap_eurex          ] Version:      0      100
RING[/opt/fmadiao/queue/lxc_ring_mdgap_eurex          ] version wrong force reset
RING[/opt/fmadiao/queue/lxc_ring_mdgap_eurex          ] Put:0 0 0x7f68002f0000
RING[/opt/fmadiao/queue/lxc_ring_mdgap_eurex          ] Get:0 0 0x7f68002f1000
[Sat Jun 24 05:57:43 2023] created ring [/opt/fmadiao/queue/lxc_ring_mdgap_eurex]
[Sat Jun 24 05:57:43 2023] New Push LXC target [/opt/fmadiao/queue/lxc_ring_mdgap_eurex]
done 0.196703Sec 0.003278Min
root@fmadiao100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512#
```

#### Step 4)

Add the container to the system configuration file using `fmadiocli` utility

Add the container using the command

```
fmadiocli "config lxc add <container name>
```

Example shown below



```

root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512# fmadiocli "config lxc add mdga
Disable cycle calibration
[Sat Jun 24 05:52:30 2023] -----
[Sat Jun 24 05:52:30 2023] _/ ____\_____ _____ __| _/|_| ____
[Sat Jun 24 05:52:30 2023] \ __\ / \__ \ /__ | | | / _ \
[Sat Jun 24 05:52:30 2023] | | | Y Y \ / __ \_ /_/ | | | ( <_> )
[Sat Jun 24 05:52:30 2023] |_| |_|_| /(\____ /\____ | |_| \____/
[Sat Jun 24 05:52:30 2023]          \ /      \ /      \ /
[Sat Jun 24 05:52:30 2023] =====
[Sat Jun 24 05:52:30 2023] +- Packets confiscated by Customs +-
[Sat Jun 24 05:52:30 2023]
[Sat Jun 24 05:52:30 2023] type '?' for command information
[Sat Jun 24 05:52:30 2023] type '???' for verbose information
[Sat Jun 24 05:52:30 2023]
[Sat Jun 24 05:52:30 2023] History: 1530
[Sat Jun 24 05:52:30 2023] CmdLine [config lxc add mdgap-20230512]
[Sat Jun 24 05:52:30 2023] Cmd [config lxc add mdgap-20230512]
[Sat Jun 24 05:52:30 2023] Added container [mdgap-20230512] to the configuration
done 0.195539Sec 0.003259Min
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512#

```

## Step 5)

Start the container using fmadiocli utility

```
fmadiocli "config lxc start <container name>"
```

Example starting the MDGap container

```

root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512# fmadiocli "config lxc start md
fmad fmadlua Jun 24 2023 (/opt/fmadio/bin/fmadiolua --nocal /opt/fmadio/bin/fmadiocli config
Disable cycle calibration
[Sat Jun 24 05:59:14 2023]      -----
[Sat Jun 24 05:59:14 2023]  _/  ____\_____  _____  __|  _/|__|  ____
[Sat Jun 24 05:59:14 2023]  \   __\ /   \  \__  \  /  __ | | | /  _ \
[Sat Jun 24 05:59:14 2023]  | | |  Y Y  \ /  __ \_ /  /_ / | | | ( <_> )
[Sat Jun 24 05:59:14 2023]  |__| |__|_| /(\_____ / \_____ | |__| \____/
[Sat Jun 24 05:59:14 2023]                      \ /      \ /      \ /
[Sat Jun 24 05:59:14 2023]  =====
[Sat Jun 24 05:59:14 2023]      +- Packets confiscated by Customs +-
[Sat Jun 24 05:59:14 2023]
[Sat Jun 24 05:59:14 2023]  type '?' for command information
[Sat Jun 24 05:59:14 2023]  type '???' for verbose information
[Sat Jun 24 05:59:14 2023]
[Sat Jun 24 05:59:14 2023]  History: 1530
[Sat Jun 24 05:59:14 2023]  CmdLine [config lxc start mdgap-20230512]
[Sat Jun 24 05:59:14 2023]  Cmd [config lxc start mdgap-20230512]
[Sat Jun 24 05:59:14 2023]  sudo lxc-start -n mdgap-20230512 --logfile /tmp/lxc_mdgap-2023051
[Sat Jun 24 05:59:16 2023]
[Sat Jun 24 05:59:16 2023]  use the following on a shell to attach to the conatiners console
[Sat Jun 24 05:59:16 2023]  sudo lxc-attach -n mdgap-20230512
[Sat Jun 24 05:59:16 2023]
done 1.899527Sec 0.031659Min
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512#

```

## Step 6)

Optionally attach to the containers console using the command

```
sudo lxc-attach -n <container name>
```

Example of MDGap container

```

root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/mdgap-20230512# lxc-attach -n mdgap-20230512
root@fmadio100v2-228U-mdgap:/#

```

Please see any container specific configuration and setup documentation

# Container Packet Processing

See our public github repo for details



public/container/centos7 at master · fmadio/public  
GitHub

# Mounting Host Scratch Disk

Mounting the host systems file system into the LXC container can be very helpful in many situations.

Easiest way is via the lxc config file example shows mounting the host /mnt/store1 directory directly into the container. Please ensure rootfs/mnt/store1 directory has been created within the container.

```
lxc.mount.entry = /mnt/store1 mnt/store1 none bind 0 0
```

NOTE that it is critical to have no leading "/" in the container mount point (making it a relative mount point)

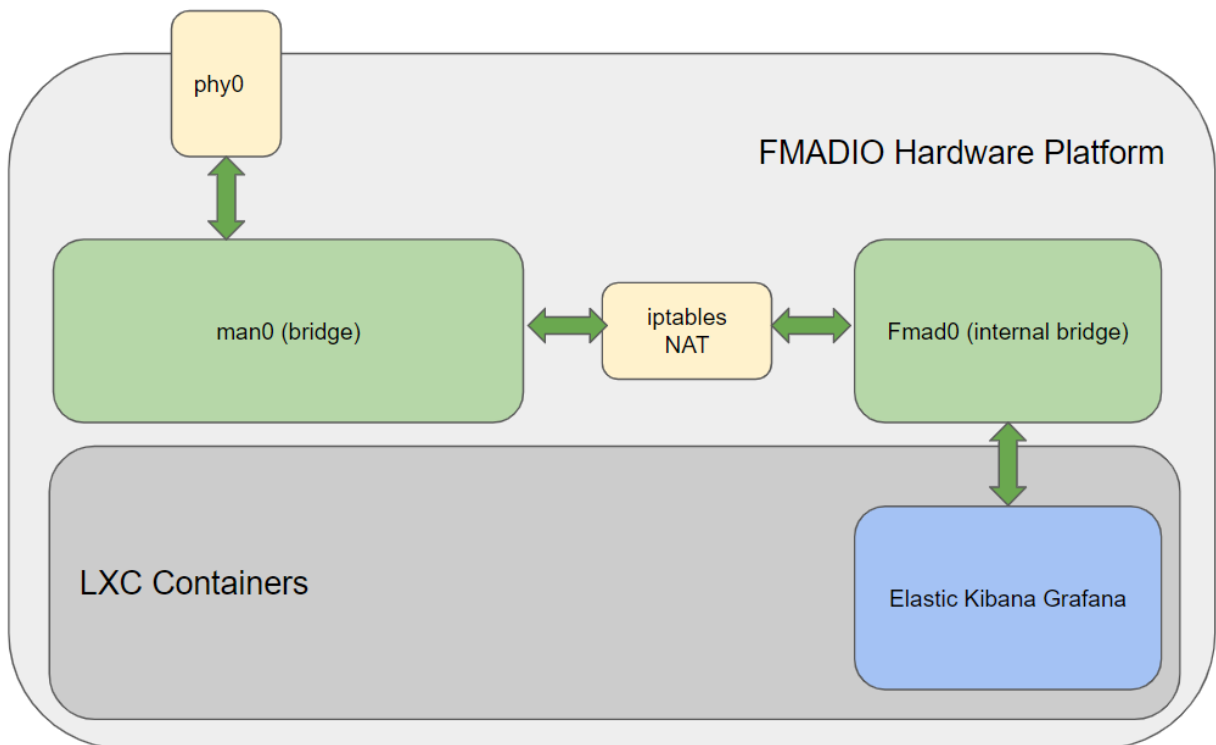
[https://wiki.debian.org/LXC#Bind\\_mounts\\_inside\\_the\\_container](https://wiki.debian.org/LXC#Bind_mounts_inside_the_container)

# Container Private Network

The container can be configured to have a independent public IP address (fully bridged), a private internal IP (internal NAT) or both. All options have Pros/Cons and need to be considered for final deployment.

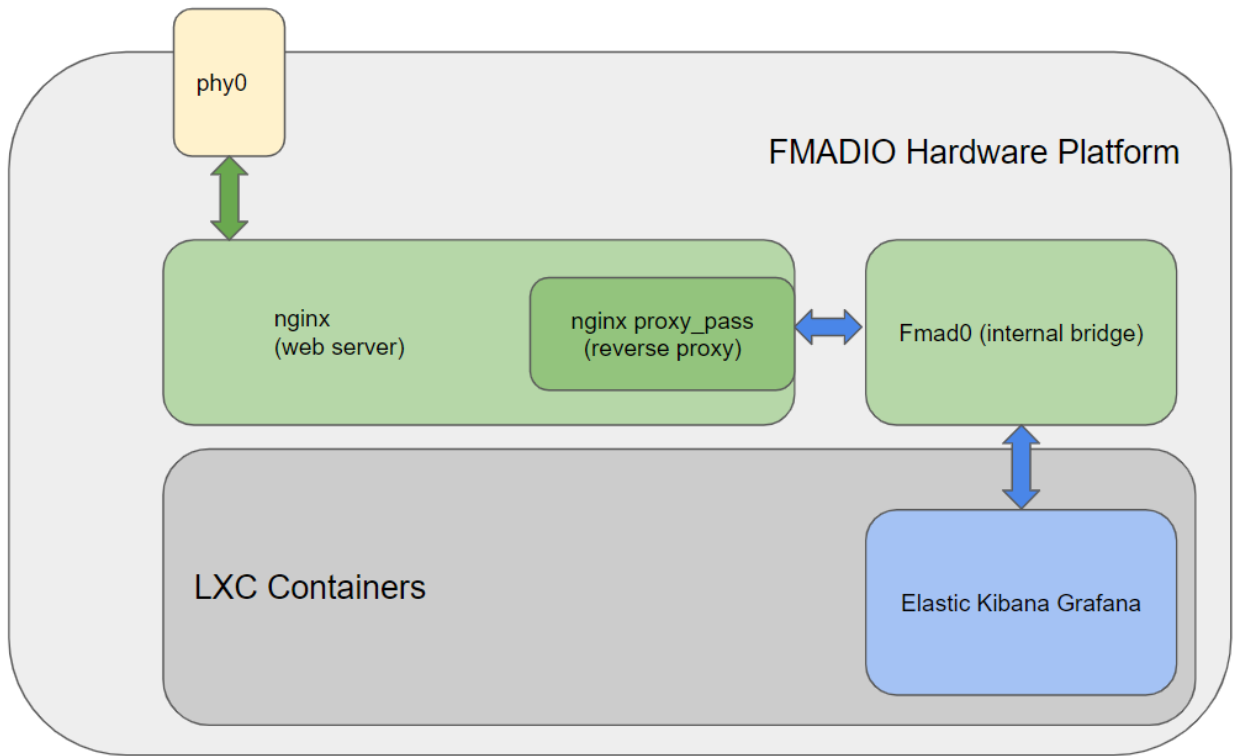
A common setup is to have the container entirely private using nginx as a proxy or iptables to port forward to the container.

Block diagram below shows Private only Network Topology



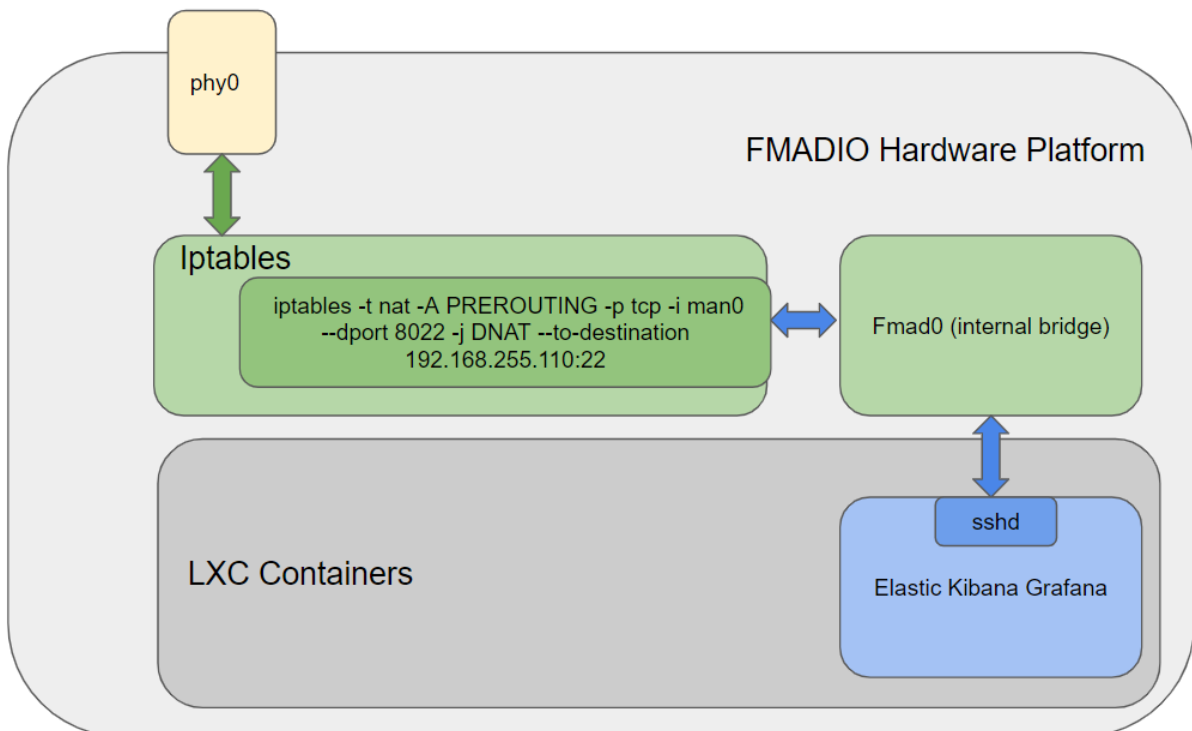
Private LXC Networking

With external access using NGINX proxy such as below.



FMADIO NGINX Proxy to Private Container

Or use iptables to port forward packets thru the NAT, such as below



FMADIO Private Container Port Forwarding

## IPTables Configuration

To enable the NATed bridge between man0 (public) network and the fmad0 (private) network the following IPTables config needs to be set

```
echo 1 >/proc/sys/net/ipv4/ip_forward
sudo iptables -A FORWARD -i fmad0 -o man0 -j ACCEPT
sudo iptables -A FORWARD -i man0 -o fmad0 -m state --state ESTABLISHED,RELATED -j ACCEPT
sudo iptables -t nat -A POSTROUTING -o man0 -j MASQUERADE
sudo iptables -t nat -A PREROUTING -i man0 -p tcp --dport 9000 -j DNAT --to-destination 192.
sudo iptables-save > /opt/fmadio/etc/iptables.conf
```

Alternatively the following /opt/fmadio/etc/iptables.conf file can be used (requires a system reboot or iptables-restore) to take effect

```
# Generated by iptables-save v1.6.1 on Tue Apr 19 16:29:55 2022
*nat
:PREROUTING ACCEPT [10:2058]
:INPUT ACCEPT [2:104]
:OUTPUT ACCEPT [2:260]
:POSTROUTING ACCEPT [2:260]
-A POSTROUTING -o man0 -j MASQUERADE
COMMIT
# Completed on Tue Apr 19 16:29:55 2022
# Generated by iptables-save v1.6.1 on Tue Apr 19 16:29:55 2022
*filter
:INPUT ACCEPT [22:3579]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [23:3592]
-A FORWARD -i fmad0 -o man0 -j ACCEPT
-A FORWARD -i man0 -o fmad0 -m state --state RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Tue Apr 19 16:29:55 2022
```

### Forwarding a specific TCP Port to the LXC

The above is general setup, to forward a specific port from the Host IP to the LXC container IP run as follows.

**NOTE: if using 10G management interface replace man0 with man10**

1) Forwarding port 9000 on the host to port 3000 on the LXC

(in this case LXC is configured as 192.168.255.191)

```
sudo iptables -A FORWARD -i fmad0 -o man0 -j ACCEPT
sudo iptables -A FORWARD -i man0 -o fmad0 -m state --state ESTABLISHED,RELATED -j ACCEPT
sudo iptables -t nat -A POSTROUTING -o man0 -j MASQUERADE
sudo iptables -t nat -A PREROUTING -i man0 -p tcp --dport 9000 -j DNAT --to-destination 192.
sudo iptables-save > /opt/fmadio/etc/iptables.conf
```

## Container Network Settings

Container network settings need to be the following

```
Gateway : 192.168.255.2
DNS      : 8.8.8.8   (or something else)
```

## Container Network IP Address

List of private container addresses

IP Address	Container	Description
192.168.255.2	FMADIO Host	FMADIO Host IP Address
192.168.255.10	FShark	FMADIO Internal Wireshark Lite
192.168.255.100	Ubuntu Desktop	Ubuntu Desktop
192.168.255.110	Elastic Search 7.x	Elastic Search 7.x Container
192.168.255.111	Elastic Search 8.x	Elastic Search 8.x Container
192.168.255.120	Suricata 6.x	Suricata 6.x Container (CentOS)
192.168.255.130	Zeek	Zeek Container (CentOS)



# FMADIO Shark

FMADIO Shark is a self contained LXC container that runs on the FMADIO Packet Capture systems. It provides a "Wireshark Lite" way to navigate thats fully compatible with Wireshark, as the backend is Wireshark.

To configure and run as follows

## Download Latest FMADIO Shark Release

Copy the tarball to /tmp

## Unpack FMADIO Shark

Unpack the tarball into the /opt/fmadio/lxc directory

```
/mnt/store0/lxc/lib/lxc$ sudo tar xfv /tmp/fmadshark-release_20220317_164527-314-4436d67.ta
./fshark_20220317_1646/
./fshark_20220317_1646/config
./fshark_20220317_1646/install.lua
./fshark_20220317_1646/rootfs/
./fshark_20220317_1646/rootfs/lib/
./fshark_20220317_1646/rootfs/lib/udev/
./fshark_20220317_1646/rootfs/lib/udev/ifupdown-hotplug
```

## Link the latest release

Symlink the latest release. Optionally removing the symlink of any older version

```
/mnt/store0/lxc/lib/lxc$ sudo ln -s fshark_20220317_1646/ fshark
```

## Run the Container Install Script

```
/mnt/store0/lxc/lib/lxc$ cd fshark

/mnt/store0/lxc/lib/lxc/fshark_20220317_1646$ sudo ./install.lua
fmad fmadlua Mar 17 2022
loading filename [./install.lua]
Host Address:192.168.2.225
sudo sed -i "s/.*CONFIG_HOST$/          ip = \"192.168.1.1\", -- (Thu Mar 17 20:52:41 2022) CO
done 0.005622Sec 0.000094Min
/mnt/store0/lxc/lib/lxc/fshark_20220317_1646$
```

## Enable in the Config File

Edit the file

```
/opt/fmadio/etc/time.lua
```

Setting the following ["FShark"] = true, if the field does not exist then create it

```
["PCAP"] =
{
    ["TimeStampMode"] = "nic",
    ["TimeResolution"] = "nsec",
    ["TimeSortDepth"] = 256,
    ["Decap"]          = false,
    ["Slice"]          = 0,
    ["DownloadIdleTimeout"] = 30000000000,
    ["FShark"]         = true,
    ["DownloadAPI"]    = "v1",
},
```

## Enable automatic start on boot

By default FShark does not start on boot, enabling this in the config uses the generic LXC container framework.

Edit the config file

```
/opt/fmadio/etc/time.lua
```

Near the bottom section of the config there is a "Container" section. The example below shows a basic FShark only configuration, depending on your usage there may be additional containers configured to run.

```
["Container"] =
{
  ["Enable"]      = true,

  ["RingCnt"]    = 4,
  ["List"]       =
  {
    [1] = { Name = "fshark", OnBoot = true},
  }
}
```

After configuration update, reboot the system

### Check FMADIO Shark is running

Check FShark is running using the fmadiocli utility as follows

```
fmadiocli "show container"
```

FShark should be installed and running as highlighted in red below.

```
fmad fmadlua Jul 27 2022 (/opt/fmadio/bin/fmadiolua --nocall /opt/fmadio/bin/fmadiocli show container )
Disable cycle calibration
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
[Thu Jul 28 14:22:37 2022]
-----
  +- Packets confiscated by Customs +-
-----
type '?' for command information
type '???' for verbose information
-----
History: 4
[Thu Jul 28 14:22:37 2022] cmdline [show container]
[Thu Jul 28 14:22:37 2022] Cmd [show container]
[Thu Jul 28 14:22:37 2022] Enable : true
[Thu Jul 28 14:22:37 2022] Ringcnt : 4
-----
Name      OnBoot  Install  State
-----
fshark    true    yes      RUNNING
elastic   true    yes      STOPPED
suricata  true    yes      RUNNING
-----
done 0.021570sec 0.000359min
fmadio@fmadio40v3SM-455:~$
```

Check FShark status

### Packet Browser

PacketBrowser and PacketScope should be visible on the GUI as follows

# PACKET BROWSER

/capture/

PCAP2	PScp	Ma...	File	Bytes	GBytes	Packet Count	Description	Del
			test1500_long_20211030_1557	1,048,576 B	0 GB		Mon . 08:34:33 . 22-07-2554	
			test2_20210810_1131	1,048,576 B	0 GB		Mon . 08:34:33 . 22-07-2554	
			asdf_20210810_1033	1,048,576 B	0 GB		Mon . 08:34:33 . 22-07-2554	
			teset1234_20220317_2047	160,043,106,304 B	160 GB		Thu . 20:51:23 . 17-03-2022	
			test_20220129_2140	160,202,489,856 B	160 GB		Sat . 21:44:14 . 29-01-2022	
			asdf_20211112_2234	47,185,920 B	0 GB		Fri . 23:59:58 . 12-11-2021	
			test1234_20211112_2217	1,048,576 B	0 GB		Fri . 22:18:15 . 12-11-2021	
			test1234_20211110_1137	1,248,591,872 B	1 GB		Wed . 12:27:45 . 10-11-2021	
			test_20211103_1701	8,003,780,608 B	8 GB		Wed . 17:05:27 . 03-11-2021	
			test_20211103_1649	8,004,829,184 B	8 GB		Wed . 16:53:09 . 03-11-2021	
			test_20211103_1631	8,004,829,184 B	8 GB		Wed . 16:34:52 . 03-11-2021	
			test_20211103_1619	8,004,829,184 B	8 GB		Wed . 16:26:12 . 03-11-2021	
			test_20211103_1610	8,004,829,184 B	8 GB		Wed . 16:13:54 . 03-11-2021	
			test_20211103_1555	8,004,829,184 B	8 GB		Wed . 15:58:25 . 03-11-2021	
			testfw_20211103_1532	8,004,829,184 B	8 GB		Wed . 15:34:54 . 03-11-2021	
			test_20211102_2034	8,004,829,184 B	8 GB		Tue . 20:37:10 . 02-11-2021	
			test_20211102_2024	8,003,780,608 B	8 GB		Tue . 20:27:24 . 02-11-2021	

FShark Packet Viewing

# FMADIO Shark2

FMADIO FShark2 is a full Ubuntu desktop accessible via RDP or HTTP client. This has the latest Wireshark binary plus additional utilis enabling full packet investigations on the system.

## Port Forward Access

In many enviroments creating an additional IP for FShark2 is problematic. Instead port forwarding ports on the FMADIO Capture Appliance to the FShark2 device is a simpler approach.

### Step 1) Install FShark2 package

Download latest fshark2 release

```
curl -O https://firmware.fmad.io/download/container/fshark2-current.tar.gz .
```

Example

```
fmadio@fmadio100v2-228U:/mnt/store0/tmp2$ curl -O https://firmware.fmad.io/download/containe
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left  Speed
100 1292M  100 1292M    0     0  14.7M      0  0:01:27  0:01:27  --:--:-- 15.8M
fmadio@fmadio100v2-228U:/mnt/store0/tmp2$
```

Extract to LXC directory

```
sudo tar xfvz fshark2-current.tar.gz -C /opt/fmadio/lxc/
```

Example output:

```
fshark2-202310251136/  
fshark2-202310251136/fmadiocli-start.lua  
fshark2-202310251136/install.lua  
fshark2-202310251136/fmadiocli-del.lua  
fshark2-202310251136/config  
fshark2-202310251136/vm-install.lua  
fshark2-202310251136/fmadiocli-stop.lua  
fshark2-202310251136/rootfs/  
fshark2-202310251136/rootfs/sys/  
fshark2-202310251136/rootfs/proc/  
fshark2-202310251136/rootfs/home/  
fshark2-202310251136/rootfs/home/fmadio/  
  
fshark2-202310251136/rootfs/home/fmadio/.bash_logout  
fshark2-202310251136/rootfs/home/fmadio/.vimrc  
fshark2-202310251136/rootfs/home/fmadio/.bashrc  
fshark2-202310251136/rootfs/home/fmadio/.config/  
fshark2-202310251136/rootfs/home/fmadio/.config/tint2/  
fshark2-202310251136/rootfs/home/fmadio/.config/tint2/tint2rc  
fshark2-202310251136/rootfs/home/fmadio/.config/openbox/  
fshark2-202310251136/rootfs/home/fmadio/.config/openbox/autostart  
fshark2-202310251136/rootfs/home/fmadio/fmadio-background.png  
fshark2-202310251136/rootfs/home/fmadio/.mozilla/  
fshark2-202310251136/rootfs/home/fmadio/.mozilla/extensions/  
fshark2-202310251136/rootfs/home/fmadio/.mozilla/firefox/  
.  
.
```

Or download an extract at the same time

