



# L2X Configuration Guide

Version 21.1.1, 29 January 2021

<b>Registered Address</b>	<b>Support</b>	<b>Sales</b>
26, Kingston Terrace, Princeton, New Jersey 08540, United States		
		+91 80 4850 5445
<a href="http://www.rtbrick.com">http://www.rtbrick.com</a>	<a href="mailto:support@rtbrick.com">support@rtbrick.com</a>	<a href="mailto:sales@rtbrick.com">sales@rtbrick.com</a>

©Copyright 2021 RtBrick, Inc. All rights reserved. The information contained herein is subject to change without notice. The trademarks, logos and service marks ("Marks") displayed in this documentation are the property of RtBrick in the United States and other countries. Use of the Marks are subject to RtBrick's Term of Use Policy, available at <https://www.rtbrick.com/privacy>. Use of marks belonging to other parties is for informational purposes only.

# Table of Contents

1. Introduction	3
1.1. Port and VLAN Cross-connects	3
1.2. L2X 802.1ad Ethertype Support	3
2. L2X Configuration Examples	4
2.1. Ingress VLAN Operation	4
2.2. Supported Match Type Validations	4
2.3. L2X Ingress Configuration Examples	6
2.3.1. Port and Any VLAN (ifp, any_vlan)	7
2.3.2. Port and Untagged VLAN (ifp, untagged)	10
2.3.3. Port and Outer VLAN (ifp, outer_vlan)	12
2.3.4. Port and Dual VLAN Tags (ifp, outer_vlan, inner_vlan)	15
2.3.5. Port + Outer VLAN + Inner VLAN Any (ifp, outer_vlan, any inner_vlan)	18
2.3.6. L2X with Exit Interface	21
2.4. Egress Interface Action	22
2.5. Support for Bidirectional Local Cross Connect	23
2.6. Deleting L2x Configuration	24
2.7. Clearing Statistics	25
2.7.1. Clearing L2X statistics	25
3. Viewing the L2X Configuration	26
3.1. Summary Commands	26
3.1.1. show l2x	26
3.1.2. show l2x <l2x-name>	26
3.1.3. show l2x type <type>	27
3.1.4. show l2x type <type> <l2x-name>	27
3.1.5. show l2x direction <direction>	28
3.1.6. show l2x direction <direction> <l2x-name>	28
3.2. Detail commands	29
3.2.1. show l2x detail	29
3.2.2. show l2x <l2x-name> detail	29
3.2.3. show l2x type <type> <l2x-name> detail	30
3.2.4. show l2x direction <direction> <l2x-name> detail	31
3.3. L2X Statistics Commands	31
3.3.1. show l2x statistics	31
3.3.2. show l2x statistics <l2x-name>	32
3.3.3. L2X clear statistics	33
3.3.4. clear l2x statistics	33
3.3.5. clear l2x statistics <l2x-name>	33

# 1. Introduction

Layer 2 Cross-Connect (L2X) is a data plane feature that connects two physical ports (IFPs) using Layer 2 switching. At the simplest, L2X can switch all the traffic between two IFPs to provide the trunk service of an Ethernet switch.

## 1.1. Port and VLAN Cross-connects

Both types of L2X switches Layer 2 traffic from input interface to output interface. The difference is that a port cross-connect switches all Layer 2 traffic arriving at an input interface, but a VLAN cross-connect only switches the Layer 2 traffic associated with a specific VLAN. A port-based L2X means a port-only configuration, so there are no VLANs involved.

Both single-tag and double-tagged (inner and outer VLAN tags) are supported. The port and VLAN L2X support both local and remote L2X configurations. In remote L2X connections, the VLAN cross-connects are typically configured on the MPLS tunnel ingress router.

Untagged traffic on L2X interfaces is also supported. However, there is no way to select only untagged traffic for cross-connecting. Therefore, only port crossconnects are supported for untagged traffic.

