Securing the Control Plane User Guide

Version 22.7.1, 25 July 2022

Registered Address	Support	Sales
26, Kingston Terrace, Princeton, New Jersey 08540, United States		
		+91 80 4850 5445
http://www.rtbrick.com	support@rtbrick.com	sales@rtbrick.com

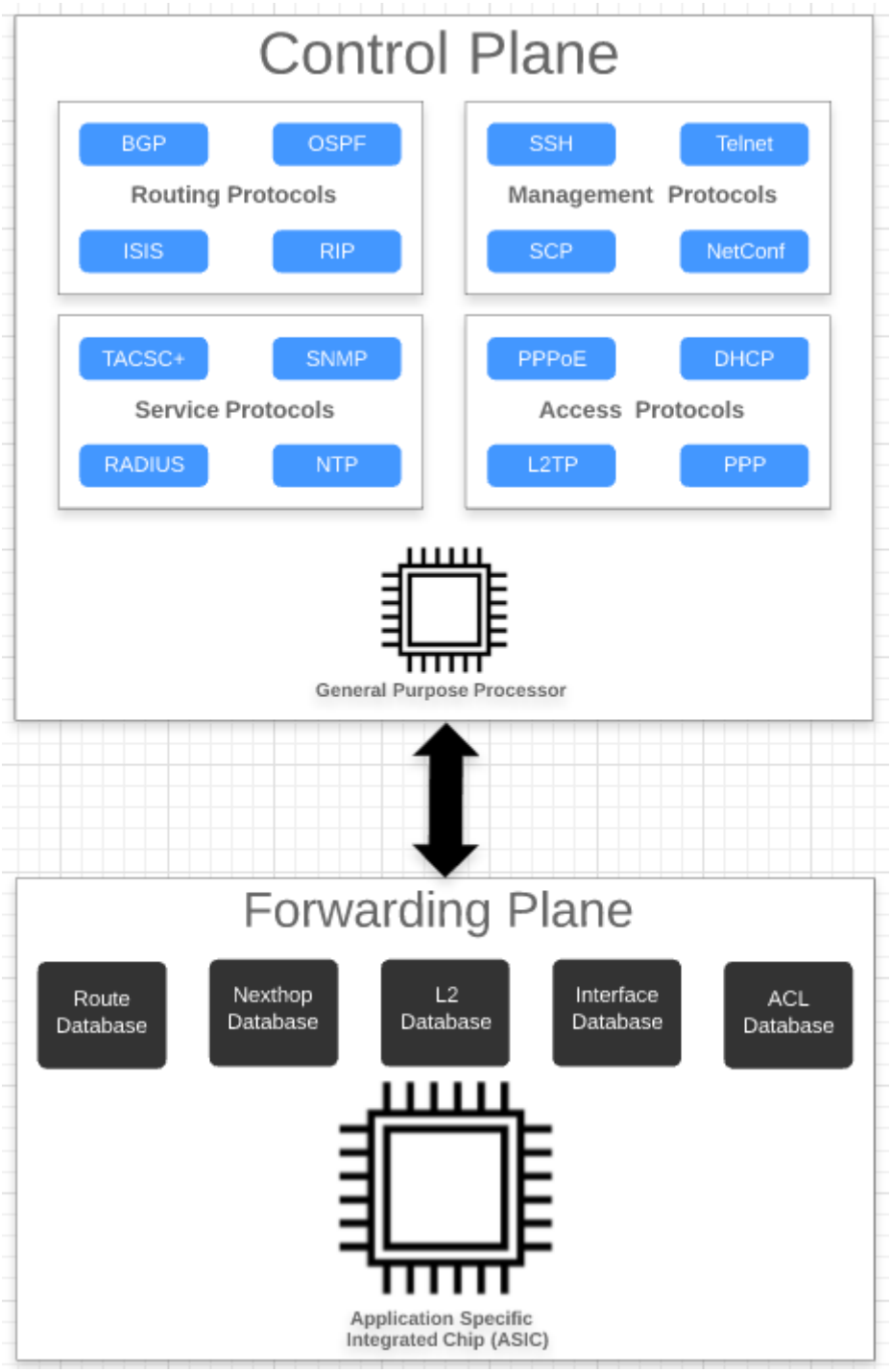
Table of Contents

1. Introduction

1.1. Control Plane Traffic

Control plane security enables you to filter or rate-limit unwanted traffic that is transmitted from the forwarding plane to the control plane. In RBFS, you can use Access Control Lists (ACL) and policers to secure the router's control plane.