```
curl -s https://firmware.fmad.io/download/container/fshark2-current.tar.gz | sudo tar xfv
```

## Step 2) Configure LXC

Change directory to the /opt/fmadio/lxc/fshark2-<insert version>/

Run the install script. If no IP address for the container is used (e.g. fully NATed / port forward) leave the IP info blank

```
./install.lua
```

Example output

```

root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/fshark2-202310251136# ./install.lua
fmad fmadlua Nov 26 2023 (/opt/fmadio/bin/fmadiolua --nocal ./install.lua )
Disable cycle calibration
man0 imported

man0 ipv4 addr      [nil]
man0 ipv4 netmask  [nil]
man0 ipv4 gw       [nil]
man0 ipv4 dns      [nil]
lxcname            [nil]
fshark2-202310251136
rm config
mkdir /mnt/store0/log/lxc/fshark2-202310251136 -p
mkdir /mnt/store0/lxc/data/fshark2-202310251136 -p
Container IP Address? (e.g. 192.168.1.100)

Container Netmask? (e.g. 24 for 255.255.255.0)

Container Gateway? (e.g. 192.168.1.1)

Container DNS? (e.g. 192.168.1.1)

-----
IP      :
CIDR    :
GW      :
DNS     :
rm /opt/fmadio/lxc/fshark2-202310251136/rootfs/etc/resolv.conf
touch /opt/fmadio/lxc/fshark2-202310251136/rootfs/etc/resolv.conf
echo nameserver > /opt/fmadio/lxc/fshark2-202310251136/rootfs/etc/resolv.conf
echo fmadio100v2-228U-fshark2 > /opt/fmadio/lxc/fshark2-202310251136/rootfs/etc/hostname
echo 127.0.0.1 fmadio100v2-228U-fshark2 >> /opt/fmadio/lxc/fshark2-202310251136/rootfs/etc/h
/opt/fmadio/bin/fmadiocli "config lxc add fshark2-202310251136"
fmad fmadlua Nov 26 2023 (/opt/fmadio/bin/fmadiolua --nocal /opt/fmadio/bin/fmadiocli config
Disable cycle calibration
[Tue Nov 28 15:07:13 2023] CmdLine [config lxc add fshark2-202310251136]
[Tue Nov 28 15:07:13 2023] Cmd [config lxc add fshark2-202310251136]
[Tue Nov 28 15:07:13 2023]**ERROR** Container named [fshark2-202310251136] already exists
[Tue Nov 28 15:07:13 2023]
[Tue Nov 28 15:07:13 2023] Example Usage:
[Tue Nov 28 15:07:13 2023] > config lxc add <container name> : adds the lxc
[Tue Nov 28 15:07:13 2023]
done 0.015017Sec 0.000250Min
done
done 2.942201Sec 0.049037Min
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/fshark2-202310251136#

```

### Step 3) Configure for NAT / Port forwarding

Comment out the lxc.net.1 (bridged interface) in the Config and set the default gateway to 192.168.255.2 (hosts internal interface)

```
lxc.net.0.ipv4.gateway = 192.168.255.2
```

## Example Config

```
# lxc config generate by ubuntu install.lua
# set cpu to fmadio100v2 analytic
lxc.cgroup.cpuset.cpus=73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95

lxc.include = /usr/share/lxc/config/ubuntu.common.conf
lxc.arch = x86_64

lxc.rootfs.path = dir:/opt/fmadio/lxc/fshark2-202310251136/rootfs
lxc.uts.name = fmadio100v2-228U-fshark2

lxc.net.0.type = veth
lxc.net.0.link = fmad0
lxc.net.0.flags = up
lxc.net.0.ipv4.address = 192.168.255.11/24
lxc.net.0.ipv4.gateway = 192.168.255.2

#lxc.net.1.type = veth
#lxc.net.1.link = man0
#lxc.net.1.flags = up
#lxc.net.1.ipv4.address = /
#lxc.net.1.ipv4.gateway =

# Mount Data Directory
lxc.mount.entry = /mnt/store0/lxc/data/fshark2-202310251136 mnt/data/ none bind,create=dir

# Mount Log Directory
lxc.mount.entry = /mnt/store0/log/lxc/fshark2-202310251136 mnt/log/ none bind,create=dir 0

# map passthru queue
#lxc.mount.entry = /opt/fmadio/queue/lxc_ring_fshark2 opt/fmadio/queue/lxc_ring_fshark2 none

lxc.prlimit.nofile = 65535
lxc.prlimit.memlock = unlimited
```

## Step 4) AutoStart FSHARK2 on system boot

To enable automatic starting of the FSHAK2 container on system boot

```
fmadiocli "config lxc boot fshark2-202310251136"
```

Example output:



```
fmadio@fmadio100v2-228U:/opt/fmadio/lxc$ fmadiocli "config lxc boot fshark2-202310251136"
fmad fmadlua Nov 26 2023 (/opt/fmadio/bin/fmadiolua --nocall /opt/fmadio/bin/fmadiocli config
Disable cycle calibration
[Tue Nov 28 15:34:09 2023] CmdLine [config lxc boot fshark2-202310251136]
[Tue Nov 28 15:34:09 2023] Cmd [config lxc boot fshark2-202310251136]
[Tue Nov 28 15:34:10 2023] Set container [fshark2-202310251136] to boot on system start
done 0.165707Sec 0.002762Min
fmadio@fmadio100v2-228U:/opt/fmadio/lxc$
```

## Step 5) Start the Container manually

To start the container

```
sudo lxc-start -n fshark2-202310251136
```

Example output

```
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/fshark2-202310251136# sudo lxc-start -n fshark
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/fshark2-202310251136#
```

If it prints any messages it means there is a configuration error somewhere

## Step 6) Confirm FSHAK2 is running

Check the port 3000 (HTTP browser) and 3389 (RDP) are open

```
sudo lxc-attach -n fshark2-202310251136 -- netstat -antl
```

Example output, can see both ports are listed

```
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/fshark2-202310251136# sudo lxc-attach -n fshar
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:4822            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp6       0      0 127.0.0.1:3350          :::*                    LISTEN
tcp6       0      0 :::3000                 :::*                     LISTEN
tcp6       0      0 :::3389                 :::*                     LISTEN
root@fmadio100v2-228U:/mnt/store0/lxc/lib/lxc/fshark2-202310251136#
```

## Step 6) Configure IP Port forwarding

Copy the following iptables to the configuration directory

```
cp /opt/fmadio/etc_ro/iptables_fshark2_portfwd.conf iptables.conf
```

Example output:

```
fmadio@fmadio100v2-228U:/opt/fmadio/etc$ cp /opt/fmadio/etc_ro/iptables_fshark2_portfwd.conf
fmadio@fmadio100v2-228U:/opt/fmadio/etc$
```

Manually load the iptables setting

```
sudo iptables-restore < ./iptables.conf
```

Example output:

```
fmadio@fmadio100v2-228U:/opt/fmadio/etc$ sudo iptables-restore < ./iptables.conf
fmadio@fmadio100v2-228U:/opt/fmadio/etc$
```

For reference the /opt/fmadio/etc\_ro/iptables\_fshark2\_portfwd.conf file looks like the following

```
# Generated by iptables-save v1.6.1 on Tue Nov 28 15:30:44 2023
*nat
:PREROUTING ACCEPT [307:49051]
:INPUT ACCEPT [3:156]
:OUTPUT ACCEPT [112:11328]
:POSTROUTING ACCEPT [110:11210]
-A PREROUTING -p tcp -m tcp --dport 7000 -j DNAT --to-destination 192.168.255.11:3000
-A PREROUTING -p tcp -m tcp --dport 7001 -j DNAT --to-destination 192.168.255.11:3389
-A POSTROUTING -o man0 -j MASQUERADE
COMMIT
# Completed on Tue Nov 28 15:30:44 2023
# Generated by iptables-save v1.6.1 on Tue Nov 28 15:30:44 2023
*filter
:INPUT ACCEPT [15028:3079580]
:FORWARD ACCEPT [1651:509287]
:OUTPUT ACCEPT [14446:4545136]
COMMIT
# Completed on Tue Nov 28 15:30:44 2023
```

## Step 7) Confirm IP Tables setting is correct

Output the iptables information

```
sudo iptables -L -n -v -t nat
```

Example output:

```

fmadio@fmadio100v2-228U:/opt/fmadio/etc$ sudo iptables -L -n -v -t nat
Chain PREROUTING (policy ACCEPT 254 packets, 39578 bytes)
  pkts bytes target     prot opt in     out     source            destination
     8   416 DNAT       tcp  --  *      *        0.0.0.0/0         0.0.0.0/0         tcp
     0     0 DNAT       tcp  --  *      *        0.0.0.0/0         0.0.0.0/0         tcp

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source            destination

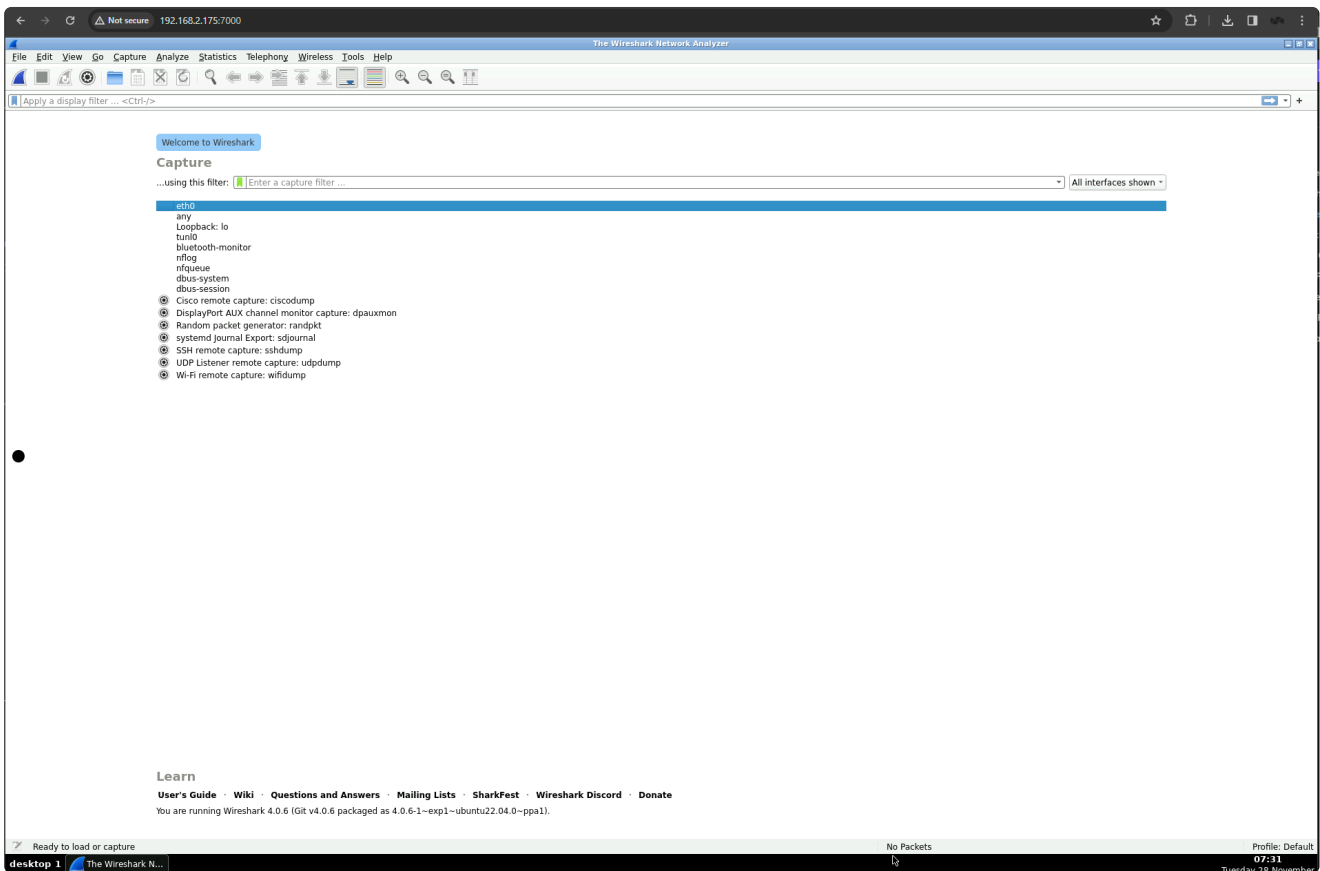
Chain OUTPUT (policy ACCEPT 71 packets, 8550 bytes)
  pkts bytes target     prot opt in     out     source            destination

Chain POSTROUTING (policy ACCEPT 79 packets, 8966 bytes)
  pkts bytes target     prot opt in     out     source            destination
     0     0 MASQUERADE all  --  *      man0    0.0.0.0/0         0.0.0.0/0
fmadio@fmadio100v2-228U:/opt/fmadio/etc$

```

## Step 8) Confirm access

Point the browser to port 7000 or RDP to port 7001 to confirm FSHARK2 is accessible



FSHARK2 in Browser

Login to frradio100v2-229U-fshark2



*Just connecting*

Session:

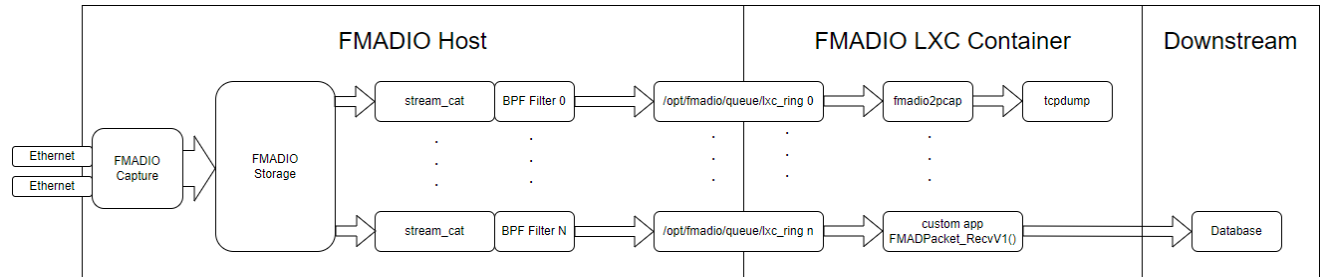
username:

password:

FSHARK2 via RDP

# Getting Packets

FMADIO LXC system high level architecture is shown below.



FMADIO LXC Architecture

This provides a full lossless filtered version of data directly into the LXC container for processing.

Backpressure is provided thru the LXC Ring structure, allowing the application to consume data as fast or slow as possible without dropping anything.

## Sending Data to LXC Offline

During development its typically easiest to send data into the LXC container in a one time replay operation. This allows sending the same data repeatedly to the application allow the engineers to debug and test the code.

In this example we will use the application "fmadio2pcap" which converts the LXC ring into a standard PCAP formatted stream of data.

Its recommended to start the data consumer aka LXC application first, then start the replay.

### Consumer - fmadio2pcap tcpdump

Run the following on the LXC container to generate TCPDUMP output thru the lxc ring named "general"

```
/opt/fmadio/platform/fmadio2pcap -i /opt/fmadio/queue/lxc_ring_general | tcpdump -r - -nn
```

It will look similar to the following on startup, it stops because no packets have been sent to the ring.

```

root@fmadio40v3-440-centos7:/opt/fmadio/platform/fmadio2pcap# ./fmadio2pcap -i /opt/fmadio/
fmadio2pcap
FMAD Ring [/opt/fmadio/queue/lxc_ring_general]
Ring size : 12595200 12595200 16777216
Ring Version: 100 100
Ring Depth : 400 400
Ring Mask : 3ff 3ff
RING: Put:1612f 12f
RING: Get:1612f 12f

```

## Producer - stream\_cat sending to LXC ring

On the FMADIO Host system find a capture you want to replay. In this example we are using capture named "inetsample\_20230615\_1513" filename can be found using "stream\_dump"

The command is as follows

```

sudo stream_cat --ring /opt/fmadio/queue/lxc_ring_general --ring-filter-bpf /opt/fmadio/que

```

Example output of the command shown below

```

fmadio@fmadio40v3-440:/mnt/store0/tmp2$ sudo stream_cat --ring /opt/fmadio/queue/lxc_ring_ge
Create FMAD Ring:0 [/opt/fmadio/queue/lxc_ring_general]
found ring: [/opt/fmadio/queue/lxc_ring_general] id:0
FMAD Ring:0 [/opt/fmadio/queue/lxc_ring_general] FilterBPF[icmp]
RING[/opt/fmadio/queue/lxc_ring_general] 00 : CPU: 0 FilterBPF:[icmp] Filt
StartChunkID: 766030
StartChunk: 766030 Offset: 0 Stride: 1
StartChunk: 766030
RING[/opt/fmadio/queue/lxc_ring_general] Size : 12595200 16777216
RING[/opt/fmadio/queue/lxc_ring_general] Version: 100 100
RING[/opt/fmadio/queue/lxc_ring_general] Put:54a 14a 0x7f30c8cc1000
RING[/opt/fmadio/queue/lxc_ring_general] Get:54a 14a 0x7f30c8cc2000
RING[/opt/fmadio/queue/lxc_ring_general] thread:0
RING[/opt/fmadio/queue/lxc_ring_general] worker thread start
0M Offset: 0GB ChunkID:766030 TS:00:00:00.000.000.000 | Pending 31590 MB 0.000Gbps 0.000
2M Offset: 2GB ChunkID:776263 TS:00:00:00.000.000.000 | Pending 29032 MB 21.164Gbps 2.31
4M Offset: 4GB ChunkID:786596 TS:00:00:00.000.000.000 | Pending 26449 MB 21.374Gbps 2.30
6M Offset: 7GB ChunkID:796886 TS:00:00:00.000.000.000 | Pending 23876 MB 21.283Gbps 2.31
.
.
.

```

And on the LXC client side the output will show ICMP packets printed in tcpdump format as shown below

```

root@fmadio40v3-440-centos7:/opt/fmadio/platform/fmadio2pcap# ./fmadio2pcap -i /opt/fmadio/
fmadio2pcap
FMAD Ring [/opt/fmadio/queue/lxc_ring_general]
Ring size : 12595200 12595200 16777216
Ring Version: 100 100
Ring Depth : 400 400
Ring Mask : 3ff 3ff
RING: Put:29c1 1c1
RING: Get:29c1 1c1
reading from file -, link-type EN10MB (Ethernet)
13:39:53.729815 IP 130.128.35.9 > 210.175.175.21: ICMP echo request, id 4494, seq 1, length 64
13:39:53.731604 IP 210.175.175.21 > 130.128.35.9: ICMP echo reply, id 4494, seq 1, length 64
13:39:53.732577 IP 45.0.191.213 > 8.8.8.8: ICMP echo request, id 13542, seq 25606, length 40
13:39:53.733466 IP 202.17.221.191 > 216.58.200.196: ICMP host 202.17.221.191 unreachable, le
13:39:53.733501 IP 202.17.221.191 > 216.58.200.196: ICMP host 202.17.221.191 unreachable, le
13:39:53.733641 IP 8.8.8.8 > 45.0.191.213: ICMP echo reply, id 13542, seq 25606, length 40
13:39:53.734071 IP 130.128.255.176 > 103.235.46.39: ICMP echo request, id 65301, seq 2, leng
13:39:53.744961 IP 130.128.255.81 > 210.175.175.21: ICMP echo request, id 4498, seq 1, lengt
13:39:53.746434 IP 210.175.175.21 > 130.128.255.81: ICMP echo reply, id 4498, seq 1, length
13:39:53.762392 IP 12.133.183.36 > 45.0.191.123: ICMP echo reply, id 46807, seq 6777, length
13:39:53.770100 IP 45.0.252.242 > 8.8.8.8: ICMP echo request, id 6172, seq 0, length 64
13:39:53.771160 IP 8.8.8.8 > 45.0.252.242: ICMP echo reply, id 6172, seq 0, length 64
13:39:53.775954 IP 130.128.5.7 > 172.21.32.22: ICMP echo request, id 63039, seq 32022, lengt
13:39:53.776217 IP 130.128.5.7 > 172.21.32.21: ICMP echo request, id 63039, seq 32278, lengt
13:39:53.776490 IP 130.128.5.7 > 172.21.32.20: ICMP echo request, id 63039, seq 32534, lengt
13:39:53.776777 IP 130.128.5.7 > 172.21.32.19: ICMP echo request, id 63039, seq 32790, lengt
13:39:53.790153 IP 45.0.191.229 > 8.8.8.8: ICMP echo request, id 34054, seq 39636, length 44
13:39:53.800539 IP 45.0.191.122 > 8.8.8.8: ICMP echo request, id 8265, seq 44218, length 64
13:39:53.801850 IP 8.8.8.8 > 45.0.191.122: ICMP echo reply, id 8265, seq 44218, length 64
.
.
.

```

## Custom Application

For a custom application to directly ingest data, the following reference code is provided

Primary header file with all functions and structures inline

[https://github.com/fmadio/platform/blob/main/include/fmadio\\_packet.h](https://github.com/fmadio/platform/blob/main/include/fmadio_packet.h)

Core example code to retrieve packets

<https://github.com/fmadio/platform/blob/main/fmadio2pcap/main.c#L160-L185>

Core loop snippet, this converts from LXC Ring into standard PCAP Packet format

```

while (!s_Exit)
{
    u64 TS;
    PCAPPacket_t* Pkt      = (PCAPPacket_t*)PktBuffer;

    // fetch packet from ring without blocking
    int ret = FMADPacket_RecvV1(s_RING, false, &TS, &Pkt->LengthWire, &Pkt->LengthCapture);

    // if it has valid data
    if (ret > 0)
    {
        // convert 64b epoch into sec/subsec for pcap
        Pkt->Sec          = TS / (u64)1e9;
        Pkt->NSec         = TS % (u64)1e9;

        // write PCAP header and payload
        fwrite(PktBuffer, 1, sizeof(PCAPPacket_t) + Pkt->LengthCapture, FPCAP);

        // general stats
        TotalPkt         += 1;
        TotalByte        += ret;
    }