## 1.2. L2X 802.1ad Ethertype Support

RBFS supports VLAN-operations like VLAN add, VLAN swap, and VLAN delete on egress interface. RBFS supports similar functionality at the ingress side as well. That is, RBFS supports the below VLAN operations:

- Single-VLAN-Add with an option to configure encapsulation (that is, 802.1q or 802.1ad)
- Single-VLAN-Delete
- Swap-Outer-VLAN

By default the encapsulation method is 802.1q. If an encapsulation method is not specified, the 802.1q will be the default mode.

In addition to setting the Ethertype for a VLAN operation, the 802.1ad support includes that ingress traffic for all tagged match options will match on both Ethertype 0x8100 (802.1q) and 0x8A88 (802.1ad) by default.

## 2. L2X Configuration Examples

These examples cover the various possibilities for VLAN value editing in local L2X.

### 2.1. Ingress VLAN Operation

RBFS supports VLAN-operations like VLAN add, VLAN swap and VLAN delete on ingress interface.

The current functionality has been extended all existing CLIs to accept ingress vlan-operation and ingress VLAN encapsulation value.

Both 802.1q and 802.1ad encapsulations are supported. The default encapsulation is 802.1q.

Traffic will be matched at ingress direction based on the match criterion. RtBrick Full Stack (RBFS) supports below match parameters.

There can be five different match types on a physical interface, that is, traffic can be matched based on the following:

1. (ifp)
2. (ifp, outer\_vlan)
3. (ifp, outer\_vlan, inner\_vlan)
4. (ifp, outer\_vlan, any inner\_vlan)
5. (ifp, any vlan)

A few modes in above list are mutually exclusive (for example (ifp, outer\_vlan, inner\_vlan) and (ifp, outer\_vlan, any inner\_vlan) configuration on the same interface is conflicting configuration).



The **match-type** attribute is mandatory for **match-untagged**, **match-any** and **match-inner-any** match criterion.

For more information, see the [L2X Ingress Configuration Examples](#) section below.

### 2.2. Supported Match Type Validations

**Case1 : IFP A, \***



The asterisk (\*) indicates *any or no vlan tags*.

Configuration A	Configuration B	Support
-----------------	-----------------	---------

IFP A, *	IFP A, ov 10	Not supported
IFP A, *	IFP A, ov 10, iv 20	Not supported
IFP A, *	IFP A, ov 10, *	Not supported
IFP A, *	IFP A, untagged	Not Supported

**Case 2: IFP A, untagged:**

Configuration A	Configuration B	Support
IFP A, untagged	IFP A, *	Not supported
IFP A, untagged	IFP A, ov 10	supported
IFP A, untagged	IFP A, ov 30, iv 20	Supported
IFP A, untagged	IFP A, ov 20, *	Supported

**Case 3: IFP A, outer\_vlan:**

Configuration A	Configuration B	Support
IFP A, ov 10	IFP A, *	Not supported
IFP A, ov 10	IFP A, ov 10, *	Not supported
IFP A, ov 10	IFP A, ov 20	Supported
IFP A, ov 10	IFP A, ov 10 , iv 20	Not supported
IFP A, ov 10	IFP A, ov 40 , iv 7	supported
IFP A, ov 10	IFP A, ov 30, *	supported
IFP A, ov 10	IFP A, untagged	supported

**Case 4: IFP A, outer\_vlan, inner\_vlan:**

Configuration A	Configuration B	Support
IFP A,ov 10, iv 20	IFP A, *	Not supported
IFP A,ov 10, iv 20	IFP A, ov 10, *	Not supported
IFP A,ov 10, iv 20	IFP A, ov 10	Not supported
IFP A,ov 10, iv 20	IFP A, ov 30	Supported
IFP A,ov 10, iv 20	IFP A, ov 20 , *	supported
IFP A,ov 10, iv 20	IFP A, untagged	supported
IFP A,ov 10, iv 20	IFP A, ov 10, iv 30	supported

**Case 5: IFP A, outer\_vlan, \***



The asterisk (\*) indicates *any* or *no vlan* tags.