All routing protocols, management protocols, service protocols run in the control plane. The output of these protocols result in certain databases like routing table, MAC table, ARP table, etc., which eventually get programmed in the forwarding plane.



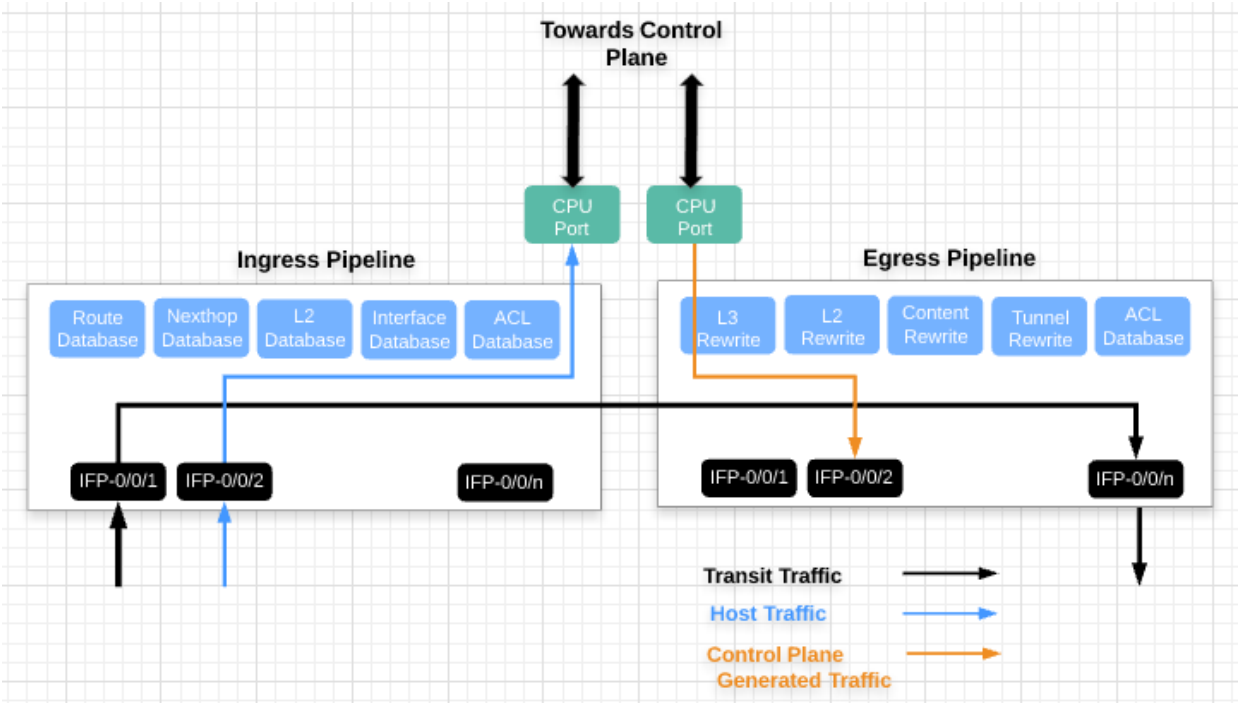
In the diagram above, you see the routing protocols (BGP, OSPF, ISIS), management protocols (SSH, RESTCONF, etc), service protocols (RADIUS, NTP, TACACS+), and access protocols (PPPoE, DHCP, L2TP, PPP) associated with the control plane. The control plane is generally implemented in software by using general-purpose processors. These protocols typically build a large number of databases like routing, switching, and ACL tables.