    // end of stream
    if (ret < 0) break;
}

```



# Troubleshooting

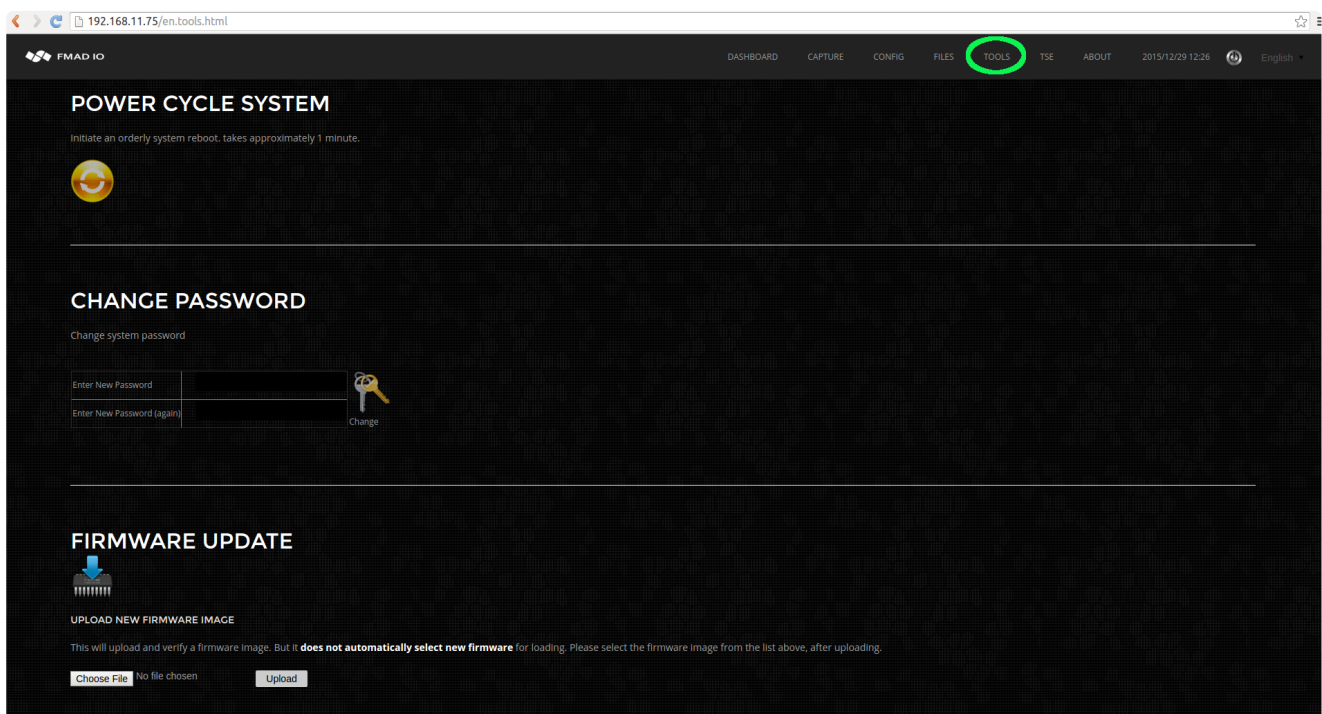
# Logfile Report (GUI)

The first step for any problem resolution is generating detailed log files for analysis to understand the exact nature of the problem. Our system automatically generates logfile information using the following steps.

NOTE: Typically logfile sizes are 200-500MB in size. It contains detailed system logs and may contain snippets of captured, packet data.

## 1) Tools page

Select the Tools menu highlighted in green below.



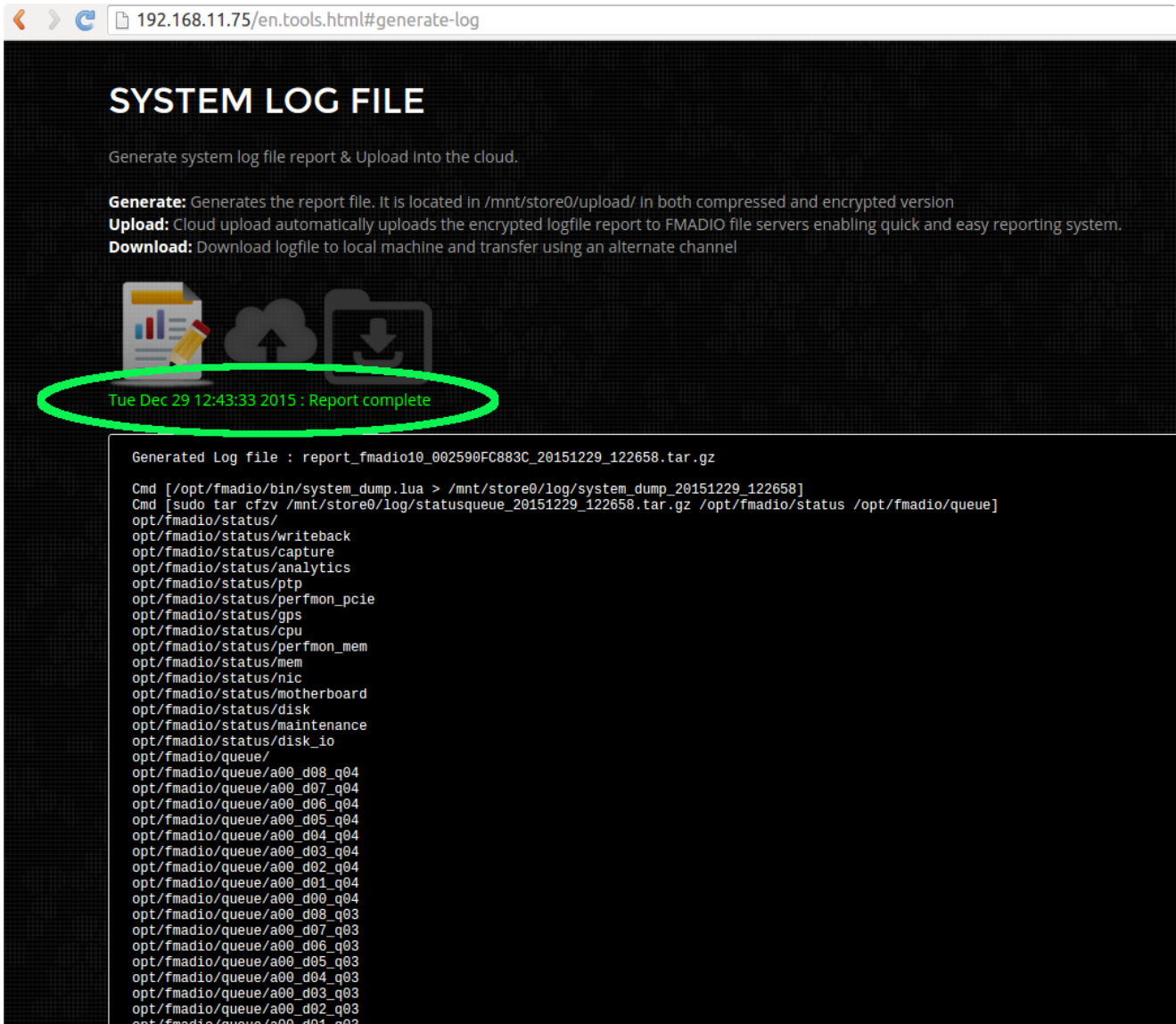
## 2) Start Log Generation

Start System Log generation, by clicking on the icon highlighted in green below.



#### 4) Log Generation Complete

When completed the status will change as highlighted in green below.



192.168.11.75/en.tools.html#generate-log

## SYSTEM LOG FILE

Generate system log file report & Upload into the cloud.

**Generate:** Generates the report file. It is located in /mnt/store0/upload/ in both compressed and encrypted version  
**Upload:** Cloud upload automatically uploads the encrypted logfile report to FMADIO file servers enabling quick and easy reporting system.  
**Download:** Download logfile to local machine and transfer using an alternate channel

Tue Dec 29 12:43:33 2015 : Report complete

```
Generated Log file : report_fmadio10_002590FC883C_20151229_122658.tar.gz
Cmd [/opt/fmadio/bin/system_dump.lua > /mnt/store0/log/system_dump_20151229_122658]
Cmd [sudo tar cfzv /mnt/store0/log/statusqueue_20151229_122658.tar.gz /opt/fmadio/status /opt/fmadio/queue]
opt/fmadio/status/
opt/fmadio/status/writeback
opt/fmadio/status/capture
opt/fmadio/status/analytics
opt/fmadio/status/ptp
opt/fmadio/status/perfmon_pcie
opt/fmadio/status/gps
opt/fmadio/status/cpu
opt/fmadio/status/perfmon_mem
opt/fmadio/status/mem
opt/fmadio/status/nic
opt/fmadio/status/motherboard
opt/fmadio/status/disk
opt/fmadio/status/maintenance
opt/fmadio/status/disk_io
opt/fmadio/queue/
opt/fmadio/queue/a00_d08_q04
opt/fmadio/queue/a00_d07_q04
opt/fmadio/queue/a00_d06_q04
opt/fmadio/queue/a00_d05_q04
opt/fmadio/queue/a00_d04_q04
opt/fmadio/queue/a00_d03_q04
opt/fmadio/queue/a00_d02_q04
opt/fmadio/queue/a00_d01_q04
opt/fmadio/queue/a00_d00_q04
opt/fmadio/queue/a00_d08_q03
opt/fmadio/queue/a00_d07_q03
opt/fmadio/queue/a00_d06_q03
opt/fmadio/queue/a00_d05_q03
opt/fmadio/queue/a00_d04_q03
opt/fmadio/queue/a00_d03_q03
opt/fmadio/queue/a00_d02_q03
opt/fmadio/queue/a00_d01_q03
```

#### 5) Download Log

You can now download the report via the icon highlighted in Green. An example downloaded log file is shown in blue below. After download, transfer to us for further analysis.

# SYSTEM LOG FILE

Generate system log file report & Upload into the cloud.

**Generate:** Generates the report file. It is located in /mnt/store0/upload/ in both compressed and encrypted version

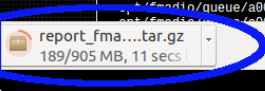
**Upload:** Cloud upload automatically uploads the encrypted logfile report to FMADIO file servers enabling quick and easy reporting system.

**Download:** Download logfile to local machine and transfer using an alternate channel



Tue Dec 29 12:43:33 2015 : Report complete

```
Generated Log file : report_fmadio10_002590FC883C_20151229_122658.tar.gz
Cmd [/opt/fmadio/bin/system_dump.lua > /mnt/store0/log/system_dump_20151229_122658]
Cmd [sudo tar cfzv /mnt/store0/log/statusqueue_20151229_122658.tar.gz /opt/fmadio/status /opt/fmadio/queue/
opt/fmadio/status/writeback
opt/fmadio/status/capture
opt/fmadio/status/analytics
opt/fmadio/status/ptp
opt/fmadio/status/perfmon_pci
opt/fmadio/status/gps
opt/fmadio/status/cpu
opt/fmadio/status/perfmon_mem
opt/fmadio/status/mem
opt/fmadio/status/nic
opt/fmadio/status/motherboard
opt/fmadio/status/disk
opt/fmadio/status/maintenance
opt/fmadio/status/disk_io
opt/fmadio/queue/
opt/fmadio/queue/a00_d08_q04
opt/fmadio/queue/a00_d07_q04
opt/fmadio/queue/a00_d06_q04
opt/fmadio/queue/a00_d05_q04
opt/fmadio/queue/a00_d04_q04
opt/fmadio/queue/a00_d03_q04
opt/fmadio/queue/a00_d02_q04
opt/fmadio/queue/a00_d01_q04
opt/fmadio/queue/a00_d00_q04
opt/fmadio/queue/a00_d08_q03
opt/fmadio/queue/a00_d07_q03
opt/fmadio/queue/a00_d06_q03
opt/fmadio/queue/a00_d05_q03
opt/fmadio/queue/a00_d04_q03
opt/fmadio/queue/a00_d03_q03
opt/fmadio/queue/a00_d02_q03
opt/fmadio/queue/a00_d01_q03
opt/fmadio/queue/a00_d00_q03
opt/fmadio/queue/a00_d08_q02
opt/fmadio/queue/a00_d07_q02
opt/fmadio/queue/a00_d06_q02
opt/fmadio/queue/a00_d05_q02
opt/fmadio/queue/a00_d04_q02
opt/fmadio/queue/a00_d03_q02
opt/fmadio/queue/a00_d02_q02
opt/fmadio/queue/a00_d01_q02
opt/fmadio/queue/a00_d00_q02
opt/fmadio/queue/a00_d08_q01
opt/fmadio/queue/a00_d07_q01
opt/fmadio/queue/a00_d06_q01
opt/fmadio/queue/a00_d05_q01
opt/fmadio/queue/a00_d04_q01
opt/fmadio/queue/a00_d03_q01
opt/fmadio/queue/a00_d02_q01
opt/fmadio/queue/a00_d01_q01
opt/fmadio/queue/a00_d00_q01]
```



# Logfile Report (CLI)

Sometimes GUI logfile generation is more difficult than a CLI version. As the GUI calls a script on the system to generate the logfile report, SSHing into the fmadio packet capture system and running the script directly is possible. The command to run in as follows

```
fmadio@fmadio20-049:/mnt/store0/upload$ sudo /opt/fmadio/bin/syslog_report.lua
```

```
fmadio@fmadio20-049:/mnt/store0/upload$ sudo /opt/fmadio/bin/syslog_report.lua
fmad fmadlua Dec 22 2015
calibrating...
0 : 00000000d09dad48          3.5000 cycles/nsec
Cycles/Sec 3499994440.0000 Std:      0cycle std( 0.00000000)
loading filename [/opt/fmadio/bin/syslog_report.lua]
Cmd [/opt/fmadio/bin/system_dump.lua > /mnt/store0/log/system_dump_20151229_132103]
loading filename [/opt/fmadio/bin/system_dump.lua]
[      iosched_direct.stdouterr_20151229]          1283855          1 MB
[      iosched_direct_20151229_1205]             1365723          2 MB
[      monitor_gps_20151229_1205]                9834318         12 MB
[      monitor_memory_20151229_1205]             809724          13 MB
[      monitor_nic_20151229_1205]                1179945         14 MB
[      statusqueue_20151229_132103.tar.gz]         40916           14 MB
[      stream_capture_sf20_20151229_1205]         288414          14 MB
[      monitor_cpu_20151229_1205]                642415          15 MB
[      scheduler_20151229_1205]                  404614          15 MB
[      sfptp_stats]                               3276884         19 MB
[      stream_writeback.stdouterr_20151229]       973105          20 MB
[      stream_writeback_20151229_1205]           1054488         21 MB
[      system_dump_20151229_132103]              1089180         22 MB
[      monitor_ptp.lua.stdouterr_20151229]        22197           22 MB
[      monitor_ptp_20151229_1205]                676222         23 MB
[      analytics.lua.stdouterr_20151229]          30954           23 MB
.
.
.
.
.
.
```

After the report has completed, the final log file is located at

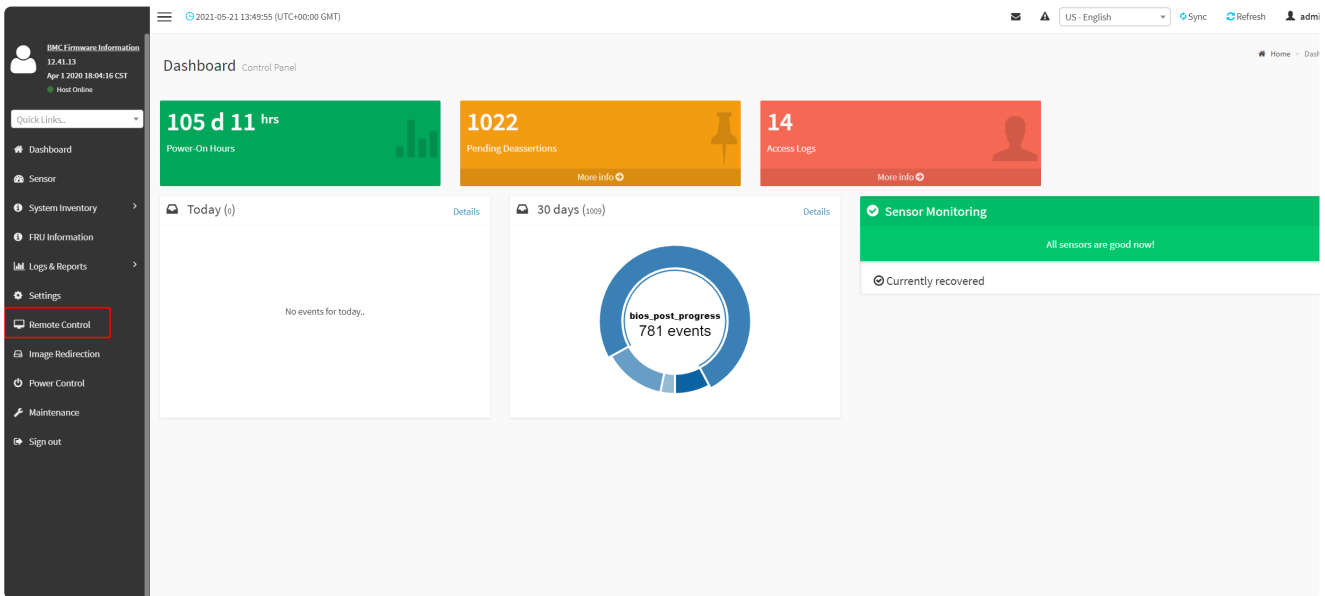
```
/mnt/store0/upload/report_*.tar.gz
```

Please scp off the device and send to support,

# System Recovery

In the unlikely event of a complete boot failure, system can be recovered by booting via the Virtual CDROM interface over a HTML BMC connection

Start by going to the BMC interface (default IP is 192.168.0.93) contact us for default login/password

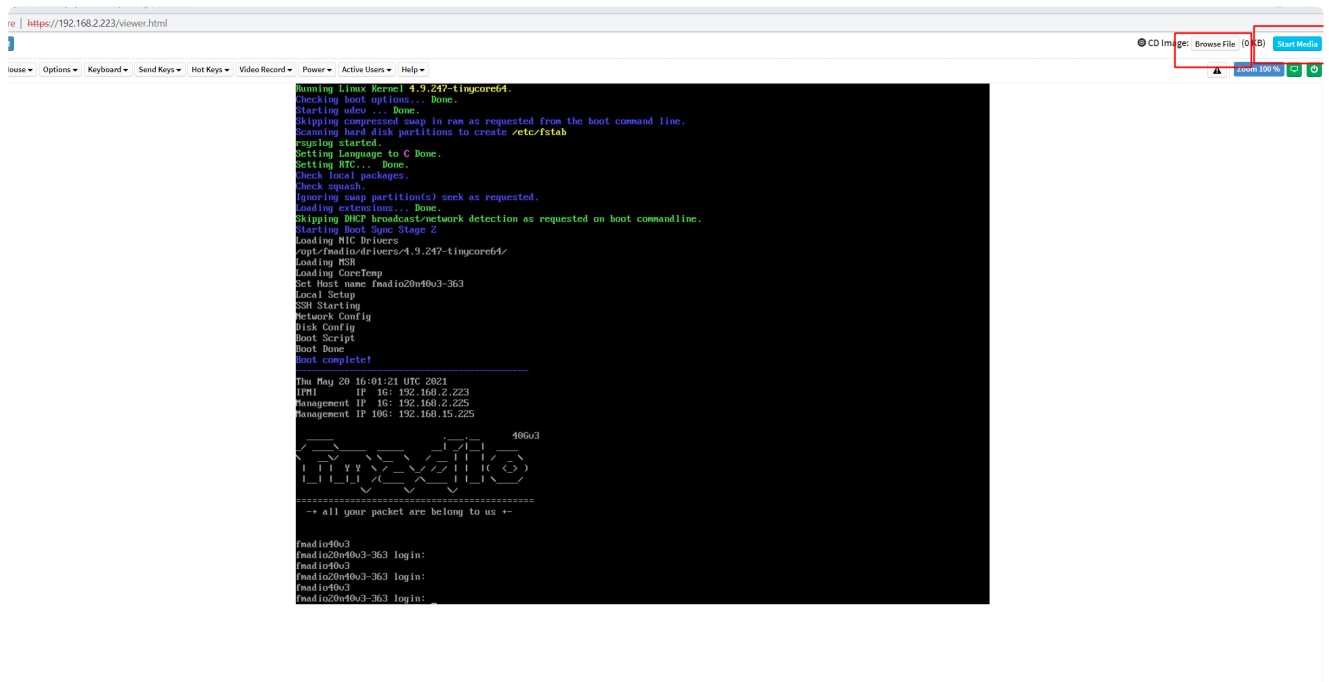


Start the Remote HTML KVM



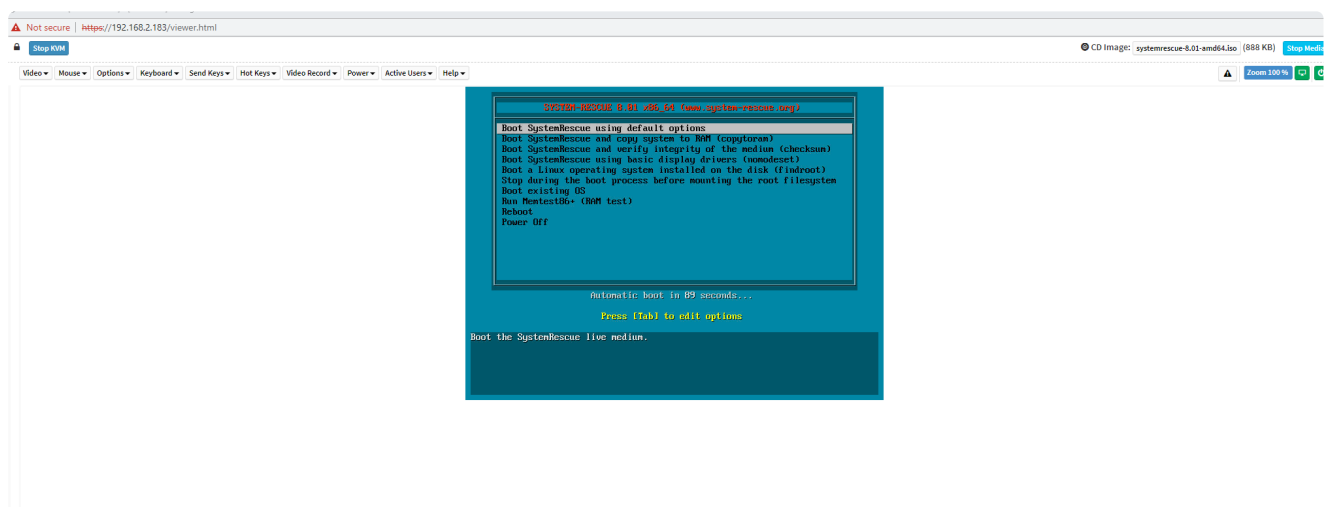
Will look like this. Select Brose Files, selecting an ISO image + Start the Media





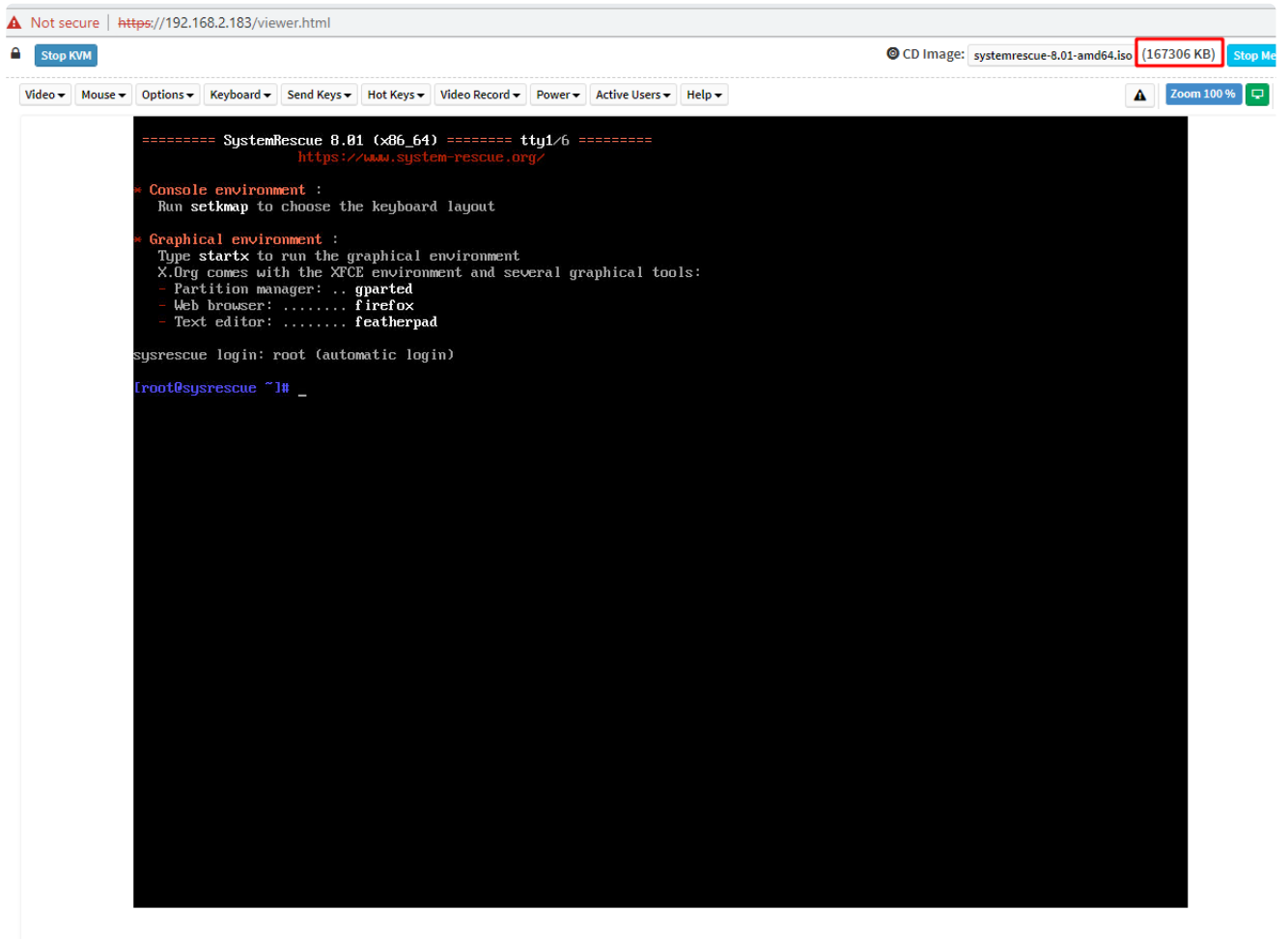
System will boot Ubuntu (for example), we are using ( systemrescue 8.01 amd64)

<https://sourceforge.net/projects/systemrescuecd/files/sysresccd-x86/8.01/systemrescue-8.01-amd64.iso/download>



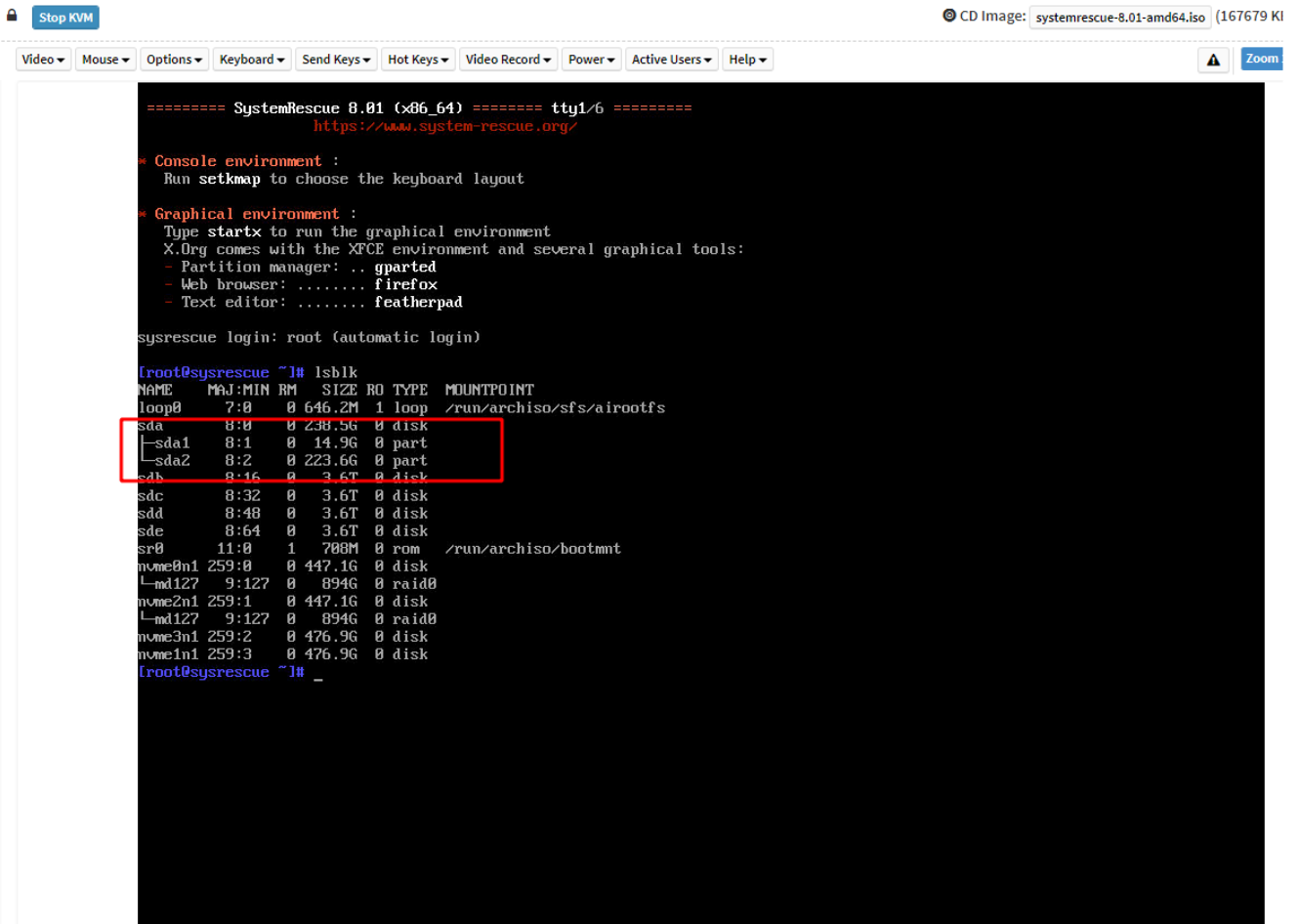
System will boot as follows, it may take several minutes depending on the speed of the HTML <-> FMADIO System connection. Recommend the closer the HTML instance is to the FMAD device the better.

If a particular boot stage is taking too long Ctrl-C can skip it



After SystemRescue CD has booted, the above is seen. Note the total number of bytes transferred over the Virtual ISO.

First step is to find the FMADIO OS and Persistent storage devices, Use the "lsblk" tool



Looking for a small (15GB) partition as the OS boot disk. In this case its sda1 and a large (224GB or larger) partition for the Persistent storage

Sometimes its easier to work over SSH. To access the system find or assign an IP address to the a reachable interface

SystemRescueCD by default has iptables setup. Disable all iptables as follows