Configuration A	Configuration B	Support
IFP A,ov 10, *	IFP A, *	Not supported
IFP A,ov 10, *	IFP A, ov 10	Not supported
IFP A,ov 10, *	IFP A, ov 10, iv 20	Not supported
IFP A,ov 10, *	IFP A, ov 20, iv 7	Supported
IFP A,ov 10, *	IFP A, ov 30	supported
IFP A,ov 10, *	IFP A, untagged	Supported
IFP A,ov 10, *	IFP A, ov 40, *	supported

## 2.3. L2X Ingress Configuration Examples

The following command shows how to configure ingress VLAN operation (Local Cross-Connect)

```

set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> match-type <match-type>

set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>

set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-action>

set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> ingress-vlan-encapsulation <encapsulation>

set l2x name <l2x-name> <direction> outgoing-interface <outgoing-interface>

set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>

set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> egress-vlan-encapsulation <encapsulation>

```

The following command shows how to configure ingress VLAN operation (remote

Cross-Connect)

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> match-type <match-type>

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-
action>

set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> ingress-vlan-encapsulation
<encapsulation>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>
```



For ingress L2x either outgoing-interface or nexthop is mandatory.

### 2.3.1. Port and Any VLAN (ifp, any vlan)

You can run the following commands in this mode:

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> match-type match-any

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> outgoing-interface <outgoing-
interface>
```



```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> match-type match-any
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-interface>
```

```
set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>
```

```
set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> match-type match-any
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>
```

```
set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>
```

```
set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>
```

```
set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>
```

```
set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>
```

```
set l2x name <l2x-name> <direction> service-label <service_label>
```

```

set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> match-type match-any

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-
action>

set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> ingress-vlan-encapsulation
<encapsulation>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>

```

## Command Arguments

Command Argument	Description
<incoming-interface>	Incoming interface is where the traffic originates
<outgoing-interface>	Outgoing interface traffic is going to
<vlan-action>	Indicates the VLAN action such as Single-Vlan-Add, Single-Vlan-Delete, or Swap-Outer-Vlan
<vlan_id>	VLAN ID value (2 to 4095)
<encapsulation>	Specifies the encapsulation value, that is, 802.1q or 802.1ad
<nexthop>	Next-Hop address
<lookup_instance>	Instance name
<lookup_afi>	AFI value: ipv4 or ipv6
<lookup_safi>	SAFI value: safi values are unicast, labeled-unicast
<service_label>	Service label value

## Configuration Example

```
set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress match-type match-any

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress match-type match-any
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 1000
set l2x name test4 ingress match-type match-any
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 200

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 2000
set l2x name test4 ingress match-type match-any
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Delete
```

### 2.3.2. Port and Untagged VLAN (ifp, untagged)

You can run the following commands in this mode:

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> match-type match-untagged

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> outgoing-interface <outgoing-
interface>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> match-type match-untagged
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-interface>
```

```
set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>
```

```
set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> match-type match-untagged
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>
```

```
set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-action>
```

```
set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> ingress-vlan-encapsulation <encapsulation>
```

```
set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>
```

```
set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>
```

```
set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>
```

```
set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>
```

```
set l2x name <l2x-name> <direction> service-label <service_label>
```



For (IFP, untagged) match criteria, the **match\_type** attribute is mandatory when you perform L2X configuration using curl.

## Command Arguments

See the [Command Arguments](#) section.

## Configuration Example

```
set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress match-type match-untagged

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress match-type match-untagged
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 1000
set l2x name test4 ingress match-type match-untagged
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 200
```

### 2.3.3. Port and Outer VLAN (ifp, outer\_vlan))

You can run the following commands in this mode:

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> outgoing-interface <outgoing-
interface>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-interface>
```

```
set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>
```

```
set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-interface>
```

```
set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>
```

```
set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>
```

```
set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>
```

```
set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>
```

```
set l2x name <l2x-name> <direction> service-label <service_label>
```

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-
action>

set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> ingress-vlan-encapsulation
<encapsulation>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>
```

### Command Arguments

See the [Command Arguments](#) section.

### Configuration Example

```
set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 100
set l2x name test4 ingress outgoing-interface ifp-0/0/8