In contrast, a forwarding plane is associated with a copy of the databases (Routing, Switching, ACL, etc) built by the control plane. These entries typically contain the match and action which decide the packet flow in the forwarding plane. The forwarding plane functionality is realized in high performance Application Specific

Integrated Circuits (ASICs) that are capable of handling very high packet rates.

There are two kinds of traffic:

1. **Control traffic**: Control traffic is destined to the device itself, that means, packets are handled by the router itself. The traffic is classified as control traffic based on matching destination IP, or because of ACL rules, or because of some kind of exception that occurred while parsing the packet (non-acceptable fields, TTL expiry, etc).
2. **Transit traffic**: Transit traffic not destined to the device itself. These packets will be sent out on one of the routers physical interfaces.



All control traffic packets will be destined to the CPU port(s). These packets are redirected to the control plane for further processing. However, general purpose processors in the control plane are not designed for packet processing, and might get overloaded if the rate of control plane traffic is too high, for example caused by a DDoS attack. Therefore you should to protect the router control plane by implementing mechanisms to filter completely or rate-limit traffic not required or unwanted at the control plane level.

1.2. Securing the Control Plane

In RBFS, there are two fundamental mechanisms how control-plane traffic is redirected to the CPU:

1. Via protocol ACLs
2. Via route lookup

Both mechanisms need to be considered and secured separately, as described in

the following sections.

1.2.1. Control Plane Traffic via Protocol ACLs

All routing protocols (BGP, OSPF, ISIS), management protocols (SSH, RESTCONF, etc), service protocols (RADIUS, NTP, TACACS+), and access protocols (PPPoE, DHCP, L2TP, PPP), if enabled by configuration, automatically create Access Control Lists (ACLs) required to punt the protocol traffic to the control plane CPU. ACLs are the building block for securing the control plane. An ACL defines a rule, which typically contains match conditions and actions. If a packet matches the rule conditions, the associated actions will be applied. Protocol ACLs do not need to be defined by configuration. Another benefit is, they are very specific, for example match on auto-discovered IPv6 link-local neighbors. The protocol ACLs can verified using the 'show acl (detail)' command.

Example 1: Protocol ACL created by LLDP

```
supervisor@rtbrick: op> show acl detail
Rule: lldp.ifp-0/0/1.trap.rule
  ACL type: l2
  Ordinal: -
    Match:
      Attachment point: ifp-0/0/1
      Direction: ingress
      Destination MAC: 01:80:c2:00:00:0e
    Action:
      Redirect to CPU: True
    Result:
      Trap ID: LLDP
<...>
```

Example 2: Protocol ACL created by RADIUS

```
supervisor@rtbrick: op> show acl detail
Rule: radius-srv1-v4-auth-trap
  ACL type: l3v4
  Ordinal: -
    Match:
      Source L4 port: 1812
      IP protocol: UDP
    Action:
      Redirect to CPU: True
    Result:
      Trap ID: Radius
<...>
```

By default, for most of the control protocols, there is a single action Redirect to CPU: True. Thereby all traffic matching the match criteria gets punted to the CPU without any rate limit. There is one exception, for PPPoE only the traffic is rate-limited by default. As shown in the following example, there is an additional action

Policer profile name: created by default that limits the PPPoE traffic to 50 Mbps per session:

Example 3: Protocol ACL with Policer created by PPPoE

```
supervisor@rtbrick: op> show acl detail
Rule: pppoed_hostif-0/0/1_7-7-1-4090_8863
  ACL type: PPPOE
  Ordinal: -
    Match:
      Attachment point: hostif-0/0/1
      Ethertype: 34915
    Action:
      Redirect to CPU: True
      Policer profile name: _DEFAULT_POLICER_50_MB
    Result:
      Trap ID: PPPoE
<...>
```

For all other protocols, rate limiting needs to enabled by configuration in order to secure the control plane. This is described in section 2.1 below.