```
iptables -F
iptables -X
systemctl stop iptables
```

Then setup a password for the root account

```
passwd
```

Then ssh access to the system is possible

```
aaron@ingress:~$ ssh root@192.168.2.121
The authenticity of host '192.168.2.121 (192.168.2.121)' can't be established.
ECDSA key fingerprint is SHA256:v2CQjmUL70YpMJh39GWhcyqanKUU4eqLXxjTg/2i35Q.
Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '192.168.2.121' (ECDSA) to the list of known hosts.
root@192.168.2.121's password:
[root@sysrescue ~]#
```

Next mount the FMAD OS and Persistent storage disks. They may be sda\* or nvme0n1p\* in this example its mapped to sda

```
[root@sysrescue ~]# cd /mnt
[root@sysrescue /mnt]# mkdir system
[root@sysrescue /mnt]# mkdir store0
[root@sysrescue /mnt]# mount /dev/sda1 system/
[root@sysrescue /mnt]# mount /dev/sda2 store0/
[root@sysrescue /mnt]#
```

Next check the contents, it should look roughly like this

```
[root@sysrescue /mnt]# ls -al /mnt/system/
total 64
drwxr-xr-x 5 root root 8192 Jan 1 1970 .
drwxr-xr-x 1 root root 80 May 22 08:53 ..
drwxr-xr-x 3 root root 8192 Apr 18 15:16 boot
drwxr-xr-x 2 root root 40960 Apr 18 15:16 firmware
drwxr-xr-x 5 root root 8192 May 11 10:13 tce
[root@sysrescue /mnt]# ls -al /mnt/store0
total 17244
drwxrwxrwx 32 root root 4096 May 11 08:32 .
drwxr-xr-x 1 root root 80 May 22 08:53 ..
drwxr-xr-x 3 root root 4096 May 11 11:29 etc
drwxr-xr-x 2 root root 4096 Dec 17 2019 filter
drwxrwxrwx 4 root root 17477632 May 22 08:37 log
drwx----- 2 root root 16384 Dec 16 2019 lost+found
drwxr-xr-x 6 1002 games 4096 Oct 12 2019 lxc
drwxr-xr-x 3 1002 games 4096 Aug 5 2020 plugin_data
drwxr-xr-x 2 root root 4096 Dec 17 2019 stream
drwx----- 4 nobody root 4096 Dec 29 03:23 tmp
drwxrwxrwx 10 root root 4096 Mar 23 10:25 tmp2
[root@sysrescue /mnt]#
```



# KVM Access via IPMI

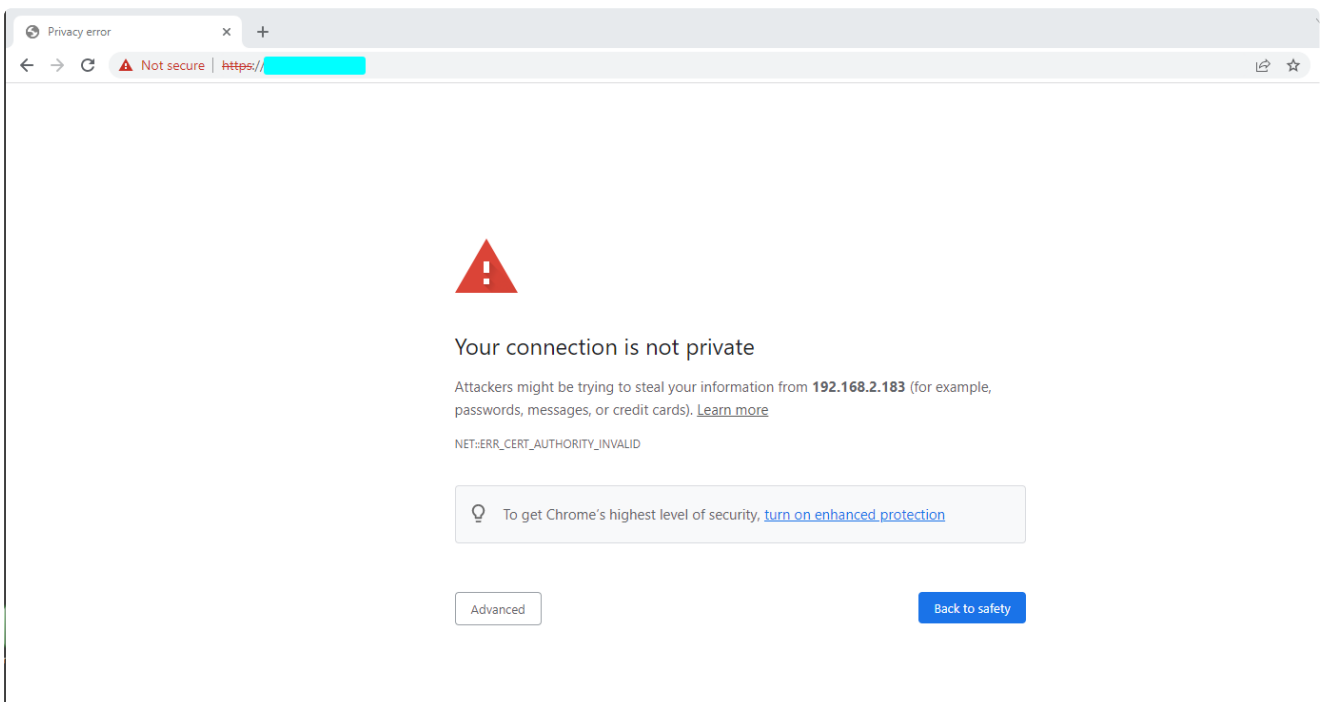
FMADIO systems include a full full Keyboard Video Mouse (KVM) remote access using the onboard AST2500/AST2600 BMC chipset. This enables remote power cycles, reboots and others all using a simple HTTPS HTML5 web browser (No requirement for Java)

To log into the system via the KVM please run the following steps

## STEP 1

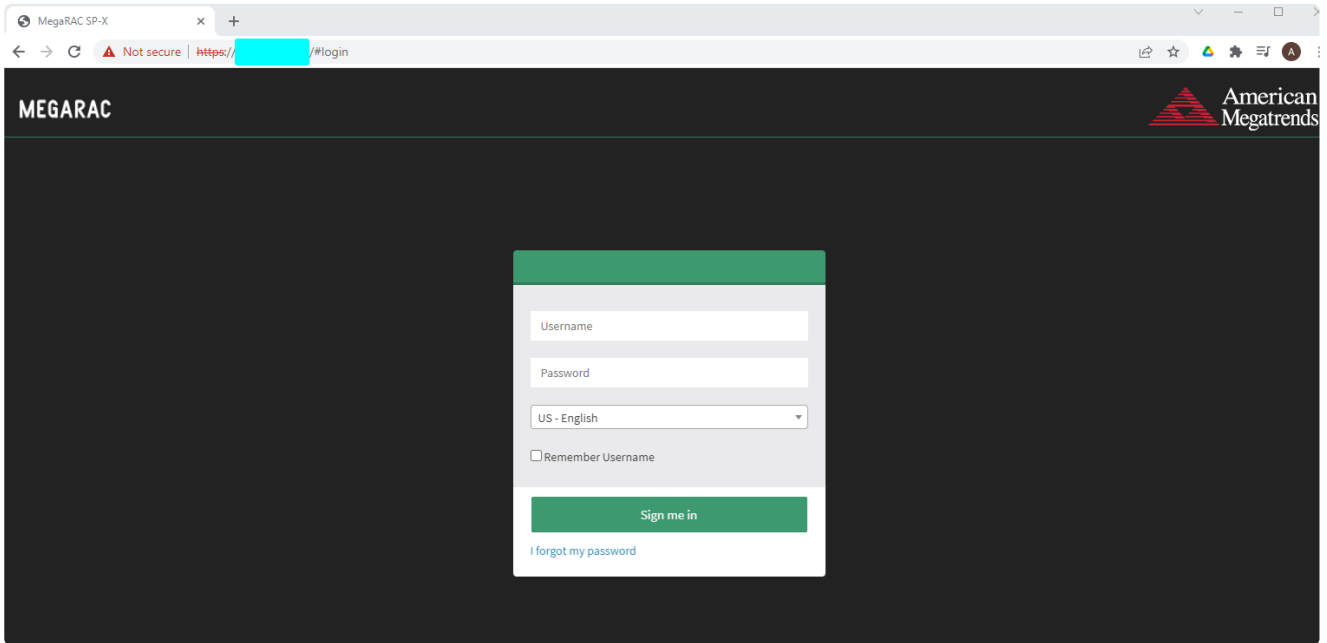
Log into the IPMI/BMC system, please contact [support@fmad.io](mailto:support@fmad.io) for default IP and credentials.

It uses a self signed certificate unless a certified certificate has been uploaded. Please click thru to the login page



Self Signed Certificate

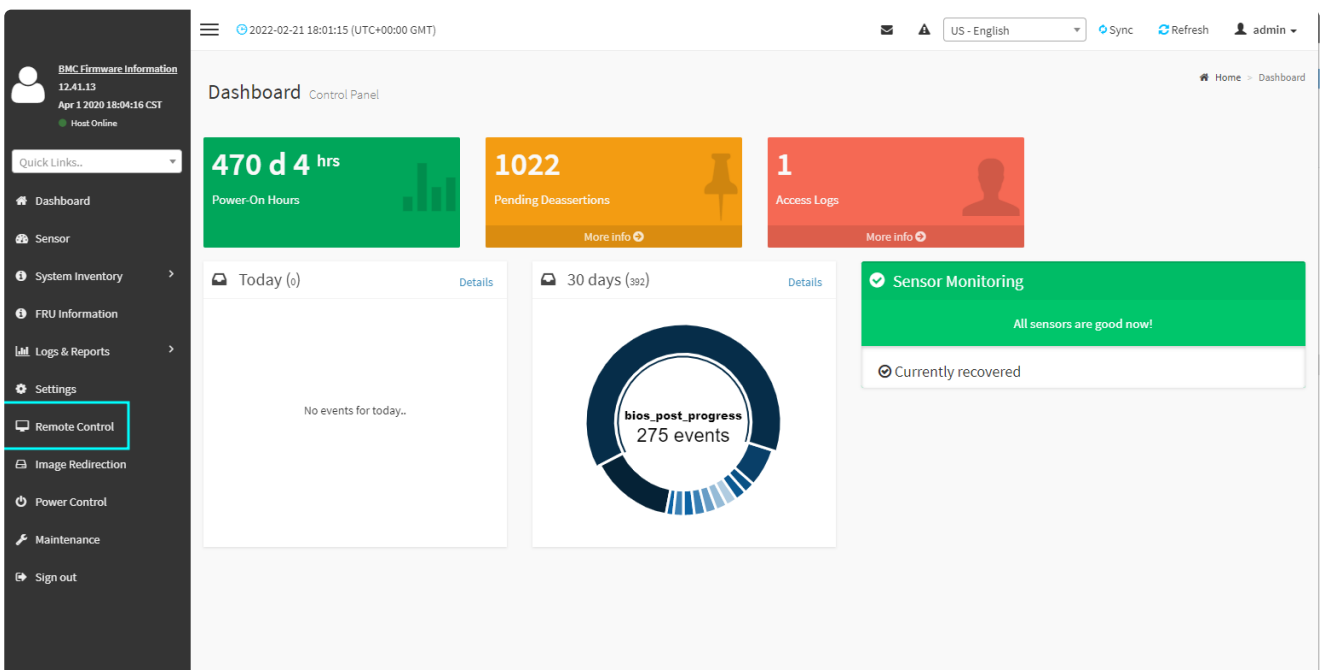
Login page looks like the following. Please contact [support@fmad.io](mailto:support@fmad.io) for default credentials



IPMI BMC Login page

## STEP 2

Select "Remote Control" on the left menu

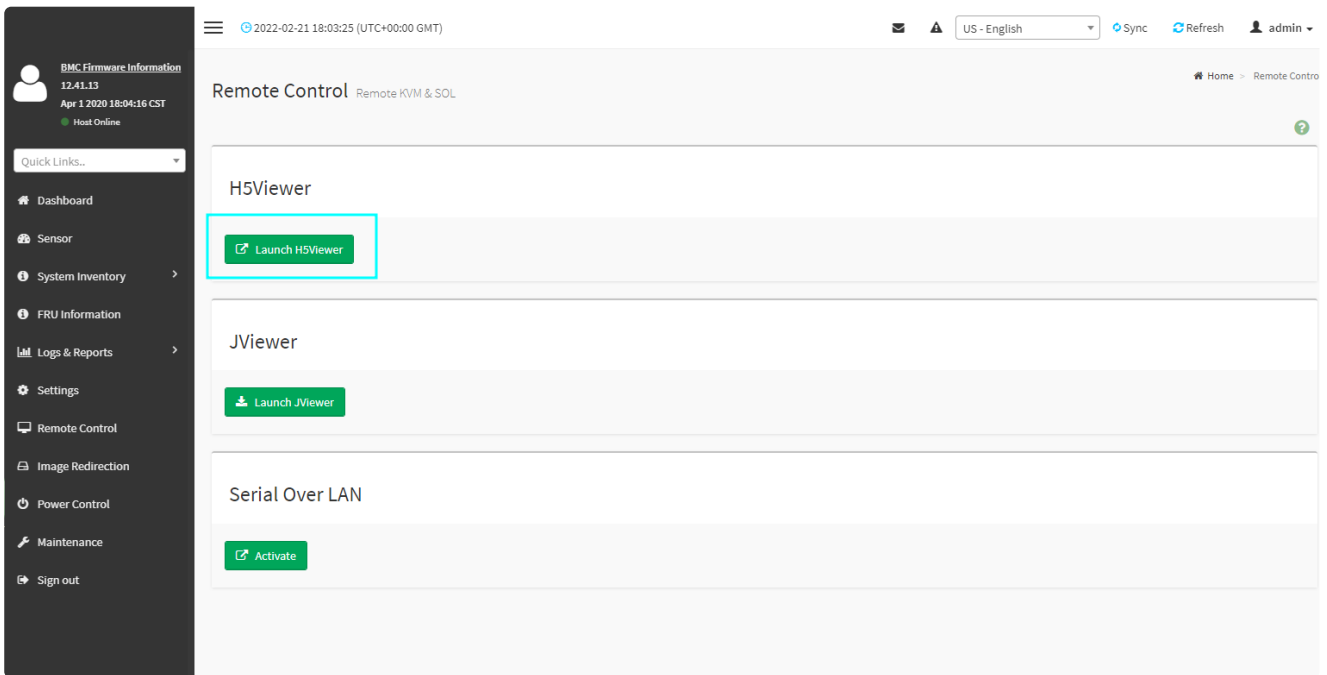


FMADIO IPMI Remote Control

## STEP 3

Select "H5Viewer" to launch the web HTML KVM page.

NOTE: this uses normal HTTPS connection no additional firewall rules or ports need to be opened



FMADIO HTML5 KVM

#### STEP 4

HTML5 KVM viewer is launched. Looks like the following below. From here please login to the FMADIO device to setup and configuration.

Due to screen saver, it may appear blank. Please press any key to disable the screen saver.

NOTE: the currently configured IPs are shown at boot.



Remote KVM [redacted] - [1024 x 768] - Google Chrome

Not secure | https://[redacted]/viewer.html

Stop KVM CD Image: Browse File (0 KB) Start Media

Video Mouse Options Keyboard Send Keys Hot Keys Video Record Power Active Users Help

Zoom 100%

```
Loading MSR
Loading CoreTemp
Set Host name fmadio20u3-287
Local Setup
SSH Starting
Network Config
Disk Config
Boot Script
Boot Done
Boot complete!

-----
Fri Feb 18 13:09:19 UTC 2022
IPMI IP 1G: 192.168 [redacted]
Management IP 1G: 192.168 [redacted]
Management IP 10G: 192.168 [redacted]

20Gu3
-----
-- all your packet are belong to us +-
-----

fmadio20u3
fmadio20u3-287 login:
fmadio20u3
fmadio20u3-287 login:
fmadio20u3
fmadio20u3-287 login: fmadio
Password:

20Gu3
-----
-- all your packet are belong to us +-
-----
Error: Failed to call git rev-parse --git-dir --show-toplevel: "fatal: not a git repository (or any parent up to mount point /mnt)
Stopping at filesystem boundary (GIT_DISCOVERY_ACROSS_FILESYSTEM not set).\n"
Git LFS initialized.
Error: Failed to call git rev-parse --git-dir --show-toplevel: "fatal: not a git repository (or any parent up to mount point /mnt)
Stopping at filesystem boundary (GIT_DISCOVERY_ACROSS_FILESYSTEM not set).\n"
Git LFS initialized.
fmadio@fmadio20u3-287:~$
```

LWIN RWIN LALT LCTRL RALT RCTRL NUM CAPS SCR

# Serial Port Access over LAN

FMADIO packet capture systems all have serial port either virtual or physical connectors. Typically serial port access over LAN is the preferred method as it provides an out of bands interface to the system that only requires a SSH terminal (No KVM or HTM or Java).

Serial port access is archived using the IPMITOOL located at the following link:

<https://github.com/ipmitool/ipmitool>

We are using IPMITOOL 1.8.16 in this example

```
aaron@ingress:~$ ipmitool -V  
ipmitool version 1.8.16
```

Start by checking the system status via IPMITOOL with the sdr argument. Contact support for the default password and replace the IP address (192.168.1.123) with the IP address of the BMC device

```
aaron@ingress:~$ ipmitool -U admin -P XXXXXXX -H 192.168.1.123 sdr
```

Watchdog	0x00	ok
SEL	0x00	ok
BPB_FAN_1A	18750 RPM	ok
BPB_FAN_1B	15000 RPM	ok
BPB_FAN_2A	18600 RPM	ok
BPB_FAN_2B	15000 RPM	ok
BPB_FAN_3A	18300 RPM	ok
BPB_FAN_3B	15000 RPM	ok
BPB_FAN_4A	18600 RPM	ok
BPB_FAN_4B	15000 RPM	ok
BPB_FAN_5A	18600 RPM	ok
BPB_FAN_5B	15000 RPM	ok
BPB_FAN_6A	18450 RPM	ok
BPB_FAN_6B	14850 RPM	ok
BPB_FAN_7A	18600 RPM	ok
BPB_FAN_7B	15000 RPM	ok
BPB_FAN_8A	18600 RPM	ok
BPB_FAN_8B	14850 RPM	ok
CPU0_Status	0x00	ok
CPU1_Status	0x00	ok
CPU0_TEMP	34 degrees C	ok
CPU1_TEMP	33 degrees C	ok
DIMMG0_TEMP	28 degrees C	ok
DIMMG1_TEMP	29 degrees C	ok
DIMMG2_TEMP	29 degrees C	ok
DIMMG3_TEMP	29 degrees C	ok
CPU0_DTS	61 degrees C	ok
CPU1_DTS	62 degrees C	ok
HDD_TEMP_0	22 degrees C	ok
HDD_TEMP_1	22 degrees C	ok
INLET_AIR_TEMP	24 degrees C	ok
MB_TEMP1	29 degrees C	ok
MB_TEMP2	31 degrees C	ok
NVMeG0_TEMP	no reading	ns
NVMeG1_TEMP	no reading	ns
NVMeG2_TEMP	29 degrees C	ok
NVMeG3_TEMP	no reading	ns
NVMeG4_TEMP	no reading	ns
NVMeG5_TEMP	no reading	ns
PCH_TEMP	34 degrees C	ok
PS1_Status	0x00	ok
PS2_Status	0x00	ok
PSU1_HOTSPOT	no reading	ns
PSU2_HOTSPOT	32 degrees C	ok
P_12V	12.03 Volts	ok
P_1V0_AUX_LAN	0.99 Volts	ok
P_1V05_PCH	1.02 Volts	ok
P_1V8_AUX_PCH	1.78 Volts	ok
P_3V3	3.24 Volts	ok
P_5V	4.93 Volts	ok
P_5V_STBY	4.91 Volts	ok

```
P_VBAT          | 2.93 Volts      | ok
P_VCCIN_CPU0    | 1.75 Volts      | ok
P_VCCIN_CPU1    | 1.75 Volts      | ok
P_VDDQ_CPU0_ABC | 1.20 Volts      | ok
P_VDDQ_CPU0_DEF | 1.20 Volts      | ok
P_VDDQ_CPU1_ABC | 1.20 Volts      | ok
P_VDDQ_CPU1_DEF | 1.20 Volts      | ok
P_VNN_PCH_AUX   | 0.97 Volts      | ok
SYS_POWER       | 350 Watts       | ok
aaron@ingress:~$
```

In the above we see the sensor status of the FMADIO device, thus the tool is connecting to the device and BMC interface is working correctly.

To connect to the serial port use the following command

```
ipmitool -U admin -P <password> -H <host IP address> -I lanplus sol activate
```

Example as below

```
aaron@ingress:~$ ipmitool -U admin -P XXXX -H 192.168.1.123 -I lanplus sol activate
[SOL Session operational. Use ~? for help]

fmadio100v2
fmadio100v2-228U login:
fmadio100v2
fmadio100v2-228U login:
```

And you now have full serial port access to the system.

## Troubleshooting

The serial port is mutually exclusive access, e.g.. only one user at a time can access it. If a previous session is still connected the following will be shown

```
aaron@ingress:~$ ipmitool -U admin -P XXXXX -H 192.168.1.123 -I lanplus sol activate
Info: SOL payload already active on another session
aaron@ingress:~$
```

In such cases, please disconnect the previous session as follows

```
aaron@ingress:~$ ipmitool -U admin -P XXXXXX -H 192.168.1.123 -I lanplus sol deactivate
aaron@ingress:~$
```

Then reconnect

```
aaron@ingress:~$ ipmitool -U admin -P XXXXXX -H 192.168.1.123 -I lanplus sol activate  
[SOL Session operational. Use ~? for help]
```

```
fmadio100v2
```

```
fmadio100v2-228U login:
```

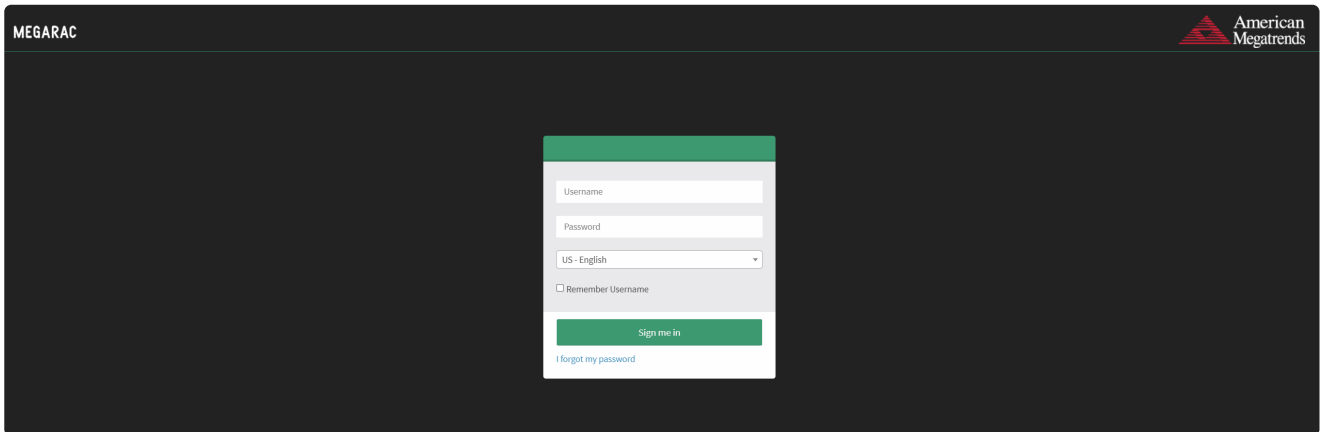
```
fmadio100v2
```