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 200
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 300
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 1000
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 400
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 2000
```

### 2.3.4. Port and Dual VLAN Tags (ifp, outer\_vlan, inner\_vlan)

You can run the following commands in this mode:



```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> incoming-inner-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-  
interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-  
interface>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> incoming-inner-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-  
interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-  
interface>
```

```
set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>
```

```
set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> incoming-inner-vlan <vlan-id>

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>
```

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> incoming-inner-vlan <vlan-id>

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-
action>

set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> ingress-vlan-encapsulation
<encapsulation>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>
```

## Command Arguments

See the [Command Arguments](#) section.

## Configuration Example

```
set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 100
set l2x name test4 ingress incoming-inner-vlan 500
set l2x name test4 ingress outgoing-interface ifp-0/0/8

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 200
set l2x name test4 ingress incoming-inner-vlan 600
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 300
set l2x name test4 ingress incoming-inner-vlan 700
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 1000
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 400
set l2x name test4 ingress incoming-inner-vlan 800
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 2000
```

### 2.3.5. Port + Outer VLAN + Inner VLAN Any (ifp, outer\_vlan, any inner\_vlan)

You can run the following commands in this mode:

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> match-type match-inner-any
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-  
interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-  
interface>
```

```
set l2x name <l2x-name> <direction>
```

```
set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction> match-type match-inner-any
```

```
set l2x name <l2x-name> <direction> incoming-interface <incoming-  
interface>
```

```
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-  
interface>
```

```
set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>
```

```
set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>
```

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> match-type match-inner-any

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>
```

```
set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> incoming-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> match-type match-inner-any

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> ingress-vlan-operation <ingress-vlan-
action>

set l2x name <l2x-name> <direction> ingress-outer-vlan <vlan-id>

set l2x name <l2x-name> <direction> ingress-vlan-encapsulation
<encapsulation>

set l2x name <l2x-name> <direction> nexthop4/nexthop6 <nexthop>

set l2x name <l2x-name> <direction> lookup-instance <lookup-instance>

set l2x name <l2x-name> <direction> lookup-afi <lookup-afi>

set l2x name <l2x-name> <direction> lookup-safi <lookup-safi>

set l2x name <l2x-name> <direction> service-label <service_label>
```

## Command Arguments

See the [Command Arguments](#) section.

## Configuration Example

```

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 100
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress match-type match-inner-any

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 200
set l2x name test4 ingress outgoing-interface ifp-0/0/8
set l2x name test4 ingress match-type match-inner-any
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 200
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 1000
set l2x name test4 ingress match-type match-inner-any
set l2x name test4 ingress ingress-vlan-operation Single-Vlan-Add
set l2x name test4 ingress ingress-outer-vlan 100

set l2x name test4 ingress
set l2x name test4 ingress incoming-interface ifp-0/0/4
set l2x name test4 ingress incoming-outer-vlan 400
set l2x name test4 ingress nexthop4 10.1.1.2
set l2x name test4 ingress lookup-instance default
set l2x name test4 ingress lookup-afi ipv4
set l2x name test4 ingress lookup-safi labeled-unicast
set l2x name test4 ingress service-label 2000

```

### 2.3.6. L2X with Exit Interface



This configuration is applicable only for link-local nexthop6 address.

To configure L2X with exit interface, enter the following command:

#### Syntax

```
set l2x name <l2x-name> <direction> exit-interface <exit-interface>
```

### Command Arguments

Command Argument	Description
<l2x-name>	Name of the Layer 2 Cross-Connect
<direction>	Indicates the direction of the traffic, that is egress or ingress
<exit-interface>	Name of the exit interface.