1.2.2. Control Plane Traffic via Route Lookup

By default, any other traffic destined to one of the router's IP addresses, commonly referred to as "my IP", and not matching any ACL is redirected to the CPU via a route lookup. This applies to loopback as well physical interface addresses. In order to secure the control plane against malicious traffic sent to one of the router's IP addresses (not matching any protocol ACL), ACLs need to be defined by configuration. In RFBS, such "manually" created ACLs are referred to user-defined ACLs.

Typically you will want to completely block some unwanted traffic sent to "my IP", but allow and rate-limit some required traffic like for example ICMP. Please note, when designing the security ACLs to protect "my IP", you do NOT need to consider the protocol traffic already handled by the protocol ACLs.

When configuring ACLs, protocol ACLs and user-defined ACLs may conflict. For example, an ACLs created by the BGP routing protocol might match on TCP traffic sent to the routers loopback address with port 179. A user-defined ACL however might deny any traffic sent to this loopback address. In this case, protocol ACLs shall take precedence over user-defined ACLs, so that you do not accidentally break the protocol operation. In RBFS, this is implemented using different ACL database priorities.

Configuring ACLs to protect "my IP" is described in section 2.2 below.

1.3. Limitations and Notes

- The control-plane security features are supported on hardware platforms. They are not supported on virtual deployments.
- On the Edgecore AS5916-54XKS platform, BGPv6 with link-local peering uses route lookup instead of protocol ACLs. Therefore traffic sent to IPv6 link local addresses cannot be restricted via ACL to not break BGPv6 link-local peerings.

1.4. Supported Platforms

Not all features are necessarily supported on each hardware platform. Refer to the *Platform Guide* for the features and the sub-features that are or are not supported by each platform.

Control Plane security is currently not supported for PIM, IGMP, and L2TP protocol traffic.

2. Configuring Control Plane Security

As highlighted above, there are two fundamental mechanisms how traffic is redirected to the CPU, via protocol ACLs, and via route lookup. These are addressed in following two sections.

2.1. Secure Control Plane Traffic via Protocol ACLs

This section describes the configuration options for control plane traffic that is redirected to the CPU via protocol ACLs. These ACLs are automatically created by the protocols, and do not need to be - and cannot be - configured manually. For example if you configure a routing protocol like BGP, the required ACLs to match and punt the BGP packets to the control plane are created automatically.

2.1.1. Enabling the Control Plane Security Feature

By default, all packets matching the protocol ACLs will be sent to the control plane without any rate limit, except for PPPoE. The RBFS Control Plane Security feature allows to add policers to all protocol ACLs. If enabled, this feature creates a set of default policers, and applies them to the protocol ACLs. Thereby the control plane gets secured against DDoS attacks matching these ACLs.

Syntax:

set forwarding-options control-plane-security <attribute> <value>

Attribute	Description
state (enable|disable)	Enable or disable the control-plane security feature. Default: disabled.

Example:

```
{
    "rtbrick-config:forwarding-options": {
      "control-plane-security": {
        "state": "enable"
      }
    }
}
```

2.1.2. Configuring Host Path QoS

The host-path-qos enable feature is disabled by default. Once it is enabled, you cannot disable it.

To enable the `host-path-qos' feature, enter the following command:

Syntax:

set forwarding-options class-of-service control-plane-qos ingress-qos <attribute> <value>

Attribute	Description
state (enable|disable)	Enable or disable the host path QoS feature. Default: disabled.

Example

```
supervisor@rtbrick>LEAF01: cfg> show config forwarding-options class-of-
service control-plane-qos
{
    "rtbrick-config:control-plane-qos": {

      "ingress-qos": {
        "state": "enable"
      }
    }
  }
```

By enabling this feature, the default scheduler and queue configurations are installed and the CPU ports queue mapping will change. Also, all control plane ACLs will be reprogrammed to update action_forward_class.