```
fmadio100v2-228U login:
```

# BMC Reset

All FMADIO Packet Capture systems have an onboard BMC, this allows remote power on/off and HTML5 based KVM utility which is very helpful when troubleshooting issues. In addition the BMC provides temperature voltage, current sensors and many other functions.

The chipset is an ASPEED2500 ( [http://www.aspeedtech.com/server\\_ast2500/](http://www.aspeedtech.com/server_ast2500/) ) which runs its own linux kernel + management software. Logging into the BMC via web interface looks like the following



FMADIO BMC Interface

There are times where the BMC after long uptime or other issues becomes unstable and requires a reboot

## BMC Reset via IPMITOOL

The easiest way to reset the BMC is via IPMITOOL. Start by confirming the BMC is up and responding by issuing the command

```
ipmitool -U admin -P <password> -H host.ip.address bmc info
```

Example output is shown below

```
aaron@ingress:~$ ipmitool -U admin -P secret -H 192.168.1.1 bmc info
Device ID           : 32
Device Revision     : 1
Firmware Revision   : 12.41
IPMI Version        : 2.0
Manufacturer ID     : 15370
Manufacturer Name   : Unknown (0x3C0A)
Product ID          : 308 (0x0134)
Product Name        : Unknown (0x134)
Device Available    : yes
Provides Device SDRs : yes
Additional Device Support :
    Sensor Device
    SDR Repository Device

    SEL Device
    FRU Inventory Device
    IPMB Event Receiver
    IPMB Event Generator
    Chassis Device
Aux Firmware Rev Info :
    0x0d
    0x00
    0x00
    0x00
aaron@ingress:~$
```

This shows the BMC is responding correctly, Next issue the BMC cold reset function using the command

```
ipmitool -U admin -P <password> -H host.ip.address bmc reset cold
```

Example run as follows

```
aaron@ingress:~$ ipmitool -U admin -P secret -H 192.168.1.1 bmc reset cold
Sent cold reset command to MC
aaron@ingress:~$
```

It will take a few minutes for the BMC Web interface and IPMITOOL tools to become active again.

**NOTE:** Please power cycle the Host system after BMC reboot to fully clear the error condition

## BMC Reset via SSH

If the above fails, a secondary option is possible using SSH into the BMC directly and issuing a reboot. Please contact [support@fmad.io](mailto:support@fmad.io) for password

```
$ ssh sysadmin@192.168.1.10
sysadmin@192.168.1.10's password:

sysadmin [~]# uptime
 07:23:13 up 21 days,  7:27,  1 users,  load average: 5.09, 5.03, 5.10

sysadmin [~]# reboot

The system is going down for reboot NOW!92509FE (pts/1) (Mon Mar 14 07:23:16
sysadmin [~]#
```

**NOTE:** Please power cycle the Host system after BMC reboot to fully clear the error condition. If a powercycle is not possible (due to BMC), a warm reboot can be run first.

## Power Disconnect

In some cases ipmitool is not responsive, thus the only option is a full power cable removal. Using the "power cycle" option on the BMC only resets the x86 server, it does not reboot the BMC. This is a last resort as physical access to the machine can be difficult in some cases.



# BMC Upgrade

## BMC Version 12.41.13

Upgrading the BMC software is simple process but requires physical access to the system. Physical access is required as the power cables need to be disconnected after BMC upgrade has been completed.

### Step 1) Version Check

Retrieve the current BMC version as follows

```
sudo ipmitool bmc info
```

Example output

```
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$ sudo ipmitool bmc info
Device ID                : 32
Device Revision          : 1
Firmware Revision        : 12.41
IPMI Version             : 2.0
Manufacturer ID          : 15370
Manufacturer Name        : Unknown (0x3C0A)
Product ID               : 308 (0x0134)
Product Name             : Unknown (0x134)
Device Available         : yes
Provides Device SDRs    : yes
Additional Device Support :
  Sensor Device
  SDR Repository Device
  SEL Device
  FRU Inventory Device
  IPMB Event Receiver
  IPMB Event Generator
  Chassis Device
Aux Firmware Rev Info    :
  0x0b
  0x00
  0x00
  0x00
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$
```

The key value difference is shown below, the firmware gets upgraded to version 0xd shown below. If the system already shows version 0xd there is no need to upgrade the BMC software

```
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$ sudo ipmitool bmc info
Device ID : 32
Device Revision : 1
Firmware Revision : 12.41
IPMI Version : 2.0
Manufacturer ID : 15370
Manufacturer Name : Unknown (0x3C0A)
Product ID : 308 (0x0134)
Product Name : Unknown (0x134)
Device Available : yes
Provides Device SDRs : yes
Additional Device Support :
  Sensor Device
  SDR Repository Device
  SEL Device
  FRU Inventory Device
  IPMB Event Receiver
  IPMB Event Generator
  Chassis Device
Aux Firmware Rev Info :
  0x0d
  0x00
  0x00
  0x00
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$
```

## Step 2) BMC Upgrade

Run the BMC update in the following directory

```
cd /opt/fmadio/firmware/bmc/
```

Then run the update program

```
sudo ./flash64.sh
```

Enter Y for preserve configuration settings.

The process will take several minutes to complete

```
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$ sudo ./flash64.sh
chmod: gigafash_x64: Read-only file system
chmod: socflash_x64: Read-only file system
gigafash v1.6.3
Do you want to preserve configuration? (Y/n)
Y
Loading Firmware...
Update Firmware
Wait 90 seconds for BMC Ready...
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$
```

## Step 3) Power disconnect

Disconnect the AC power from the system. Wait for 1 minute

## Step 4) Power Connect

Reconnect AC power to the system

## Step 5) Boot BMC

Wait 5minutes for the BMC to fully reboot and host system boot

## Step 6) Reboot host

Reboot the linux Host server

```
sudo reboot
```

## Step 7) Confirm update

After host linux system has rebooted, Check BMC version is updated. It should show version 0xD per image below

```
sudo ipmitool bmc info
```

```
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$ sudo ipmitool bmc info
Device ID           : 32
Device Revision    : 1
Firmware Revision  : 12.41
IPMI Version       : 2.0
Manufacturer ID    : 15370
Manufacturer Name  : Unknown (0x3C0A)
Product ID        : 308 (0x0134)
Product Name      : Unknown (0x134)
Device Available   : yes
Provides Device SDRs : yes
Additional Device Support :
  Sensor Device
  SDR Repository Device
  SEL Device
  FRU Inventory Device
  IPMB Event Receiver
  IPMB Event Generator
  Chassis Device
Aux Firmware Rev Info :
  0x0d
  0x00
  0x00
  0x00
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$
```

## Step 8) Done

Update is complete

## BMC Upgrade Version 12.61.01

BMC upgrade to version 12.60.39 resolves CVEs and makes possible boot without prompt when enabling full disk encryption.

From FW: 8940 the BMC firmware is located in

```
/opt/fmadio/firmware/bmc/
```

As the version jump from the factory installed to this version is very large, all BMC settings are lost during the upgrade, and need to be set again.

**This includes BMC Network information, meaning BMC network connectivity will be lost during this process**

In addition BMC passwords and other items are also lost. The way to update this is via the FMADIO x86 Host system, where the host is always powered on enabling it to set the BMC Network settings and User passwords directly.

### Step 1) setup UEFI on Boot partition

After updating the FMADIO Firmware the following files are located at

```
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$ ls -altr
total 0
drwxr-xr-x  3 fmadio  staff      60 Aug 11 15:41 ./
drwxr-xr-x  2 fmadio  staff     140 Aug 11 15:41 bmc_126101/
drwxr-xr-x  2 fmadio  staff     200 Aug 11 15:41 bios_r23/
drwxr-xr-x  3 fmadio  staff      60 Aug 11 15:41 EFI/
drwxr-xr-x  5 fmadio  staff     120 Aug 11 15:41 ./
lrwxrwxrwx  1 root    root       67 Aug 11 15:45 startup.nsh -> /tmp/tc1oop/fmadio100v2_current/opt/fmadio/firmware/bmc/startup.nsh
fmadio@fmadio100v2-228U:/opt/fmadio/firmware/bmc$
```

Copy the files as follows

```
sudo cp startup.nsh /mnt/system/
```

```
sudo cp -RL EFI /mnt/system/
```

NOTE: -L in the cp forces a literal copy (no sym links)

### Step 2) Flash BMC

Start the BMC update process using the following commands on the FMADIO host system

```
cd /opt/fmadio/firmware/bmc/bmc/
```

```
sudo ./bmc_fw_update_linux.sh
```

Example output shown below, it will take about 5minutes to run.

**Note the warning, Do you want preseve configuration, enter N**

```

fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bmc_126101$ sudo ./bmc_fw_update_linux.sh
gigaflash v2.0.7
Do you want to preserve configuration? (Y/N)
N
Loading Firmware...

Update Firmware
Find ASPEED Device 1a03:2000 on 4:0.0
MMIO Virtual Address: b1c1f000
Relocate IO Base: 1000
Found ASPEED Device 1a03:2500 rev. 41
Static Memory Controller Information:
CS0 Flash Type is SPI
CS1 Flash Type is SPI
CS2 Flash Type is SPI
CS3 Flash Type is NOR
CS4 Flash Type is NOR
Boot CS is 0
Option Information:
CS: 0
Flash Type: SPI
[Warning] Don't AC OFF or Reboot System During BMC Firmware Update!!
Find Flash Chip #1: 64MB SPI Flash
Update Flash Chip #1 O.K.
Update Flash Chip O.K.
Wait 90 seconds for BMC Ready...
fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bmc_126101$
fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bmc_126101$

```

### Step 3) BMC Version check

After the BMC is flashed and has rebooted confirm the new BMC version is 12.61.01 using the following command

```
sudo ipmitool bmc info
```

The output should look like the following

```

fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bmc_126101$ sudo ipmitool bmc info
Device ID           : 32
Device Revision     : 1
Firmware Revision   : 12.61
IPMI Version        : 2.0
Manufacturer ID     : 15370
Manufacturer Name   : Unknown (0x3C0A)
Product ID          : 308 (0x0134)
Product Name        : Unknown (0x134)
Device Available    : yes
Provides Device SDRs : yes
Additional Device Support :
  Sensor Device
  SDR Repository Device
  SEL Device
  FRU Inventory Device
  IPMB Event Receiver
  IPMB Event Generator
  Chassis Device
Aux Firmware Rev Info :
  0x01
  0x00
  0x00
  0x00
fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bmc_126101$

```

#### Step 4) BMC Network Config

Set the new network configuration information.

Set to use static ip per below

```
sudo ipmitool lan set 1 ipsrc static
```

```
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039# sudo ipmitool lan set 1 ipsrc static
```

Set a new IP address, netmask and gateway, replace addresses with the assigned BMC network address

```
sudo ipmitool lan set 1 ipaddr 192.168.2.173
```

```
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039# sudo ipmitool lan set 1 ipaddr 192.168.2.173
Setting LAN IP Address to 192.168.2.173
```

```
sudo ipmitool lan set 1 netmask 255.255.255.0
```

```
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039# sudo ipmitool lan set 1 netmask 255.255.255.0
Setting LAN Subnet Mask to 255.255.255.0
```

```
sudo ipmitool lan set 1 defgw ipaddr 192.168.2.254
```

```
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039# sudo ipmitool lan set 1 defgw ipaddr 192.168.2.254
Setting LAN Default Gateway IP to 192.168.2.254
```

At this point the network should be reachable via ping, however the username password will be reverted to the default setting.

Confirm the network settings using the command

```
sudo ipmitool lan print
```

Example output is shown below

```
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039# sudo ipmitool lan print
Set in Progress      : Set Complete
Auth Type Support    : NONE MD2 MD5 PASSWORD OEM
Auth Type Enable     : Callback : MD5
                    : User       : MD5
                    : Operator  : MD5
                    : Admin    : MD5
                    : OEM      : MD5
IP Address Source    : Static Address
IP Address           : 192.168.2.173
Subnet Mask          : 255.255.255.0
MAC Address          : eo:ds:se:50:21:56
SNMP Community String : AMI
IP Header            : TTL=0x40 Flags=0x40 Precedence=0x00 TOS=0x10
BMC ARP Control      : ARP Responses Enabled, Gratuitous ARP Disabled
Gratuitous ARP Intrvl : 10 seconds
Default Gateway IP   : 192.168.2.254
Default Gateway MAC   : 00:00:00:00:00:00
Backup Gateway IP    : 0.0.0.0
Backup Gateway MAC    : 00:00:00:00:00:00
802.1q VLAN ID       : Disabled
802.1q VLAN Priority : 0
RMCP+ Cipher Suites : 0,1,2,3,6,7,8,11,12,15,16,17
Cipher Suite Priv Max : caaaaaaaaaaXXX
                    : X=Cipher Suite Unused
                    : c=CALLBACK
                    : u=USER
                    : o=OPERATOR
                    : a=ADMIN
                    : O=OEM
Bad Password Threshold : 0
Invalid password disable: no
Attempt Count Reset Int.: 0
User Lockout Interval  : 0
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039#
```

The BMC webpage should be accessible at this point.

### Step 5) BMC User Password

The BMC update will delete all the settings, these need to be added back

First one is to set the admin password as follows, replacing "secret" with the password chosen

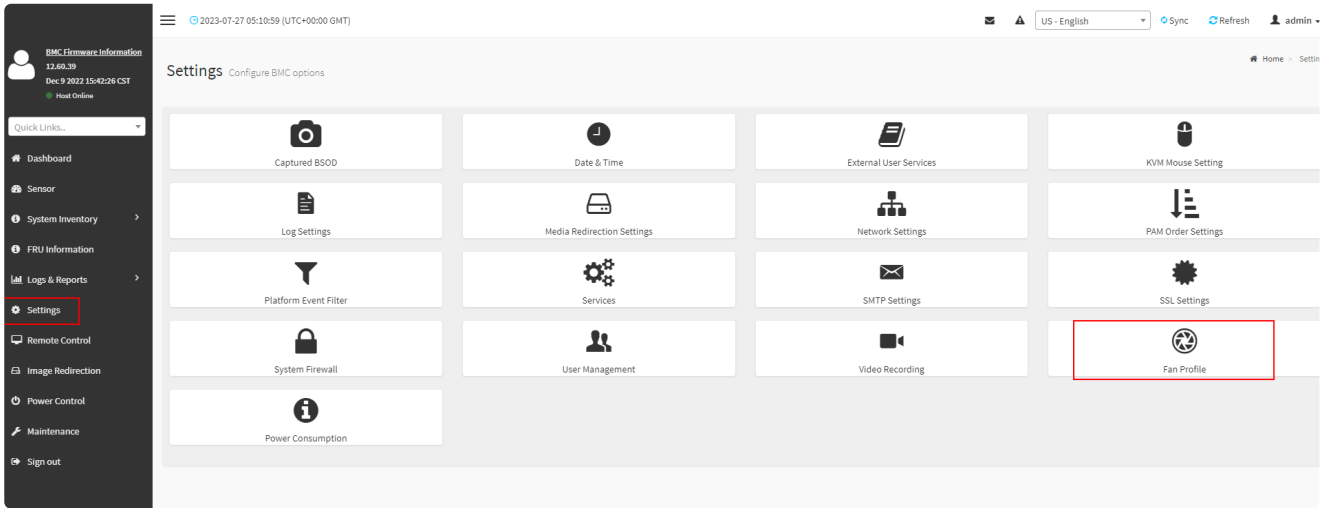
```
sudo ipmitool user set password 2 secret
```

```
root@fmadio100v2-228U:/mnt/store0/tools/bmc/126039# sudo ipmitool user set password 2 [REDACTED]
Set User Password command successful (user 2)
```

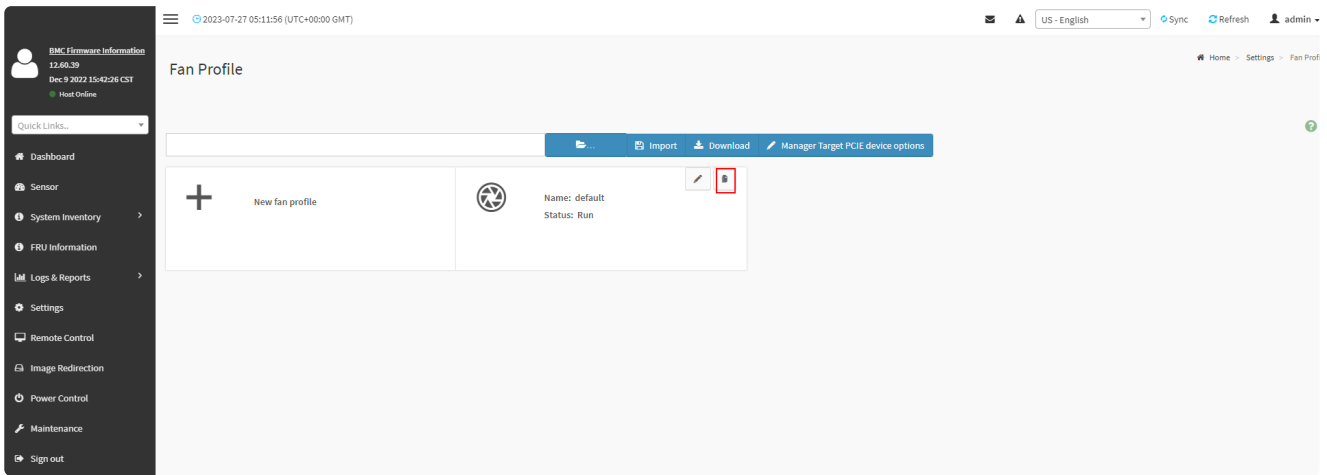
After setting this, logging into the BMC using the admin account and above password.

### Step 6) BMC Fan Profile

As all BMC settings are disabled, the first critical setting is a custom FAN profile. Usually this is set at the factory however it needs to be created again after the BMC update



Then copy the default fan profile as follows



The new fan profile is named "fmdio100v2" with the following settings. Ensure all CPU sensor and all FANs are selected



2023-07-27 05:13:58 (UTC+00:00 GMT) US - English Sync Refresh admin

### Edit New Fan Profile

Back Save New policy Delete selected policy

Name: fmad100v2

Policy: 0 Algorithm: Slope Sensor Type: Temperature 1 Initialize Duty: 50

Sensor: CPU0\_DTS CPU1\_DTS

Fan: BPB\_FAN\_1A BPB\_FAN\_1B BPB\_FAN\_2A BPB\_FAN\_2B BPB\_FAN\_3A BPB\_FAN\_3B BPB\_FAN\_4A BPB\_FAN\_4B BPB\_FAN\_5A BPB\_FAN\_5B BPB\_FAN\_6A BPB\_FAN\_6B BPB\_FAN\_7A BPB\_FAN\_7B BPB\_FAN\_8A BPB\_FAN\_8B

Policy Execute Condition

CPU Tdp (W): 165

Ambient Sensor: INLET\_AIR\_TEMP

Ambient Sensor Value: 0

PCIe Device: Not check pcie device, ignore this execute condition

Reference	Duty (%)	Action
Reference 0: 45	50	Delete
Reference 1: 4	100	Delete

Duty (%) vs Sensor Reading Value

Click Save

2023-07-27 05:15:14 (UTC+00:00 GMT) US - English Sync Refresh admin

### Edit New Fan Profile

Back Save New policy Delete selected policy

Name: fmad100v2

Policy: 0 Algorithm: Slope Sensor Type: Temperature 1 Initialize Duty: 50

Sensor: CPU0\_DTS CPU1\_DTS

Fan: BPB\_FAN\_1A BPB\_FAN\_1B BPB\_FAN\_2A BPB\_FAN\_2B BPB\_FAN\_3A BPB\_FAN\_3B BPB\_FAN\_4A BPB\_FAN\_4B BPB\_FAN\_5A BPB\_FAN\_5B BPB\_FAN\_6A BPB\_FAN\_6B BPB\_FAN\_7A BPB\_FAN\_7B BPB\_FAN\_8A BPB\_FAN\_8B

Policy Execute Condition

CPU Tdp (W): 165

Ambient Sensor: INLET\_AIR\_TEMP

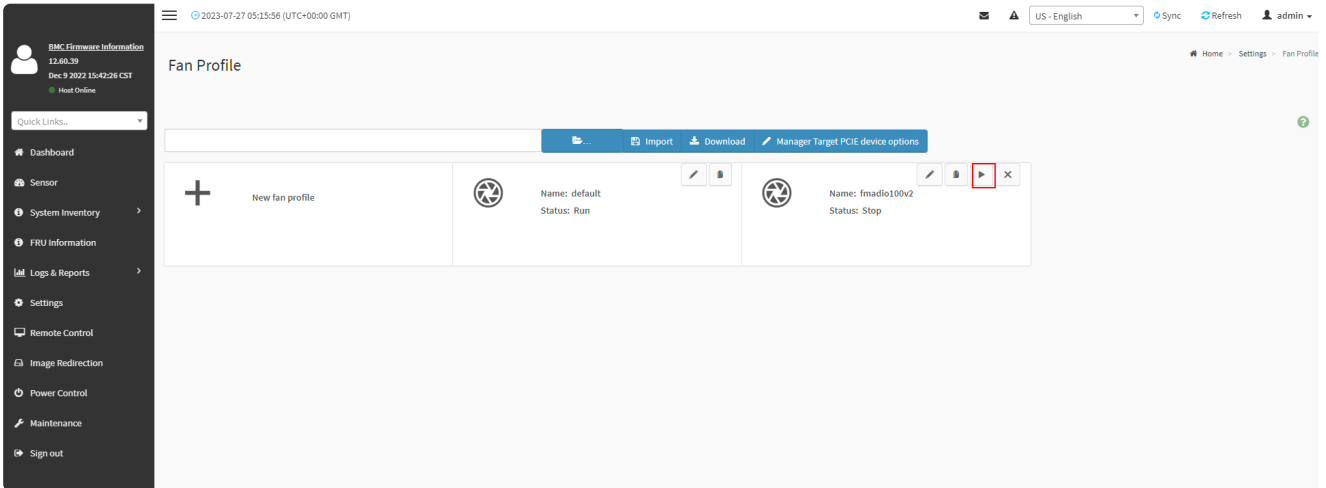
Ambient Sensor Value: 0

PCIe Device: Not check pcie device, ignore this execute condition

Reference	Duty (%)	Action
Reference 0: 45	50	Delete
Reference 1: 4	100	Delete

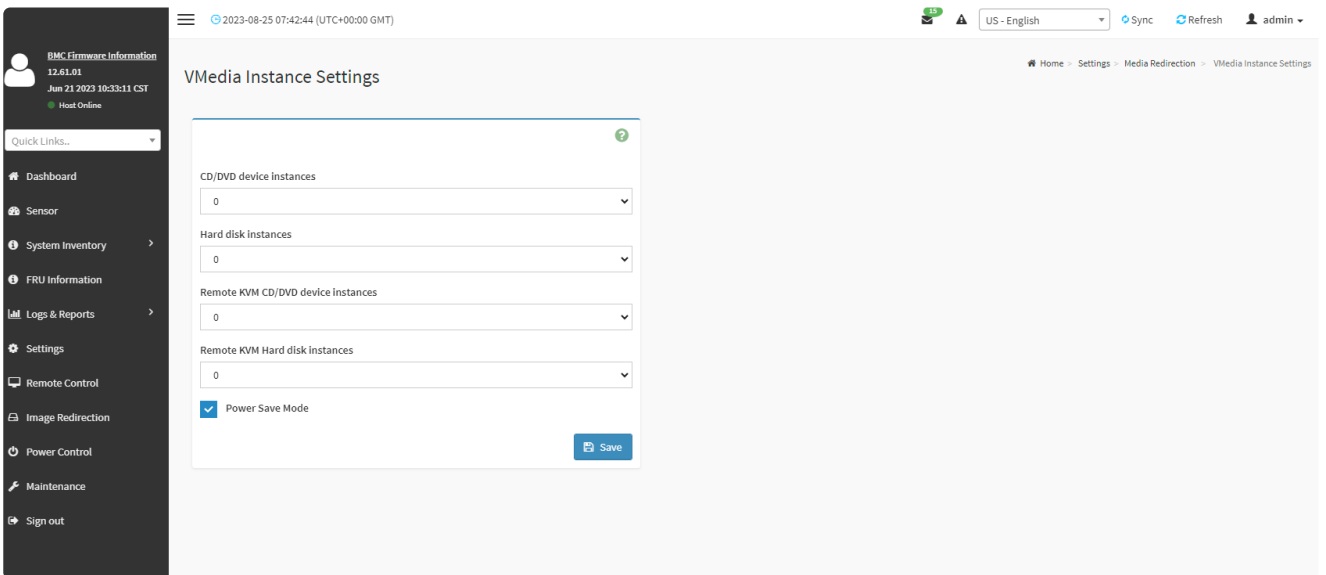
Duty (%) vs Sensor Reading Value

Finally activate the profile



## Step 7) Remote Media settings

Disable all remote media settings as follows



## Step 8) BIOS Upgrade

BIOS update, bios update is located in

```
/opt/fmadio/firmware/bmc/bios_r23
```

Files look like the following

```
fmadio@fmadio100v2-2280:/opt/fmadio/firmware/bmc/bios_r23$ ls -altr
total 0
drwxr-xr-x 4 fmadio staff 80 Aug 11 14:12 ./
drwxr-xr-x 2 fmadio staff 200 Aug 11 14:12 ../
lrwxrwxrwx 1 root root 75 Aug 11 15:03 readme_afu1nx.txt -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/readme_afu1nx.txt
lrwxrwxrwx 1 root root 75 Aug 11 15:03 readme.txt -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/readme.txt
lrwxrwxrwx 1 root root 78 Aug 11 15:03 image.bin.mds -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/image.bin.mds
lrwxrwxrwx 1 root root 74 Aug 11 15:03 image.bin -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/image.bin
lrwxrwxrwx 1 root root 69 Aug 11 15:03 f.sh -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/f.sh
lrwxrwxrwx 1 root root 74 Aug 11 15:03 afu1nx_64 -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/afu1nx_64
lrwxrwxrwx 1 root root 71 Aug 11 15:03 FME.sh -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/FME.sh
lrwxrwxrwx 1 root root 79 Aug 11 15:03 .gitattributes -> /tmp/tcloop/fmadio100v2_current/opt/fmadio/firmware/bmc/bios_r23/.gitattributes
```

To update the BIOS run the command