### Example

```
set l2x name test4 ingress exit-interface if1-0/0/52/13
```

## 2.4. Egress Interface Action

In the Egress direction, incoming-interface is not required, and RBFS supports the below commands.

```
set l2x name <l2x-name> <direction>  
  
set l2x name <l2x-name> <direction> service-label <service_label>  
  
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-  
interface>
```

```
set l2x name <l2x-name> <direction>  
  
set l2x name <l2x-name> <direction> service-label <service_label>  
  
set l2x name <l2x-name> <direction> outgoing-interface <outgoing-  
interface>  
  
set l2x name <l2x-name> <direction> egress-vlan-operation <vlan-action>  
  
set l2x name <l2x-name> <direction> outgoing-outer-vlan <vlan-id>
```

### Command Arguments

See the [Command Arguments](#) section.

## Configuration Example

```

set l2x name test4 egress
set l2x name test4 egress service-label 1234
set l2x name test4 egress outgoing-interface ifp-0/0/8

set l2x name test4 egress
set l2x name test4 egress service-label 1234
set l2x name test4 egress outgoing-interface ifp-0/0/8
set l2x name test4 egress egress-vlan-operation Single-Vlan-Add
set l2x name test4 egress outgoing-outer-vlan 100

```

## 2.5. Support for Bidirectional Local Cross Connect

In the earlier releases of RBFS, to establish cross connection between two ports, there was a need to install two l2x configurations, that is, one connection for the ifp1 to ifp2 and another connection for ifp2 to to ifp1.

With bidirectional cross connect, you can establish cross connection between two local ports say ifp1 ↔ ifp2 with single l2x configuration.

To support this feature, the existing L2X commands are extended to accept “bi-directional” direction attribute, and this bi-directional attribute is applicable to only for local cross connect. If bi-directional direction is used for remote cross connect (that is, if nexthop4, nexthop6 or service-label is present) configuration will fail with an error.



The VLAN operations are not supported for bi-directional local cross connect.

### Syntax

```

set l2x name <l2x-name> <direction>

set l2x name <l2x-name> <direction> match-type <match-type>

set l2x name <l2x-name> <direction> incoming-interface <incoming-
interface>

set l2x name <l2x-name> <direction> outgoing-interface <outgoing-
interface>

```

### Example



```
set l2x name test4 bi-directional
set l2x name test4 bi-directional incoming-interface ifp-0/0/4
set l2x name test4 bi-directional outgoing-interface ifp-0/0/8
set l2x name test4 bi-directional match-type match-any
```

The following example shows the show running-configuration in JSON format.

```
{
  "data": {
    "rtbrick-config:l2x": {
      "name": [
        {
          "name": "test4",
          "direction": "bi-directional",
          "incoming-interface": "ifp-0/0/4",
          "outgoing-interface": "ifp-0/0/8",
          "match-type": "match-any"
        }
      ]
    }
  }
}
```

## 2.6. Deleting L2x Configuration

To delete the L2X configuration, enter the following command:

### Syntax

```
delete l2x name <l2x-name> <direction>
```

### Command Arguments

Command Argument	Description
<l2x-name>	Name of the Layer 2 Cross-Connect
<direction>	Indicates the direction of the traffic, that is egress or ingress

### Example

```
delete l2x name test4 bi-directional
```

## 2.7. Clearing Statistics

By clearing the L2X statistics and interface physical counters, you can reset all of the previous packet statistics and counters.

### 2.7.1. Clearing L2X statistics

To clear the statistics, enter the following command:

```
rtb fibd clear bcm l2x statistics
```

## 3. Viewing the L2X Configuration

The L2X show commands display data from FIB local table. Therefore local L2X with down ports or remote l2x with unresolved nexthop address are not displayed.

### 3.1. Summary Commands

The summary commands display L2X information in tabular format. Key information will be displayed in the summary output.



Since the availability of space is less in the summary output, the L2X name will be truncated after certain length. In such cases, you can refer detail command output where full name will be printed.

#### 3.1.1. show l2x

This command displays all l2x configuration information such as ingress l2x and egress l2x.

##### Syntax

```
show l2x
```

##### Example

```
supervisor@rtbrick: op> show l2x
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
                  Inner VLAN      Service label
test2
-                  -               egress     -                   ifp-0/0/4
-                  -               1234
test1
Any                Any             ingress    ifp-0/0/4          ifp-0/0/10
```

#### 3.1.2. show l2x <l2x-name>

This command displays the L2X configuration of the specified l2x-name.