```
sudo ./f.sh
```

It will take a few minutes to update, the console output looks like the following

```
fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bios_r23$ sudo ./f.sh
===== Update BIOS Region =====
+-----+
|          AMI Firmware Update Utility v5.11.03.1778          |
|   Copyright (C)2018 American Megatrends Inc. All Rights Reserved.   |
+-----+
Reading flash ..... done
- ME Data Size checking . ok
Secure Flash enabled, recalculate ROM size with signature... Enable.
- FFS checksums ..... ok
- Check RomLayout ..... ok.
Loading capsule to secure memory buffer ... done
Erasing Boot Block ..... done
Updating Boot Block ..... done
Verifying Boot Block ..... done
Erasing Main Block ..... done
Updating Main Block ..... done
Verifying Main Block ..... done
Erasing NVRAM Block ..... done
Updating NVRAM Block ..... done
Verifying NVRAM Block ..... done
Erasing NCB Block ..... done
Updating NCB Block ..... done
Verifying NCB Block ..... done
Erasing RomHole Block ..... done
Updating RomHole Block ..... done
Verifying RomHole Block ..... done

Process completed.
===== Update BIOS Region Complete =====
===== !!!!!! NOTE !!!!!! =====
==== 1.Please do power cycle to reboot system.
==== 2.Boot into same file system location.
==== 3.Execute FME.sh to Update ME Region.
=====
fmadio@fmadio40v3-571:/opt/fmadio/firmware/bmc/bios_r23$
```

Then power off the system + power it on

**A full power off is required to load the new BIOS**

### Step 9) BIOS Settings

After BIOS update all settings are lost and need to be set-again, the system will fail to boot also as the BIOS settings have not been configured

Setting the Boot settings as follows

Advanced -> Trusted computing set the following

Advanced

Configuration

Security Device Support

[Disable]

Disable Block Sid

[Disabled]

NO Security Device Found

Enables or Disables BIOS support for security device. O.S. will not show Security Device. TCG EFI protocol and INT1A interface will not be available.

++: Select Screen

↑↓: Select Item

K/M: Scroll Help Area Up/Down.

Enter: Select

+/-: Change Opt.

F1: General Help

F3: Previous Values

F9: Optimized Defaults

F10: Save & Exit

ESC: Exit

Advanced -> Serial Port Console

COM1

Console Redirection

[Enabled]

▶ Console Redirection Settings

Legacy Console Redirection

▶ Legacy Console Redirection Settings

Serial Port for Out-of-Band Management/

Windows Emergency Management Services (EMS)

Console Redirection EMS [Disabled]

▶ Console Redirection Settings

Console Redirection Enable  
or Disable.

⇐⇐: Select Screen  
↑↓: Select Item  
K/M: Scroll Help Area  
Up/Down.  
Enter: Select  
+/-: Change Opt.  
F1: General Help  
F3: Previous Values  
F9: Optimized Defaults  
F10: Save & Exit  
ESC: Exit

Advance -> PCIe Subsystem, configure as follows

```

PCI Bus Driver Version      A5.01.18
MEZZ_1 I/O ROM             [Disabled]
PCI Express Slot #1C I/O ROM [Disabled]
Riser1 I/O ROM             [Disabled]
Riser2 I/O ROM             [Disabled]
MEZZ_2 I/O ROM             [Enabled]
PCI Express Slot #4B I/O ROM [Disabled]
PCI Express Slot #4C I/O ROM [Disabled]
PCI Express Slot #4D I/O ROM [Disabled]
Riser3 I/O ROM             [Disabled]
PCI Express Slot #5B I/O ROM [Disabled]
PCI Express Slot #5C I/O ROM [Disabled]
PCI Express Slot #5D I/O ROM [Disabled]
PCI Express Slot #6C I/O ROM [Disabled]
PCI Express Slot #6D I/O ROM [Disabled]
PCI Express Slot #7 I/O ROM [Disabled]
Onboard LAN1 Controller    [Enabled]
Onboard LAN2 Controller    [Enabled]
Onboard LAN1 I/O ROM       [Disabled]
Onboard LAN2 I/O ROM       [Disabled]

PCI Devices Common Settings:
Above 4G Decoding          [Enabled]
SR-I/OV Support            [Enabled]
    
```

▲ Enable/Disable MEZZ\_2 I/O ROM.

---

⇐⇐: Select Screen  
 ⇕: Select Item  
 K/M: Scroll Help Area Up/Down.  
 Enter: Select  
 +/-: Change Opt.  
 F1: General Help  
 F3: Previous Values  
 F9: Optimized Defaults  
 F10: Save & Exit  
 ESC: Exit

Advanced -> Network Stack

Advanced

Network Stack

[Disabled]

Enable/Disable UEFI  
Network Stack

←→: Select Screen  
↑↓: Select Item  
K/M: Scroll Help Area  
Up/Down.  
Enter: Select  
+/-: Change Opt.  
F1: General Help  
F3: Previous Values  
F9: Optimized Defaults  
F10: Save & Exit  
ESC: Exit

Advance -> NVMe Configuration, configure as follows

## Advanced

## NVMe Configuration

NVMe OPRoM Select

[NVMe Device]

NVMe LED Control

[Enable]

- ▶ NVMe P00: Micron\_9300\_MTFDHAL15T3TDP
- ▶ NVMe P01: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P02: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P03: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P04: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P05: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P06: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P07: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P08: SAMSUNG MZQL23T8HCLS-00A07
- ▶ NVMe P09: SAMSUNG MZQL23T8HCLS-00A07

BIOS Build-In is default setting. Select Device Itself, then this NVMe page will not display any NVMe device. Unless the device doesn't have OPRoM, it will show.

←→: Select Screen  
↑↓: Select Item  
K/M: Scroll Help Area Up/Down.  
Enter: Select  
+/-: Change Opt.  
F1: General Help  
F3: Previous Values  
F9: Optimized Defaults  
F10: Save & Exit  
ESC: Exit

Advanced -&gt; Chipset Configuration



Restore AC Power Loss

[Power On]

P2P Bridge IO Size

[0x1000]

Chassis Opened Warning

[Disabled]

Specify what state when power is re-applied after a power failure (G3 state).

←→: Select Screen  
↑↓: Select Item  
K/M: Scroll Help Area  
Up/Down.  
Enter: Select  
+/-: Change Opt.  
F1: General Help  
F3: Previous Values  
F9: Optimized Defaults  
F10: Save & Exit  
ESC: Exit

Boot configure as follows

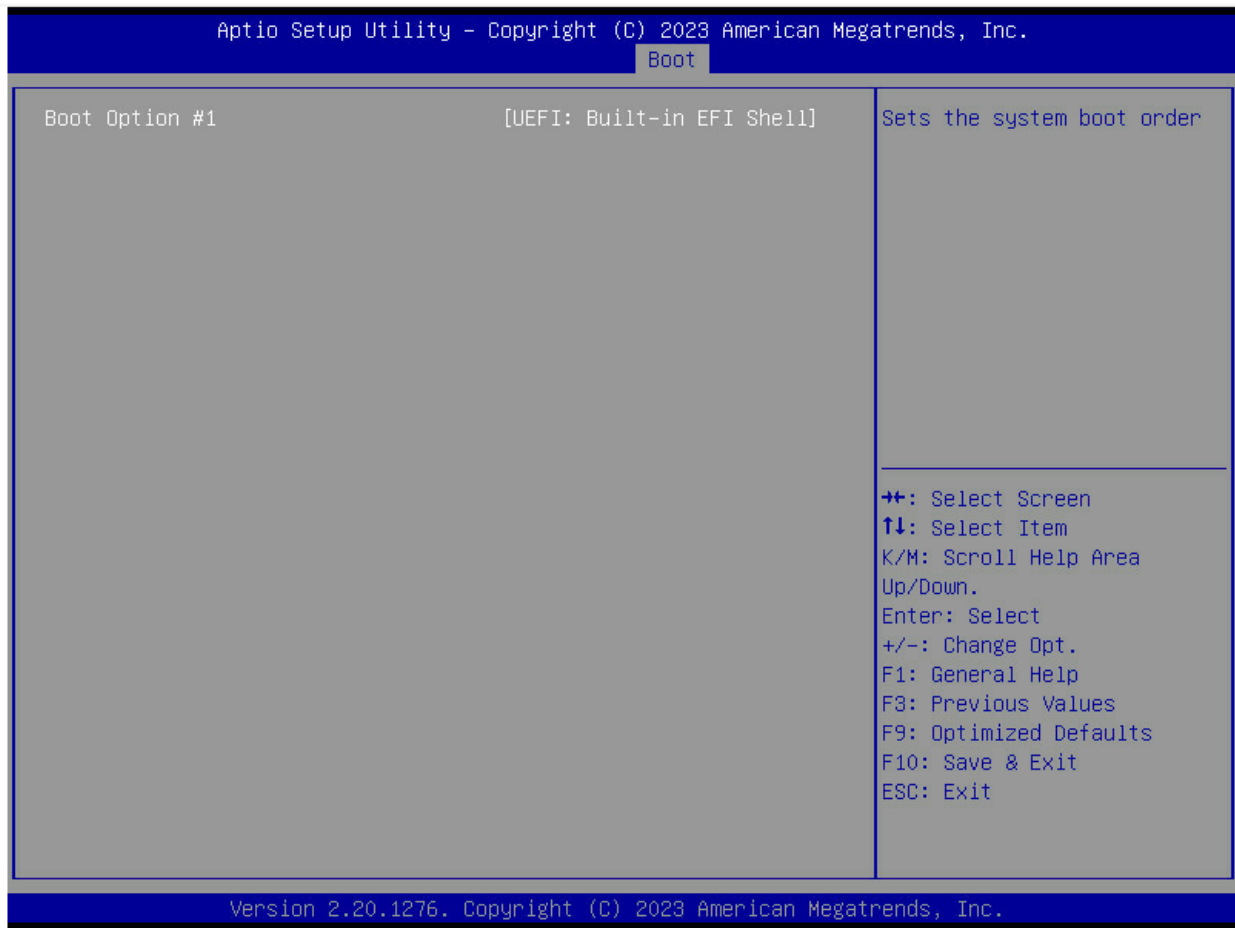
Aptio Setup Utility - Copyright (C) 2023 American Megatrends, Inc.

Main Advanced Chipset Server Mgmt Security **Boot** Save & Exit

<pre> Boot Configuration Setup Prompt Timeout          1 Bootup NumLock State         [On] Quiet Boot                    [Disabled]  Setup Flash Dump full Setup Data Dump non-default Setup Data Restore Setup Data  Boot mode select              [UEFI]  FIXED BOOT ORDER Priorities Boot Option #1                [UEFI AP] Boot Option #2                [Disabled] Boot Option #3                [Disabled] Boot Option #4                [Disabled] Boot Option #5                [Disabled]  ▶ UEFI Hard Disk Drive BBS Priorities ▶ UEFI NETWORK Drive BBS Priorities ▶ UEFI Application Boot Priorities </pre>	<p>Sets the system boot order</p> <hr/> <pre> ⇧⇧: Select Screen ↑↓: Select Item K/M: Scroll Help Area Up/Down. Enter: Select +/-: Change Opt. F1: General Help F3: Previous Values F9: Optimized Defaults F10: Save &amp; Exit ESC: Exit </pre>
--	---

Version 2.20.1276. Copyright (C) 2023 American Megatrends, Inc.

Boot -> UEFI Application Boot Priorities



Save changes and exit

### Step 10) System boot

System boot will look different as it now boots via UEFI, similar to the following.

```

0)/HD(1,GPT,9CFF5AB9-0D02-41D2-9839-EFBB6D09846B,0x800,0x1DCD001)
  BLK0: Alias(s):
    PciRoot(0x7)/Pci(0x0,0x0)/Pci(0x0,0x0)/NVMe(0x1,C6-60-67-25-01-75-A0-0
0)
  BLK2: Alias(s):
    PciRoot(0x7)/Pci(0x0,0x0)/Pci(0x0,0x0)/NVMe(0x1,C6-60-67-25-01-75-A0-0
0)/HD(2,GPT,47FD2ED6-55E4-4B7C-BA89-54F04E8BE2EF,0x1DCE000,0x6FA9FF000)
Press ESC in 1 seconds to skip startup.nsh or any other key to continue.
Shell> echo "FMADIO UEFI"
FMADIO UEFI
Shell> blk1:\EFI\syslinux\syslinux.efi

```

```

  _____
 /         \      |      |      |      |      |      |
|         |      |      |      |      |      |      |
|   |   |  Y Y  |  /   \  /   \  /   \  /   \  /   \  /
|   |   |      |  /   \  /   \  /   \  /   \  /   \  /
|   |   |      | /   \  /   \  /   \  /   \  /   \  /
|   |   |      | \   /  \   /  \   /  \   /  \   /  \
|   |   |      |  \   /  \   /  \   /  \   /  \   /  \
 \         /      |      |      |      |      |      |
  _____
 --+ Packet Capture Masters +--
=====
Press <Enter> to begin or F2, F3, or F4 to view boot options.
boot:
Loading /boot/vmlinuz64... ok
Loading /boot/fmadio100v2-corepure64.gz...ok

```

After boot completion the usual prompt will be shown on both the VGA and Serial ports



# Network Traffic Profile

Generating the network traffic profile can help debug many issues

FMADIO Capture System has built in utility "capinfos2" which can provide fast and compact summary information about a PCAP.,

For network profile information please run as follows

```
sudo stream_cat <insert capture name> | capinfos2 -v --histo-size
```

This will generate a packet size histogram such as the following

```

3.43GB 2.848 Gbps 2.594 Mpps
3.79GB 2.861 Gbps 2.610 Mpps
4.20GB 3.293 Gbps 3.005 Mpps
packet stream end
20220408_025013 12.500s : SUCCESS
Total Packets: 25351479
TotalBytes : 3947124499
TotalPackets : 25351479
PayloadCRC : 15a79d9f1dbb9944
ErrorSeq : 0
ErrorPktSize : 0
LastByte : 0x00000000
SeqStart : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
SeqEnd : 0x00000000 0x00000000 0x00000000 0x00000000 : 0x00000000
PacketCnt : 0 0 0
Timeorder : 0
CRCFail : 0
Time First : 20220407_140446 14:04:46.607.682.562 (1649340286.607682562)
Time Last : 20220407_235958 23:59:58.580.699.475 (1649375998.580699475)
TotalPCAPTime : 35711973016913 ns
Bandwidth : 0.001 Gbps
Packet Rate : 0.001 Mpps

```

Packet Size Histogram: wire

```

64 B : 39720 : 0.0016 : *
66 B : 2771 : 0.0017 : *
67 B : 2 : 0.0017 : *
68 B : 14 : 0.0017 : *
69 B : 5 : 0.0017 : *
70 B : 11147031 : 0.4414 : *****
71 B : 107 : 0.4414 : *
72 B : 6 : 0.4414 : *
73 B : 1 : 0.4414 : *
74 B : 240 : 0.4414 : *
75 B : 595732 : 0.4649 : *****
76 B : 1787331 : 0.5354 : *****
77 B : 1304 : 0.5354 : *
78 B : 2987733 : 0.6533 : *****
80 B : 360 : 0.6533 : *
81 B : 158 : 0.6533 : *
82 B : 437 : 0.6533 : *
83 B : 432 : 0.6533 : *
84 B : 1191282 : 0.7003 : *****
85 B : 7351 : 0.7006 : *
86 B : 1792 : 0.7007 : *
87 B : 595609 : 0.7242 : *****
88 B : 437 : 0.7242 : *
89 B : 382 : 0.7242 : *
90 B : 596158 : 0.7477 : *****
91 B : 595 : 0.7478 : *
92 B : 277 : 0.7478 : *
93 B : 595597 : 0.7713 : *****
94 B : 597195 : 0.7948 : *****
95 B : 121 : 0.7948 : *
96 B : 444 : 0.7949 : *
97 B : 241 : 0.7949 : *
98 B : 5392 : 0.7951 : *
99 B : 473 : 0.7951 : *
100 B : 595754 : 0.8186 : *****
101 B : 593 : 0.8186 : *
102 B : 874 : 0.8186 : *
103 B : 8 : 0.8186 : *
104 B : 1191153 : 0.8656 : *****
105 B : 403 : 0.8657 : *
106 B : 12 : 0.8657 : *
107 B : 15 : 0.8657 : *
108 B : 28 : 0.8657 : *
109 B : 596291 : 0.8892 : *****
110 B : 68 : 0.8892 : *
111 B : 1 : 0.8892 : *
112 B : 5 : 0.8892 : *
113 B : 356 : 0.8892 : *
114 B : 439 : 0.8892 : *
115 B : 792 : 0.8892 : *
116 B : 379 : 0.8893 : *
117 B : 49447 : 0.8912 : *
118 B : 553041 : 0.9130 : *****
119 B : 572 : 0.9130 : *
120 B : 44 : 0.9130 : *
121 B : 434 : 0.9131 : *
122 B : 713 : 0.9131 : *
123 B : 356 : 0.9131 : *
124 B : 2 : 0.9131 : *
125 B : 118 : 0.9131 : *
126 B : 187 : 0.9131 : *
127 B : 6 : 0.9131 : *
128 B : 413 : 0.9131 : *
129 B : 507 : 0.9132 : *
130 B : 76 : 0.9132 : *
131 B : 28 : 0.9132 : *
132 B : 12 : 0.9132 : *
133 B : 57 : 0.9132 : *
134 B : 4159 : 0.9133 : *
135 B : 367 : 0.9133 : *
136 B : 519 : 0.9134 : *
137 B : 445 : 0.9134 : *
138 B : 395 : 0.9134 : *
139 B : 123 : 0.9134 : *
140 B : 85 : 0.9134 : *
141 B : 86 : 0.9134 : *
142 B : 157 : 0.9134 : *
143 B : 69 : 0.9134 : *
144 B : 214 : 0.9134 : *
145 B : 100 : 0.9134 : *
146 B : 467 : 0.9134 : *
147 B : 138 : 0.9134 : *
148 B : 371 : 0.9135 : *
149 B : 72 : 0.9135 : *

```

With a more compact histogram version shown below. Note the first graph is the packet size histogram of packets on the wire, the second is size based on the packets captured. In this case packets have been sliced @ 128B thus the histogram is quite short

```

Packet Size Histogram Compact: Wire
 64 B : 20150180 (0.795) : 0.7948 : *****
 96 B : 2998641 (0.118) : 0.9131 : *****
128 B : 14654 (0.001) : 0.9137 : *
160 B : 79303 (0.003) : 0.9168 : *
192 B : 597429 (0.024) : 0.9404 : ***
224 B : 71532 (0.003) : 0.9432 : *
256 B : 1122 (0.000) : 0.9433 : *
288 B : 850 (0.000) : 0.9433 : *
320 B : 81279 (0.003) : 0.9465 : *
352 B : 718 (0.000) : 0.9465 : *
384 B : 32857 (0.001) : 0.9478 : *
416 B : 738 (0.000) : 0.9478 : *
448 B : 413 (0.000) : 0.9479 : *
480 B : 14338 (0.001) : 0.9484 : *
512 B : 14659 (0.001) : 0.9490 : *
544 B : 63643 (0.003) : 0.9515 : *
576 B : 2781 (0.000) : 0.9516 : *
608 B : 147 (0.000) : 0.9516 : *
640 B : 17 (0.000) : 0.9516 : *
672 B : 1603 (0.000) : 0.9517 : *
704 B : 28 (0.000) : 0.9517 : *
736 B : 725 (0.000) : 0.9517 : *
768 B : 37 (0.000) : 0.9517 : *
800 B : 912 (0.000) : 0.9518 : *
832 B : 1 (0.000) : 0.9518 : *
864 B : 62 (0.000) : 0.9518 : *
896 B : 5 (0.000) : 0.9518 : *
928 B : 7 (0.000) : 0.9518 : *
960 B : 43 (0.000) : 0.9518 : *
992 B : 67 (0.000) : 0.9518 : *
1024 B : 21 (0.000) : 0.9518 : *
1056 B : 13 (0.000) : 0.9518 : *
1088 B : 1816 (0.000) : 0.9518 : *
1120 B : 3 (0.000) : 0.9518 : *
1152 B : 28 (0.000) : 0.9518 : *
1184 B : 9 (0.000) : 0.9518 : *
1216 B : 307 (0.000) : 0.9519 : *
1248 B : 10 (0.000) : 0.9519 : *
1280 B : 412 (0.000) : 0.9519 : *
1312 B : 368 (0.000) : 0.9519 : *
1344 B : 4 (0.000) : 0.9519 : *
1376 B : 7 (0.000) : 0.9519 : *
1408 B : 3 (0.000) : 0.9519 : *
1440 B : 2 (0.000) : 0.9519 : *
1472 B : 2802 (0.000) : 0.9520 : *
1504 B : 1216883 (0.048) : 1.0000 : *****
Packet Size Histogram Compact: capture
 64 B : 20150180 (0.795) : 0.7948 : *****
 96 B : 2998641 (0.118) : 0.9131 : *****
128 B : 2202658 (0.087) : 1.0000 : *****
Packet Size Histogram: Profile
SizeProfile 0.794832 SizeProfile 0.118283 SizeProfile 0.000578 SizeProfile 0.003128 SizeProfile 0.023566 SizeProfile 0.002822 SizeProfile 0.000044
01 SizeProfile 0.000036 SizeProfile 0.000000 SizeProfile 0.000002 SizeProfile 0.000000 SizeProfile 0.000000 SizeProfile 0.000000 SizeProfile 0.000002 SizeProfile 0.000000
complete
fmadio@fmadio20v3-287: /mnt/store0/develop_20220407_rc3/capinfos2$

```

FMADIO Network Packet Size Histogram Compact

Finally at the end there is a number list, which can be sent to FMADIO Support so we can replicate your network traffic profile locally with our synthetic packet generator.

```

0.794832 0.118283 0.000578 0.003128 0.023566 0.002822 0.000044 0.000034 0.003206 0.000028 0.001296 0.000029 0.000016 0.000566 0.000578 0.002510 0.000110 0.000006 0.000001 0.000063 0.000001 0.000029 0.000001 0.000000
complete
fmadio@fmadio20v3-287: /mnt/store0/develop_20220407_rc3/capinfos2$

```



# 100G Capture Port Link Up

FW: 8224+

Getting 100G links link up either works no problem out of the box or be fickle to setup and get a link.

Usually the main source of problems is FEC vs non-FEC setting on both the switch and on the FMADIO system. FMADIO Packet Capture systems will try detect if FEC is on the link by alternating FEC enable / FEC disabled when trying to link up.

This can cause problems if the switch its connected to is also cycling thru the different combinations.

To check the current status the fmadiocli utility provides the easiest way to monitor the state. Use the command, below to get the current link state.

```
show interface status
```

```
wed Aug 24 02:35:11 2022] > show interface status
wed Aug 24 02:35:11 2022] Port      Description Status      Speed      Transceiver  RxPower    TxPower    Temperature  FEC      Vendor      Vendor PN
wed Aug 24 02:35:15 2022] -----
wed Aug 24 02:35:15 2022] man0      connected   1G          40G Base-SR4 0.7824 mW  0.0000 mW  36.90 C      AVAGO     AFBR-79EQDZ
wed Aug 24 02:35:15 2022] man10     connected   40G         100G CR      0.000 mW   0.000 mA   0.000 C      FS        Q28-PC03
wed Aug 24 02:35:15 2022] cap0      notconnected 100G        100G CR      0.000 mW   0.000 mA   0.000 C      Arista Networks Q28-PC01
wed Aug 24 02:35:15 2022] cap1      connected   100G        100G CR      0.000 mW   0.000 mA   0.000 C      Arista Networks Q28-PC01
wed Aug 24 02:35:16 2022] >
```

In the above we see capture port 0 (cap0) is link down and capture port 1(cap1) is link up

The link may be bounced by shutting down using the command as follows

```
config interface shutdown cap0
config interface shutdown cap1
```

It may take 1min for the link to go down, use the show interface status command to wait for link down to complete.

```
wed Aug 24 02:38:07 2022] > show interface status
wed Aug 24 02:38:07 2022] Port      Description Status      Speed      Transceiver  RxPower    TxPower    Temperature  FEC      Vendor      Vendor PN
wed Aug 24 02:38:07 2022] -----
wed Aug 24 02:38:07 2022] man0      connected   1G          40G Base-SR4 0.7812 mW  0.0000 mW  36.90 C      AVAGO     AFBR-79EQDZ
wed Aug 24 02:38:07 2022] man10     connected   40G         100G CR      0.000 mW   0.000 mA   0.000 C      FS        Q28-PC03
wed Aug 24 02:38:07 2022] cap0      notconnected 100G        100G CR      0.000 mW   0.000 mA   0.000 C      Arista Networks Q28-PC01
wed Aug 24 02:38:07 2022] cap1      notconnected 100G        100G CR      0.000 mW   0.000 mA   0.000 C      Arista Networks Q28-PC01
wed Aug 24 02:40:09 2022] >
```

Once the link is down, disable the shutdown using the command

```
config interface no shutdown cap0
config interface no shutdown cap1
```

NOTE: status currently takes 30-60sec to update. Please be patient.

## FEC Link Config

The default setting of FMADIO Packet Capture system is FEC running in "auto" mode. This means the system will try link up without FEC, if it fails FEC is enabled and link up is tried again. This process is repeated indefinitely.

For some switches and NICs this works for others it has problems. e.g. if both end points are cycling like this its possible it will ever link up due to phase/offset of the cycles.

Its best to force the FEC setting if your expecting FEC to run on the link. This is done as follows

```
config interface fec cap0
config interface fec cap1
```

```
[Wed Aug 24 05:35:59 2022] > config interface fec cap0
[Wed Aug 24 05:35:59 2022] Disable cycle calibration
[Wed Aug 24 05:35:59 2022] FEC Force
[Wed Aug 24 05:35:59 2022] FECENable: 1 PortMask:0001
[Wed Aug 24 05:35:59 2022] [0] FECENable: 1 FECForce:1
[Wed Aug 24 05:35:59 2022] [1] FECENable: 1 FECForce:0
[Wed Aug 24 05:35:59 2022] set interface [cap0] fec (true) -> (true)
[Wed Aug 24 05:35:59 2022]
[Wed Aug 24 05:36:00 2022] > config interface fec cap1
[Wed Aug 24 05:36:00 2022] Disable cycle calibration
[Wed Aug 24 05:36:00 2022] FEC Force
[Wed Aug 24 05:36:00 2022] FECENable: 1 PortMask:0002
[Wed Aug 24 05:36:00 2022] [0] FECENable: 1 FECForce:1
[Wed Aug 24 05:36:00 2022] [1] FECENable: 1 FECForce:1
[Wed Aug 24 05:36:00 2022] set interface [cap1] fec (false) -> (true)
[Wed Aug 24 05:36:00 2022] >
```

After forcing FEC on the links the interface status will show a "force-" option as follows

```
[Wed Aug 24 05:36:55 2022] > show interface status
[Wed Aug 24 05:36:55 2022] Port Description Status Speed Transceiver RxPower TxPower Temperature FEC Vendor Vendor PN
[Wed Aug 24 05:36:55 2022] ---
[Wed Aug 24 05:36:55 2022] man0 connected 1G
[Wed Aug 24 05:36:55 2022] man10 connected 40G 40G Base-SR4 0.7762 mW 0.0000 mW 36.38 C Force-on AVAGO AFBR-79EQD2
[Wed Aug 24 05:36:55 2022] cap0 connected 100G 100G CR 0.000 mW 0.000 mA 0.000 C Force-on FS Q28-PC03
[Wed Aug 24 05:36:55 2022] cap1 connected 100G 100G CR 0.000 mW 0.000 mA 0.000 C Force-on Arista Networks Q28-PC01
[Wed Aug 24 05:36:55 2022] >
```

This indicates FEC has been forced on for the specific interfaces.

NOTE: bouncing the port on the switch may be required, it depends what state the switch port is. It may have entered an error mode, or backed off on the autoneg. Bouncing the port on the switch most of the time will resolve the issue.

### Additional Link Up Debug

If the link is still failing to link up, you can trace the state machine for link up procedure using the command

```
fnic_test --trans-trace -v
```

By default it runs on cap0 using the --port 1 can run the trace on the other port. This prints out the realtime event history of the link up process, below is an example of a link bounce.

```

Settle [1] RESET : 000000002faF080 ( 1.000 sec)
Settle [2] TRANS : 00000001dcd6500 ( 10.000 sec)
Settle [3] USR_RESET : 000000002faF080 ( 1.000 sec)
Settle [4] RXALIGN : 000000002faF080 ( 1.000 sec)
Settle [5] TXWAIT : 000000002faF080 ( 1.000 sec)
Settle [6] TXENABLE : 00000007fffff ( 42.950 sec)

rncvoper trace
TS:1661319773476321000 1661319773476.321045 msec: 0 700000150fff038f 00000007fffff 41710001138e0018 State:7 LinkCnt:21 Settle:4171 rx_block_lock:f rx_fec_lock:f rx_sync:f
TS:1661319776298566000 2822.245000 msec: 0 10000015000003e9 00000007fffff 02fa000110103020 State:1 linkCnt:21 Settle:02fa rx_block_lock:0 rx_fec_lock:0 rx_sync:0 mod_res
TS:1661319777298599000 1000.033000 msec: 0 20000016000003e9 00000007fffff 1dcd000110103022 State:2 linkCnt:22 Settle:1dcd rx_block_lock:0 rx_fec_lock:0 rx_sync:0 mod_res
TS:16613197772989161000 0.562000 msec: 0 20000016000003e9 00000007fffff 1dcd000110003001 State:2 linkCnt:22 Settle:1dcd rx_block_lock:0 rx_fec_lock:0 rx_sync:0 mod_res
TS:1661319777299590000 0.429000 msec: 0 20000016000003e9 00000007fffff 1dcd000110003000 State:2 linkCnt:22 Settle:1dcd rx_block_lock:0 rx_fec_lock:0 rx_sync:0 mod_res
TS:1661319777299663000 0.073000 msec: 0 20000016000003e9 00000007fffff 1dcd000110103000 State:2 linkCnt:22 Settle:1dcd rx_block_lock:0 rx_fec_lock:0 rx_sync:0 mod_res
TS:1661319777467330000 167.667000 msec: 0 20000016008003e9 00000007fffff 1d4c000110103000 State:2 linkCnt:22 Settle:1d4c rx_block_lock:0 rx_fec_lock:8 rx_sync:0 mod_res
TS:1661319777467998000 0.668000 msec: 0 2000001600a003e9 00000007fffff 1d4c000110103000 State:2 linkCnt:22 Settle:1d4c rx_block_lock:0 rx_fec_lock:a rx_sync:0 mod_res
TS:1661319777468835000 0.837000 msec: 0 2000001600b003e9 00000007fffff 1d4b000110103000 State:2 linkCnt:22 Settle:1d4b rx_block_lock:0 rx_fec_lock:b rx_sync:0 mod_res
TS:1661319777469252000 0.417000 msec: 0 20000016000003e9 00000007fffff 1d4b000110103000 State:2 linkCnt:22 Settle:1d4b rx_block_lock:0 rx_fec_lock:0 rx_sync:0 mod_res
TS:1661319777470091000 0.839000 msec: 0 200000160ff003e9 00000007fffff 1d4a000110903008 State:2 linkCnt:22 Settle:1d4a rx_block_lock:f rx_fec_lock:f rx_sync:0 mod_res
TS:1661319777470515000 0.424000 msec: 0 200000160fff03e9 00000007fffff 1d4a000110883008 State:2 linkCnt:22 Settle:1d4a rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319777471753000 1.238000 msec: 0 200000160fff039f 00000007fffff 1d490001308e0018 State:2 linkCnt:22 Settle:1d49 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319777526915000 55.162000 msec: 0 200000160fff038f 00000007fffff 1d1f0001108c0018 State:2 linkCnt:22 Settle:1d1f rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:166131978298597000 9771.682000 msec: 0 300000160fff038f 00000007fffff 02fa0001108c0018 State:3 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319788297295000 998.698000 msec: 0 400000160fff039f 00000007fffff 02fa0001f08e0018 State:4 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319788297387000 0.092000 msec: 0 400000160fff038f 00000007fffff 02fa0001d08c0018 State:4 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319788297522000 0.135000 msec: 0 400000160fff039f 00000007fffff 02fa0001f08e0018 State:4 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res

TS:1661319789295359000 0.061000 msec: 0 400000160fff038f 00000007fffff 00010001d08c0018 State:4 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789295420000 0.061000 msec: 0 400000160fff039f 00000007fffff 00010001f08e0018 State:4 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789295598000 0.178000 msec: 0 400000160fff038f 00000007fffff 00010001d08c0018 State:4 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789295671000 0.073000 msec: 0 400000160fff039f 00000007fffff 00010001f08e0018 State:4 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789295801000 0.130000 msec: 0 400000160fff038f 00000007fffff 00010001d08c0018 State:4 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789295873000 0.072000 msec: 0 400000160fff039f 00000007fffff 00010001f08e0018 State:4 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res

TS:1661319789295944000 0.071000 msec: 0 500000160fff038f 00000007fffff 02fa0001d08c0018 State:5 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789296073000 0.129000 msec: 0 500000160fff039f 00000007fffff 02fa0001f08e0018 State:5 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789296203000 0.130000 msec: 0 500000160fff039f 00000007fffff 02fa0001f08c0018 State:5 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789296277000 0.074000 msec: 0 500000160fff039f 00000007fffff 02fa0001f08e0018 State:5 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319789296465000 0.188000 msec: 0 500000160fff038f 00000007fffff 02fa0001d08c0018 State:5 linkCnt:22 Settle:02fa rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res

TS:1661319790294557000 0.072000 msec: 0 500000160fff039f 00000007fffff 00010001f08e0018 State:5 linkCnt:22 Settle:0001 rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res
TS:1661319790294630000 0.073000 msec: 0 700000160fff038f 00000007fffff 7fff0001138c0018 State:7 linkCnt:22 Settle:7fff rx_block_lock:f rx_fec_lock:f rx_sync:f mod_res

```

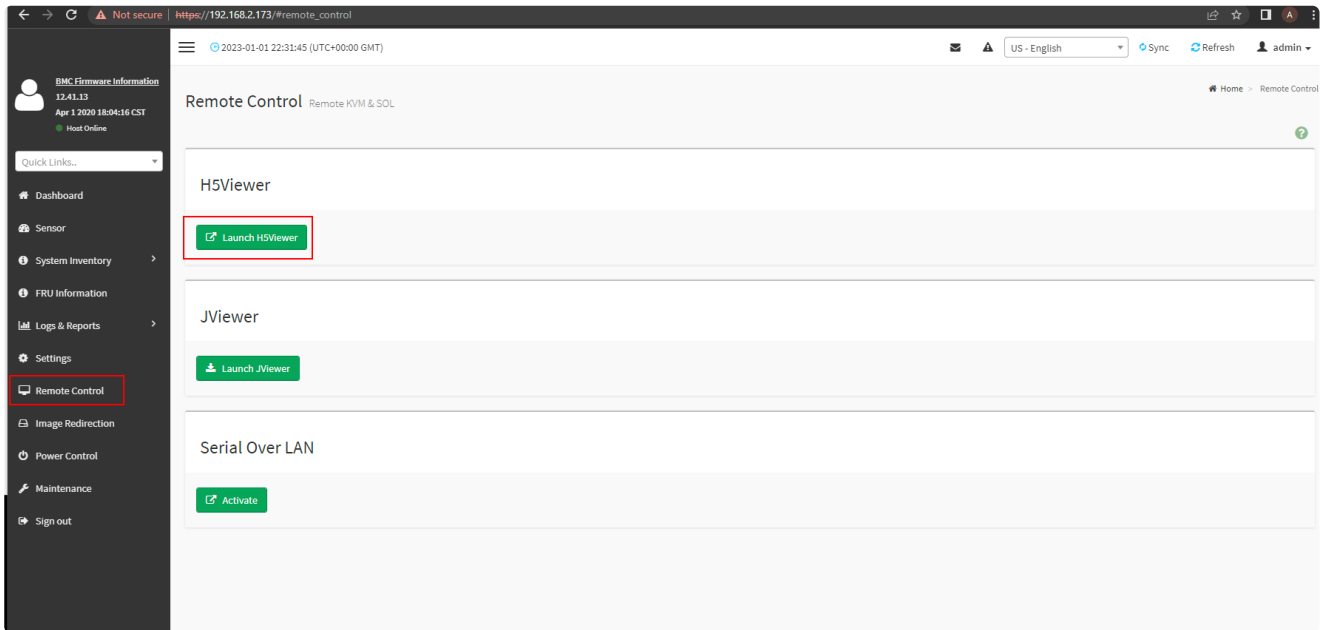
In the above output we see the link state go from State 1 -> State 7. Where state 7 is a stable link up.

For additional assistance on link problems, please run the above and pipe to a logfile. Send the logfile to support @ fmad . io for further assistance.

# SSH Password Reset

To reset the SSH password system the system requires KVM either directly with a physical monitor and keyboard or via the BMC HTML viewer. We recommend HTML viewer and will show the steps here. The process is the same for a physical monitor and keyboard

1) Connect to the BMC login and launch the HTML5 KVM



2) The KVM viewer looks similar to below. If the screen is blank press a few random keys to disable the screensaver

Not secure | https://192.168.2.173/viewer.html

Stop KVM CD Image: Browse File (0 KB) Start Media

Video Mouse Options Keyboard Send Keys Hot Keys Video Record Power Active Users Help Zoom 100%

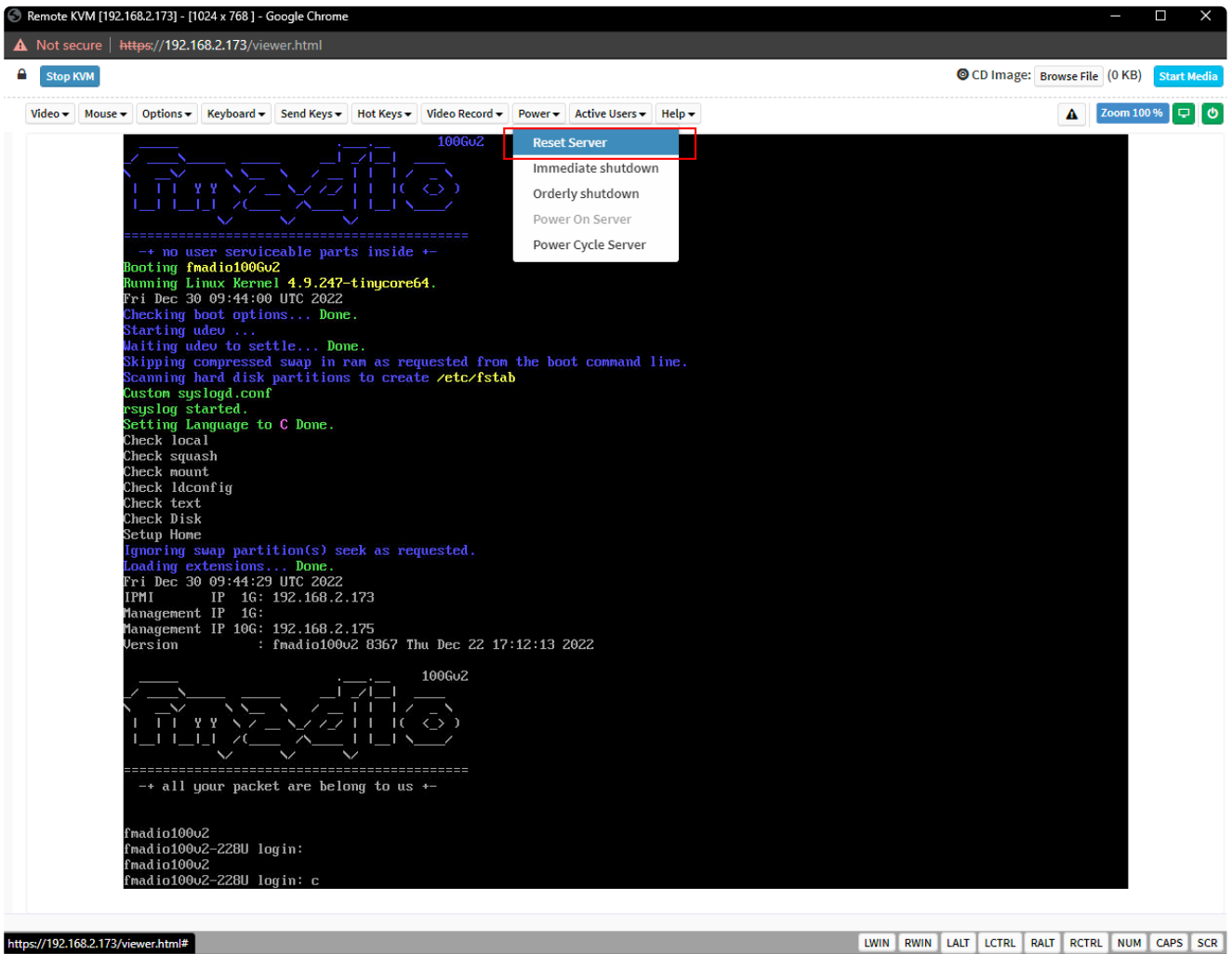
```
100Gu2
=====
  + no user serviceable parts inside +
Booting fmadio100Gu2
Running Linux Kernel 4.9.247-tinycore64.
Fri Dec 30 09:44:00 UTC 2022
Checking boot options... Done.
Starting udev ...
Waiting udev to settle... Done.
Skipping compressed swap in ram as requested from the boot command line.
Scanning hard disk partitions to create /etc/fstab
Custom syslogd.conf
rsyslog started.
Setting Language to C Done.
Check local
Check squash
Check mount
Check ldconfig
Check text
Check Disk
Setup Home
Ignoring swap partition(s) seek as requested.
Loading extensions... Done.
Fri Dec 30 09:44:29 UTC 2022
IPMI IP 1G: 192.168.2.173
Management IP 1G:
Management IP 10G: 192.168.2.175
Version : fmadio100u2 8367 Thu Dec 22 17:12:13 2022

100Gu2
=====
  + all your packet are belong to us +

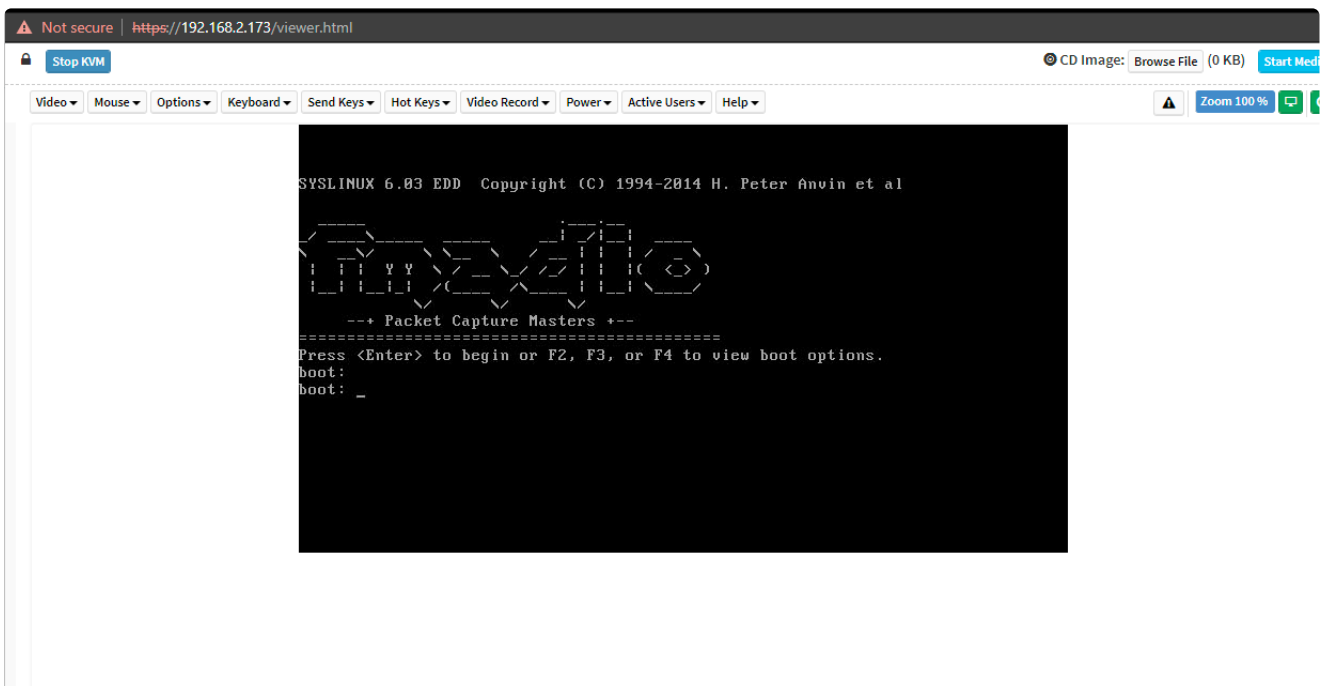
fmadio100u2
fmadio100u2-228U login:
fmadio100u2
fmadio100u2-228U login:
```

LWIN RWIN LALT LCTRL RALT RCTRL NUM CAPS SCR

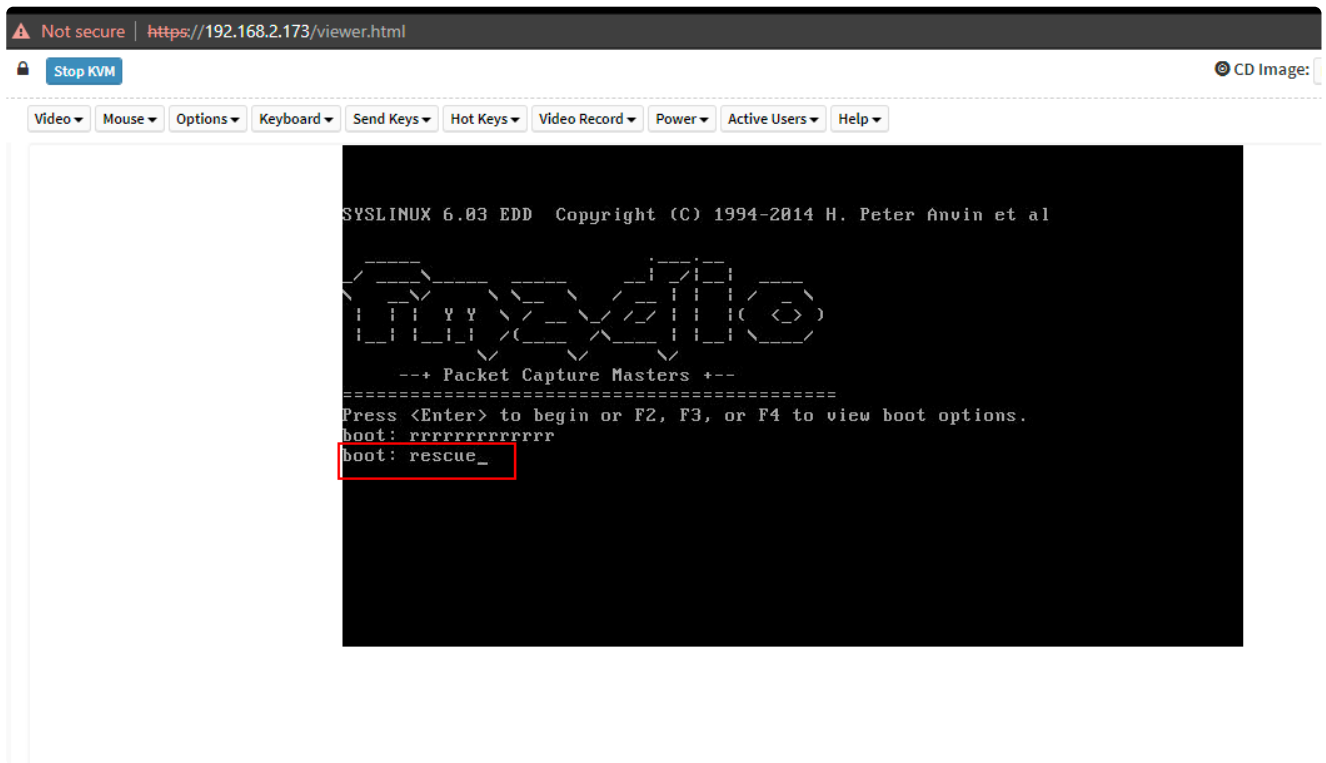
3) Reboot the system using the Power menu



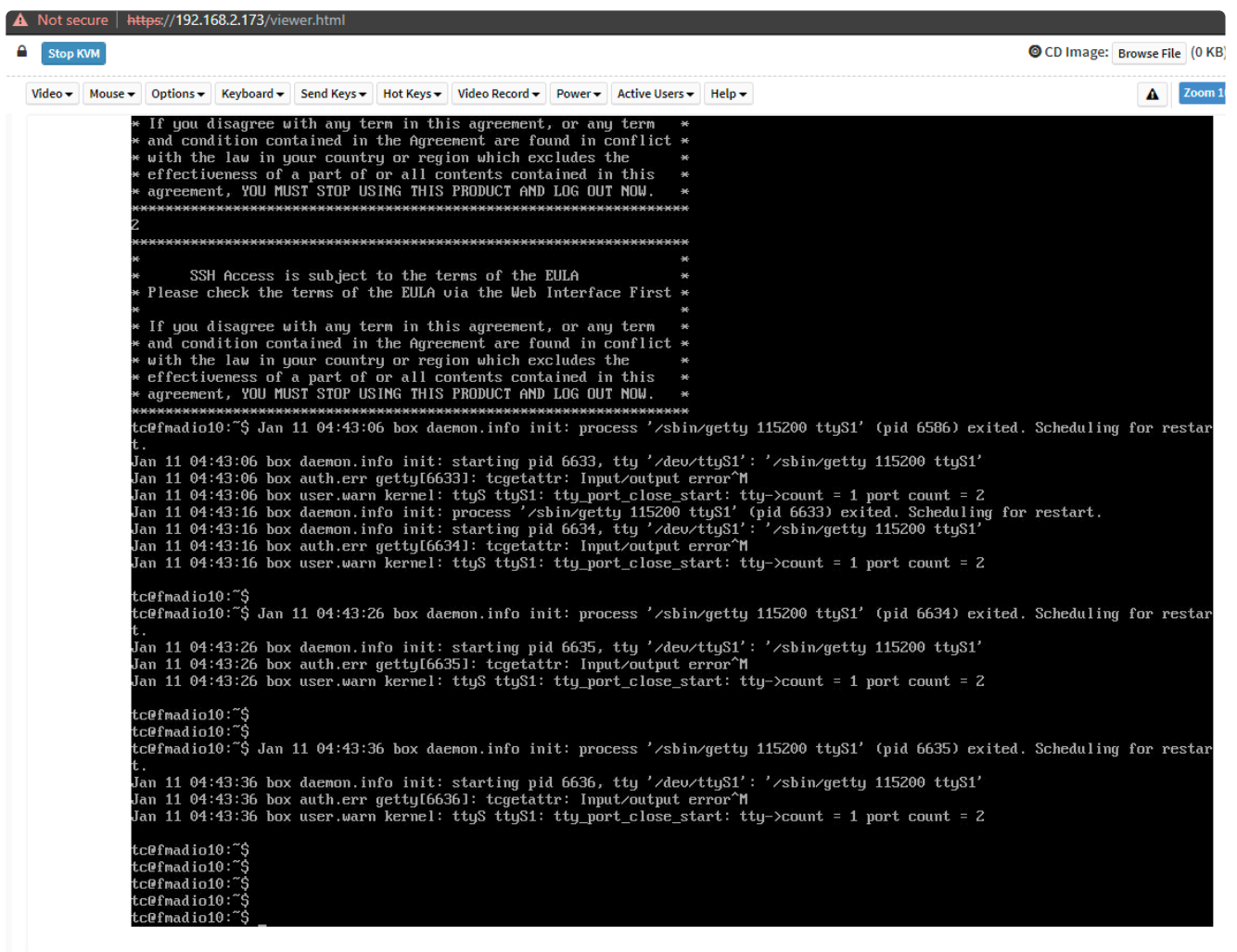
4) Need to catch the system boot prompt here. One way is to tap space key or letter "r" every few seconds. This will prevent the bootloader from automatically starting.



5) Enter system "rescue" mode at the prompt and hit enter. This will boot the system into rescue mode.



6) Rescue mode drop to a default prompt as follows



7) Set a new password for fmadio user as follows