##### Syntax

```
show l2x <l2x-name>
```

## Example

```
supervisor@rtbrick: op> show l2x test2
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
test2              -               egress    -                  ifp-0/0/4
-                  -               1234
```

### 3.1.3. show l2x type <type>

This command displays the L2X configuration of the specified type, where the type can be local or remote.

#### Syntax

```
show l2x type <type>
```

## Example

```
supervisor@rtbrick: op> show l2x type local
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
test1              -               ingress   ifp-0/0/4          ifp-0/0/10
Any                Any              -
```

```
supervisor@rtbrick: op> show l2x type remote
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
test2              -               egress    -                  ifp-0/0/4
-                  -               1234
```

### 3.1.4. show l2x type <type> <l2x-name>

This command displays the L2X information of the specified type l2x-name.

#### Syntax

```
show l2x type <type> <l2x-name>
```

## Example

```
supervisor@rtbrick: op> show l2x type remote test2
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
test2              -               egress     -                   ifp-0/0/4
-                  -               1234
```

### 3.1.5. show l2x direction <direction>

This command displays L2X information of the specified direction, where direction can be ingress, egress or bi-directional.

#### Syntax

```
show l2x direction <direction>
```

#### Example

```
supervisor@rtbrick: op> show l2x direction ingress
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
test1              -               ingress    ifp-0/0/4          ifp-0/0/10
Any                Any              -
```

### 3.1.6. show l2x direction <direction> <l2x-name>

This command displays L2X information of the specified direction and l2x-name.

#### Syntax

```
show l2x direction <direction> <l2x-name>
```

#### Example

```
supervisor@rtbrick: op> show l2x direction egress test2
Name
Intf/Next Hop      Outer VLAN      Direction  Incoming Intf      Outgoing
test2              -               egress     -                   ifp-0/0/4
-                  -               1234
```

## 3.2. Detail commands

The following commands provided detailed information L2X.

### 3.2.1. show l2x detail

Similar to the summary output, this command displays all L2X information in detailed format.

#### Syntax

```
show l2x detail
```

#### Example

```
supervisor@rtbrick: op> show l2x detail
L2X name: test2
  Direction: egress
  Outgoing interface: ifp-0/0/4
  Service label: 1234
  Subtype: Service Label Match
  Incoming outer VLAN: -
  Incoming inner VLAN: -
  Ingress vlan operation:
  NextHop:
    NextHop type: Remote egress cross connect
    NextHop action: No vlan manipulation - l2 forward
  Egress vlan operation:
L2X name: test1
  Direction: ingress
  Incoming interface: ifp-0/0/4
  Outgoing interface: ifp-0/0/10
  Subtype: Incoming Port - Any Vlan Match
  Incoming outer VLAN: Any
  Incoming inner VLAN: Any
  Ingress vlan operation:
  NextHop:
    NextHop type: Local egress cross connect
    NextHop action: No vlan manipulation - l2 forward
  Egress vlan operation:
```

### 3.2.2. show l2x <l2x-name> detail

This command displays detailed l2x information of the specified l2x-name.

#### Syntax

```
show l2x <l2x-name> detail
```

## Example

```
supervisor@rtbrick: op> show l2x test1 detail
L2X name: test1
  Direction: ingress
  Incoming interface: ifp-0/0/4
  Outgoing interface: ifp-0/0/10
  Subtype: Incoming Port - Any Vlan Match
  Incoming outer VLAN: Any
  Incoming inner VLAN: Any
  Ingress vlan operation:
  NextHop:
    NextHop type: Local egress cross connect
    NextHop action: No vlan manipulation - l2 forward
  Egress vlan operation:
```

### 3.2.3. show l2x type <type> <l2x-name> detail

This command displays detailed L2X information of the specified type and l2x-name.