```
sudo passwd fmadio
```

```
Jan 11 04:45:16 box auth.err getty[6666]: tcgetatr: Input/output error^M
Jan 11 04:45:16 box user.warn kernel: ttyS ttyS1: tty_port_close_start: tty->count = 1 port count = 2
tc@fmadio10:~$ sudo passwd fmadio
Jan 11 04:45:17 box authpriv.notice sudo: tc : TTY=ttty1 ; PWD=/home/tc ; USER=root ; COMMAND=/usr/bin/passwd fmadio
Changing password for fmadio
Enter the new password (minimum of 5, maximum of 8 characters)
Please use a combination of upper and lower case letters and numbers.
New password:
Re-enter new password:
passwd: password changed.
tc@fmadio10:~$
```

8) Mount the persistent storage volume as follows. The exact path depends on the system SKU

**FMADIO100v2 or FMADIO100p3:**

```
sudo mount /dev/nvme0n1p2 /mnt/store0
```

**FMADIO20v3 or FMADIO40v3:**

```
sudo mount /dev/sda2 /mnt/store0
```

```
tc@fmadio10:~$
tc@fmadio10:~$
tc@fmadio10:~$
tc@fmadio10:~$
tc@fmadio10:~$ sudo mount /dev/nvme0n1p2 /mnt/store0
```

9) Copy the shadow password file to persistent storage

```
sudo cp /etc/shadow /mnt/store0/etc/shadow
```

```
tc@fmadio10:~$ sudo cp /etc/
tc@fmadio10:~$
tc@fmadio10:~$ sudo cp /etc/shadow /mnt/store0/etc/shadow
tc@fmadio10:~$
```

10) reboot the system

```
sudo reboot
```

11) SSH login fmadio has the newly set password



```
Not secure | https://192.168.2.173/viewer.html
Stop KVM
CD Image: Browse File (0 KB) Start M
Video Mouse Options Keyboard Send Keys Hot Keys Video Record Power Active Users Help
Zoom 100 %

Custom syslogd.conf
rsyslog started.
Setting Language to C Done.
Check local
Check squash
Check mount
Check ldconfig
Check text
Check Disk
Setup Home
Ignoring swap partition(s) seek as requested.
Loading extensions... Done.
Wed Jan 11 04:51:13 UTC 2023
IPMI IP 1G: 192.168.2.173
Management IP 1G:
Management IP 10G: 192.168.2.175
Version : fmadio100v2 8367 Thu Dec 22 17:12:13 2022

100Gv2
=====
-+ all your packet are belong to us +-
=====

fmadio100v2
fmadio100v2-228U login: fmadio
Password:

100Gv2
=====
-+ all your packet are belong to us +-
=====
Error: Failed to call git rev-parse --git-dir --show-toplevel: "fatal: not a git repository (or any parent up to mount point /mnt)\nStopping at filesystem boundary (GIT_DISCOVERY_ACROSS_FILESYSTEM not set).\n"
Git LFS initialized.
SSH_AUTH_SOCK=/tmp/ssh-Wlma7taBE8dU/agent.11504;
SSH_AGENT_PID=11505;
Error: Failed to call git rev-parse --git-dir --show-toplevel: "fatal: not a git repository (or any parent up to mount point /mnt)\nStopping at filesystem boundary (GIT_DISCOVERY_ACROSS_FILESYSTEM not set).\n"
Git LFS initialized.
fmadio@fmadio100v2-228U:~$
```

## Web User Password Reset

In addition to the above web user password and access list can be configured using fmadiocli tool. Documentation is here

<https://docs.fmad.io/fmadio-documentation/cli-reference/fmadiocli#config-userlist-password>

**Maintenance**

# SSD Replacement

SSD Replacement is quite simple, the FMADIO device needs to be powered down, but does not need to be un-racked. Front access to the system is all that's required.

Below is a picture of the FMADIO100G 1U



FMADIO 100G SSD replacement

## 3-6 Installing the Hard Disk Drive

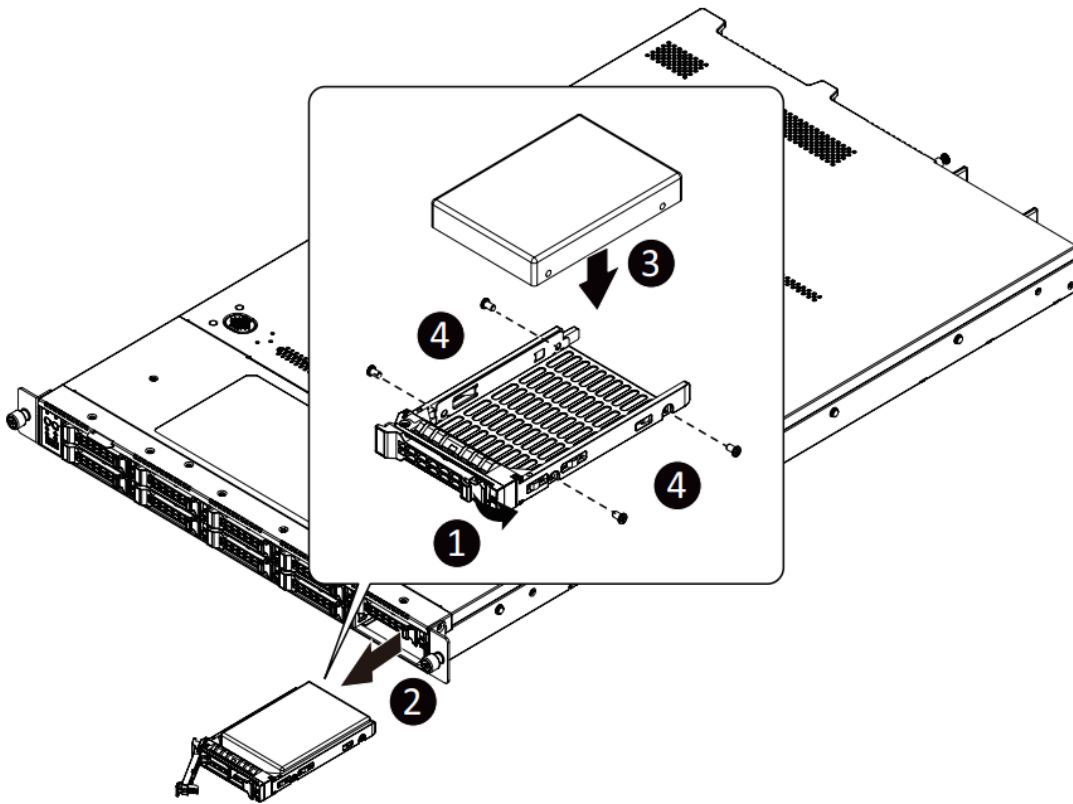


Read the following guidelines before you begin to install the Hard disk drive:

- Take note of the drive tray orientation before sliding it out.
- The tray will not fit back into the bay if inserted incorrectly.
- Make sure that the HDD is connected to the HDD connector on the backplane.

### Follow these instructions to install a 3.5" hard disk drive:

1. Press the release button.
2. Extend the locking lever and pull the locking lever to remove the HDD tray.
3. Place the hard disk drive into the HDD tray.
4. Secure the hard disk drive to the HDD tray with four screws.



#### 1) Power down the system

It's preferred the system is powered down when replacing SSDs

#### 2) Remove the SSD

Remove the SSD by pressing on the orange tab such as picture below



**3) Remove the SSD entirely from the chassis**

Remove the SSD + Caddie entirely from the chassis



**4) Remove SSD from Caddie**

Remove the SSD from the Caddie via 2 screws on each side of the drive



# Firmware Update (CLI)

## System Firmware Update

Upload Firmware into the system is the following process

1. scp or curl the \*.bin firmware file to the system
2. upload the firmware into the system
3. install the firmware on the system
4. reboot the system

Its fairly straight forward to do, in many instances CLI based update is easier than a GUI

### 1) Copy firmware to the system

This is either scp or curl -O the firmware to the home directory, example below uses curl directly from the webpage

### 2) Upload the firmware into the system

While the .bin file may be on the system, It needs to be uploaded and processed by the system to make it accessible. Using the following command line on the FW file from 1)

```
sudo firmware_install.lua --upload <full firware filename>
```

NOTE: the firmware filename must be exactly as downloaded, e.g. no (1) or other suffix appended.

Below example filename is "fmadio20v3\_20210831\_1136.bin"

Example upload process



```
fmadio@fmadio20v3-287:~$ sudo firmware_install.lua --upload fmadio20v3_20210831_1136.bin
fmad fmadlua Aug 31 2021
calibrating...
0 : 2095081812          2.0951 cycles/nsec offset:4.918 Mhz
Cycles/Sec 2095081812.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
argv /usr/local/bin/fmadiolua
argv fmadio20v3_20210831_1136.bin
loading filename [/opt/fmadio/bin/firmware_install.lua]
Upload Firmware [fmadio20v3_20210831_1136.bin]
mkdir: can't create directory '/mnt/store0/tmp/fw/': File exists
Filename [fmadio20v3_20210831_1136]
unpack FW
unpack: [fmadio20v3_20210831_1136.core]
unpack: [fmadio20v3_20210831_1136.kernel]
unpack: [fmadio20v3_20210831_1136.mydata]
unpack: [fmadio20v3_20210831_1136.post.lua]
unpack: [fmadio20v3_20210831_1136.pre.lua]
unpack: [fmadio20v3_20210831_1136.rom.2x10G]
unpack: [fmadio20v3_20210831_1136.rom.2x1G]
unpack: [fmadio20v3_20210831_1136.sign]
unpack: [fmadio20v3_20210831_1136.syslinux.cfg]
unpack: [fmadio20v3_20210831_1136.syslinux.cfg.analytics]
unpack: [fmadio20v3_20210831_1136.tcz]
check TCZ
check sign
verify signature
sign: gpg: WARNING: unsafe ownership on homedir `/home/fmadio/.gnupg/'
sign: 5c154ff9bc902ac6d4d44b1fa0d60f33  fmadio20v3_20210831_1136.tcz
sign: b6cad8c08e51fd80b412b6c1602af4b4  fmadio20v3_20210831_1136.pre.lua
sign: 180fe4e475b521f7bf36f7f9d2acc6c8  fmadio20v3_20210831_1136.post.lua
sign: 31d280c2c5585b25d0585a67a1021853  fmadio20v3_20210831_1136.core
sign: a98281879ffc8040dc0a78f1669cb647  fmadio20v3_20210831_1136.kernel
sign: 4fa3ff6d2087f9797cad6aee2df0aea9  fmadio20v3_20210831_1136.mydata
sign: gpg: Signature made Tue Aug 31 11:37:07 2021 JST using RSA key ID 35173534
sign: gpg: checking the trustdb
sign: gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
sign: gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
sign: gpg: Good signature from "fmadio <support@fmad.io>" [ultimate]
Found Signatures:
fmadio20v3_20210831_1136.tcz : 5c154ff9bc902ac6d4d44b1fa0d60f33  fmadio20v3_20210831_1136.tc
fmadio20v3_20210831_1136.post.lua : 180fe4e475b521f7bf36f7f9d2acc6c8  fmadio20v3_20210831_11
fmadio20v3_20210831_1136.kernel : a98281879ffc8040dc0a78f1669cb647  fmadio20v3_20210831_1136
fmadio20v3_20210831_1136.mydata : 4fa3ff6d2087f9797cad6aee2df0aea9  fmadio20v3_20210831_1136
fmadio20v3_20210831_1136.core : 31d280c2c5585b25d0585a67a1021853  fmadio20v3_20210831_1136.c
fmadio20v3_20210831_1136.pre.lua : b6cad8c08e51fd80b412b6c1602af4b4  fmadio20v3_20210831_113
Signatures good
[fmadio20v3_20210831_1136.tcz ] Expect:(5c154ff9bc902ac6d4d44b1fa0d60f33  fmadio20v3_202108
[fmadio20v3_20210831_1136.post.lua] Expect:(180fe4e475b521f7bf36f7f9d2acc6c8  fmadio20v3_202
[fmadio20v3_20210831_1136.kernel] Expect:(a98281879ffc8040dc0a78f1669cb647  fmadio20v3_20210
[fmadio20v3_20210831_1136.mydata] Expect:(4fa3ff6d2087f9797cad6aee2df0aea9  fmadio20v3_20210
[fmadio20v3_20210831_1136.core ] Expect:(31d280c2c5585b25d0585a67a1021853  fmadio20v3_202108
[fmadio20v3_20210831_1136.pre.lua] Expect:(b6cad8c08e51fd80b412b6c1602af4b4  fmadio20v3_2021
```



```
Firmware is valid  
Firmware Copy Took 1.473866 sec  
Firmware Update Complete  
done 18.199007Sec 0.303317Min
```

### 3) Firmware Install

Next the firmware needs to be installed using the command

```
$ sudo firmware_install.lua --install <firmware filename.bin>
```

Example output is shown below. Note if the capture is running as shown, it will eventually timeout and complete the update.



```
done 0.000086Sec 0.000001Min
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.tcz /mnt/nvme0n1p1//tce/optional/
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.core /mnt/nvme0n1p1//boot/fmadio2
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.kernel /mnt/nvme0n1p1//boot/vmlin
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.mydata /mnt/nvme0n1p1//tce/mydata
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.syslinux.cfg /mnt/nvme0n1p1//boot
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.rom.2x1G /mnt/nvme0n1p1//boot/bit
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.rom.2x10G /mnt/nvme0n1p1//boot/bi
Bitstream Config [2x10G]
Copy [cp /mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.rom.2x10G /mnt/nvme0n1p1//boot/bi
/mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.post.lua
fmad fmadlua Jul 21 2021
calibrating...
0 : 2100016641          2.1000 cycles/nsec offset:0.017 Mhz
Cycles/Sec 2100016641.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
argv /opt/fmadio/bin/fmadiolua
loading filename [/mnt/nvme0n1p1//firmware/fmadio20v2_20210906_1606.post.lua]
PostInstall: post install script
PostInstall: POSTINSTALL_GOOD
done 0.003273Sec 0.000055Min
Firmware Install Complete
done 34.044320Sec 0.567405Min
fmadio@fmadio20v2-149:$
```

#### 4) Reboot

System needs to be rebooted, It will power cycle also after the 1st reboot to complete the update process.

Example

```
$ sudo reboot
Connection to 192.168.2.145 closed by remote host.
```

# Firmware Revert (CLI)

FMADIO Systems use a RAM based linux distribution, where all persistent files have to be explicitly specified. This approach allows easy forward and backward changing of the system firmware.

Forward updates to new FW is documented below



Firmware Update (CLI)

Below is how to revert to a previous image.

## Step 1

Check if the reverted firmware is already uploaded on the system

```
ls -al /mnt/system/firmware/*.bin
```

Example output shown below

```
fmadio@fmadio100v2-228U:~$ ls -al /mnt/system/firmware/*.bin
-rwxr-xr-x  1 root  root  197094389 Mar  1 12:13 /mnt/system/firmware/fmadio100v2_20
-rwxr-xr-x  1 root  root  459983681 Feb 21 15:00 /mnt/system/firmware/fmadio100v2_20
-rwxr-xr-x  1 root  root  488447559 Feb 21 15:51 /mnt/system/firmware/fmadio100v2_20
-rwxr-xr-x  1 root  root  491484052 Mar  3 13:10 /mnt/system/firmware/fmadio100v2_20
-rwxr-xr-x  1 root  root  471198901 Mar  7 19:26 /mnt/system/firmware/fmadio100v2_20
-rwxr-xr-x  1 root  root  491383334 Mar  9 16:50 /mnt/system/firmware/fmadio100v2_20
fmadio@fmadio100v2-228U:~$
```

If the firmware file is shown with correct MD5 sum skip to Step 3

## Step 2

Upload the target firmware onto the system. There is no harm in overwriting an existing firmware file. In the below example using firmware FW:8475 fmadio100v2\_20230307\_1925.bin

```
sudo firmware_install.lua --upload fmadio100v2_20230307_1925.bin
```

Example output similar to this

```
fmadio@fmadio100v2-228U:~$ sudo firmware_install.lua --upload fmadio100v2_20230307_1925.bin
fmad fmadlua Mar  9 2023 (/usr/local/bin/fmadiolua /opt/fmadio/bin/firmware_install.lua --up
Upload Firmware [fmadio100v2_20230307_1925.bin]
mkdir: can't create directory '/mnt/store0/tmp/fw/': File exists
Filename [fmadio100v2_20230307_1925]
unpack FW
unpack: [fmadio100v2_20230307_1925.core]
unpack: [fmadio100v2_20230307_1925.kernel]
unpack: [fmadio100v2_20230307_1925.mydata]
unpack: [fmadio100v2_20230307_1925.post.lua]
unpack: [fmadio100v2_20230307_1925.pre.lua]
unpack: [fmadio100v2_20230307_1925.rom.2x100G]
unpack: [fmadio100v2_20230307_1925.rom.2x100G_replay]
unpack: [fmadio100v2_20230307_1925.rom.2x40G]
unpack: [fmadio100v2_20230307_1925.rom.8x10G]
unpack: [fmadio100v2_20230307_1925.sign]
unpack: [fmadio100v2_20230307_1925.syslinux.cfg]
unpack: [fmadio100v2_20230307_1925.syslinux.cfg.analytics]
unpack: [fmadio100v2_20230307_1925.tcz]
check TCZ
check sign
verify signature
sign: gpg: WARNING: unsafe ownership on homedir `/home/fmadio/.gnupg/'
sign: fe54748ff985ec1ebc118a985d459b1a  fmadio100v2_20230307_1925.tcz
sign: b6cad8c08e51fd80b412b6c1602af4b4  fmadio100v2_20230307_1925.pre.lua
sign: 180fe4e475b521f7bf36f7f9d2acc6c8  fmadio100v2_20230307_1925.post.lua
sign: 7f1b7577ab5d836e870bb7045586efd0  fmadio100v2_20230307_1925.core
sign: ed6276d1815c6b8ce808dd1ec6f14d5d  fmadio100v2_20230307_1925.kernel
sign: 3d1a0e22186d43592c7718ef6a1eb0d0  fmadio100v2_20230307_1925.mydata
sign: gpg: Signature made Tue Mar  7 19:25:09 2023 SGT using RSA key ID 35173534
sign: gpg: checking the trustdb
sign: gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
sign: gpg: depth: 0  valid:   1  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 1u
sign: gpg: Good signature from "fmadio <support@fmad.io>" [ultimate]
Found Signatures:
fmadio100v2_20230307_1925.tcz : fe54748ff985ec1ebc118a985d459b1a  fmadio100v2_20230307_1925.
fmadio100v2_20230307_1925.pre.lua : b6cad8c08e51fd80b412b6c1602af4b4  fmadio100v2_20230307_1
fmadio100v2_20230307_1925.post.lua : 180fe4e475b521f7bf36f7f9d2acc6c8  fmadio100v2_20230307_
fmadio100v2_20230307_1925.mydata : 3d1a0e22186d43592c7718ef6a1eb0d0  fmadio100v2_20230307_19
fmadio100v2_20230307_1925.kernel : ed6276d1815c6b8ce808dd1ec6f14d5d  fmadio100v2_20230307_19
fmadio100v2_20230307_1925.core : 7f1b7577ab5d836e870bb7045586efd0  fmadio100v2_20230307_1925
Signatures good
[fmadio100v2_20230307_1925.tcz ] Expect:(fe54748ff985ec1ebc118a985d459b1a  fmadio100v2_20230
[fmadio100v2_20230307_1925.pre.lua] Expect:(b6cad8c08e51fd80b412b6c1602af4b4  fmadio100v2_20
[fmadio100v2_20230307_1925.post.lua] Expect:(180fe4e475b521f7bf36f7f9d2acc6c8  fmadio100v2_2
[fmadio100v2_20230307_1925.mydata] Expect:(3d1a0e22186d43592c7718ef6a1eb0d0  fmadio100v2_202
[fmadio100v2_20230307_1925.kernel] Expect:(ed6276d1815c6b8ce808dd1ec6f14d5d  fmadio100v2_202
[fmadio100v2_20230307_1925.core] Expect:(7f1b7577ab5d836e870bb7045586efd0  fmadio100v2_20230
Firmware is valid
Firmware Copy Took 1.845648 sec
Check fmadio100v2_20230307_1925.tcz          Expect:          120856576 Found:
Check fmadio100v2_20230307_1925.pre.lua     Expect:          87 Found:
```

```
Check fmadio100v2_20230307_1925.post.lua          Expect:          754 Found:
Check fmadio100v2_20230307_1925.mydata          Expect:          21573 Found:
Check fmadio100v2_20230307_1925.kernel          Expect:          4943968 Found:
Check fmadio100v2_20230307_1925.core            Expect:          314395612 Found:
Firmware Update Complete
done 9.954817Sec 0.165914Min
fmadio@fmadio100v2-228U:~$
```

### Step 3

Next install the firmware image as follows

```
sudo firmware_install.lua --install fmadio100v2_20230307_1925.bin
```

Example output looks similar to this

```

fmadio@fmadio100v2-228U:~$ sudo firmware_install.lua --install fmadio100v2_20230307_1925.bin
fmad fmadlua Mar  9 2023 (/usr/local/bin/fmadiolua /opt/fmadio/bin/firmware_install.lua --in
Open fSysCapture_t* [/opt/fmadio/status/capture:0/100]
FW [fmadio100v2] System[fmadio100v2]
PreInstall script [/mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.pre.lua]
fmad fmadlua Mar  9 2023 (/opt/fmadio/bin/fmadiolua /mnt/nvme0n1p1//firmware/fmadio100v2_202
calibrating...
0 : 2095079878          2.0951 cycles/nsec offset:4.920 Mhz
Cycles/Sec 2095079878.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
done 0.000041Sec 0.000001Min
PreInstall: pre install script
PreInstall: PREINSTALL_GOOD
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.tcz /mnt/nvme0n1p1//tce/optional
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.core /mnt/nvme0n1p1//boot/fmadio
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.kernel /mnt/nvme0n1p1//boot/vmli
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.mydata /mnt/nvme0n1p1//tce/mydat
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.syslinux.cfg /mnt/nvme0n1p1//boc
Bitstream Config [2x40G]
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.rom.2x100G /mnt/nvme0n1p1//boot/
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.rom.2x40G /mnt/nvme0n1p1//boot/b
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.rom.8x10G /mnt/nvme0n1p1//boot/b
Copy [cp /mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.rom.2x40G /mnt/nvme0n1p1//boot/b
os[sudo /opt/fmadio/bin/bitstream_update.lua --noreboot --write /mnt/nvme0n1p1//boot/bitstre
fmad fmadlua Mar  9 2023 (/opt/fmadio/bin/fmadiolua /opt/fmadio/bin/bitstream_update.lua --n
calibrating...
0 : 2095079660          2.0951 cycles/nsec offset:4.920 Mhz
Cycles/Sec 2095079660.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
cp: '/mnt/nvme0n1p1//boot/bitstream.rom' and '/mnt/nvme0n1p1//boot/bitstream.rom' are the sa
done 0.136745Sec 0.002279Min
/mnt/nvme0n1p1//firmware/fmadio100v2_20230307_1925.post.lua
fmad fmadlua Mar  9 2023 (/opt/fmadio/bin/fmadiolua /mnt/nvme0n1p1//firmware/fmadio100v2_202
calibrating...
0 : 2095082226          2.0951 cycles/nsec offset:4.918 Mhz
Cycles/Sec 2095082226.0000 Std:          0 cycle std( 0.00000000) Target:2.10 Ghz
PostInstall: post install script
done 0.001404Sec 0.000023Min
PostInstall: POSTINSTALL_GOOD
Firmware Install Complete
done 4.144247Sec 0.069071Min
fmadio@fmadio100v2-228U:~$

```

#### Step 4

Reboot the system, it make take up to 5 minutes. The system will reboot 2 times, once for fpga update, and again for the final pass.

#### Step 5

Verify the firmware version is correct

```
cat /opt/fmadio/version
```

Example output

```
fmadio@fmadio100v2-228U:~$ cat /opt/fmadio/version
fmadio100v2
8545
Thu Mar  9 16:49:20 2023
fmadio@fmadio100v2-228U:~$
```

## Finished

Its possible when difference in firmware version numbers is extremely large there may be additional steps. Please contact [support@fmad.io](mailto:support@fmad.io) for assistance in such cases.