## Syntax

```
show l2x type <type> <l2x-name> detail
```

## Example

```
supervisor@rtbrick: op> show l2x type local test1 detail
L2X name: test1
  Direction: ingress
  Incoming interface: ifp-0/0/4
  Outgoing interface: ifp-0/0/10
  Subtype: Incoming Port - Any Vlan Match
  Incoming outer VLAN: Any
  Incoming inner VLAN: Any
  Ingress vlan operation:
  NextHop:
    NextHop type: Local egress cross connect
    NextHop action: No vlan manipulation - l2 forward
  Egress vlan operation:
```

### 3.2.4. show l2x direction <direction> <l2x-name> detail

This command displays detailed L2X information of the specified direction and l2x-name.

#### Syntax

```
show l2x direction <direction> <l2x-name> detail
```

#### Example

```
supervisor@rtbrick: op> show l2x direction egress test2 detail
L2X name: test2
  Direction: egress
  Outgoing interface: ifp-0/0/4
  Service label: 1234
  Subtype: Service Label Match
  Incoming outer VLAN: -
  Incoming inner VLAN: -
  Ingress vlan operation:
  NextHop:
    NextHop type: Remote egress cross connect
    NextHop action: No vlan manipulation - l2 forward
  Egress vlan operation:
```

## 3.3. L2X Statistics Commands

The following commands enable you to verify l2x statistics.

### 3.3.1. show l2x statistics

This command displays statistics of all the installed L2X.

#### Syntax

```
show l2x statistics
```

#### Example



```
supervisor@rtbrick: op> show l2x statistics

L2X Name: l2x-test1/0
  Physical Interface Name: ifp-0/0/4
  Logical Interface Type: L2x ingress vlan interface
  Port-Mapping-Core: 0
  Vlan-Port-ID: 1149251592
  MPLS-Port-ID: N/A
  Counters:
    In-Forward-Packets: 57
    In-Forward-Bytes: 5700
    In-Drop-Packets: 0
    In-Drop-Bytes: 0
    Out-Forward-Packets: 0
    Out-Forward-Bytes: 0
    Out-Drop-Packets: 0
    Out-Drop-Bytes: 0
L2X Name: l2x-d3b529d74770f91fb2acf5e38da70eb9213473dd7e996c6a
  Physical Interface Name: ifp-0/0/10
  Logical Interface Type: L2x egress vlan interface
  Port-Mapping-Core: 0
  Vlan-Port-ID: 1149251591
  MPLS-Port-ID: N/A
  Counters:
    In-Forward-Packets: 0
    In-Forward-Bytes: 0
    In-Drop-Packets: 0
    In-Drop-Bytes: 0
    Out-Forward-Packets: 0
    Out-Forward-Bytes: 0
    Out-Drop-Packets: 0
    Out-Drop-Bytes: 0
```

### 3.3.2. show l2x statistics <l2x-name>

This command displays statistics of the specified l2x-name.

#### Syntax

```
show l2x statistics <l2x-name>
```

#### Example

```
supervisor@rtbrick: op> show l2x statistics l2x-test1/0
L2X Name: l2x-test1/0
  Physical Interface Name: ifp-0/0/4
  Logical Interface Type: L2x ingress vlan interface
  Port-Mapping-Core: 0
  Vlan-Port-ID: 1149251592
  MPLS-Port-ID: N/A
  Counters:
    In-Forward-Packets: 61
    In-Forward-Bytes: 6100
    In-Drop-Packets: 0
    In-Drop-Bytes: 0
    Out-Forward-Packets: 0
    Out-Forward-Bytes: 0
    Out-Drop-Packets: 0
    Out-Drop-Bytes: 0
```

### 3.3.3. L2X clear statistics

The following commands allow you to clear L2X statistics.

### 3.3.4. clear l2x statistics

This command allows you to clear all L2X statistics.

#### Syntax

```
clear l2x statistics
```

#### Example

```
supervisor@rtbrick: op> clear l2x statistics
```

### 3.3.5. clear l2x statistics <l2x-name>

This command allows you to clear the statistics of the specified l2x-name.

#### Syntax

```
clear l2x statistics <l2x-name>
```

#### Example

```
supervisor@rtbrick: op> clear l2x statistics l2x-test1/0
```