2.1.3. Marking Outbound Control Plane Traffic

RBFS enables you to configure the various protocols to mark the egress control plane traffic. The control plane traffic can be marked with type-of-service (ToS) values.

- For BGP, OSPF, RADIUS, PIM, L2TPv2, and DHCP protocols, the remark-type should be configured as ToS
- For the IGMP and PPPoE protocols, the remark-type can be be configured as p-bit or tos
- The outbound-marking attributes such as name, code-point and remark-type are mandatory
- If the name of the protocol is L2-all, then the remark-type should configured as p-bit
- If the name of the protocol is L3-all, then the remark-type should configured as tos

set forwarding-options class-of-service control-plane-qos outbound-marking protocol <protocol-name> <remark-type-value> **codepoint** <codepoint-value>

Option	Description
<protocol-name>	Specifies the protocol name.
<remark-type-value>	Specifies the remark type value that can be p-bit or tos.
<codepoint-value>	Specifies the codepoint value. The supported range for p-bit outbound-marking is 0-7.

Example: Marking Outbound CP Traffic Configuration

```
supervisor@rtbrick>LEAF01: cfg> show config
{
  "ietf-restconf:data": {
    "rtbrick-config:forwarding-options": {
      "class-of-service": {
        "control-plane-qos": {
          "outbound-marking": {
            "protocol": [
              {
                "protocol": "bgp",
                "remark-type": "tos",
                "codepoint": 192
              },
              {
                "protocol": "dhcp",
                "remark-type": "p-bit",
                "codepoint": 5
              },
              {
                "protocol": "l3-all",
                "remark-type": "tos",
                "codepoint": 224
              },
                    {
                "protocol": "igmp",
                "remark-type": "p-bit",
                "codepoint": 7
              },
              {
                "protocol": "igmp",
                "remark-type": "tos",
                "codepoint": 192
              },
              {
                "protocol": "ppp",
                "remark-type": "p-bit",
                "codepoint": 7
              },
              {
                "protocol": "ppp",
                "remark-type": "tos",
                "codepoint": 192
              },
              {
                "protocol": "radius",
                "remark-type": "tos",
                "codepoint": 100
            }
            ]
          }
        }
      }
    }
  }
}
```

2.1.4. Restricting Management Access

If you enable inband management access for example via SSH, protocol ACLs will be created that match on the enabled protocols and redirect the management traffic to the control plane. By default, this traffic is not restricted in terms of source IP addresses. You can optionally restrict management access to trusted IP addresses by applying a source prefix list. An additional match condition will then be added to the protocol ACLs for inband management.

2.1.4.1. Configuring a Prefix List

Syntax:

set forwarding-options prefix-list <options>

Option	Description
<prefix-list-name> ipv4-prefix <ipv4_prefix>	Prefix list configuration for IPv4.
<prefix-list-name> ipv6-prefix <ipv6_prefix>	Prefix list configuration for IPv6.

2.1.4.2. Applying a Prefix List

Syntax:

set inband-management instance <instance-name> source-prefix-list <list-name>

Example: Inband Management Configuration with Source Prefix List

```
"rtbrick-config:inband-management": {
      "instance": [
        {
          "name": "default",
          "ssh": "true",
          "ntp": "true",
          "source-prefix-list": "list1"
        }
      ]
    },

"rtbrick-config:forwarding-options": {
      "prefix-list": [
        {
          "prefix-list-name": "list1",
          "ipv4-prefix": [
            {
              "ipv4-prefix": "100.100.100.100/32"
            },
            {
              "ipv4-prefix": "3.3.3.3/32"
            },
            {
              "ipv4-prefix": "4.4.4.4/32"
            }
          ]
        }
```

The inband management is provided to only the source address specified in the prefix list if the prefix list is configured. If the prefix list not configured, it works for all source IPs. Also, the prefix addresses configured should be of /32.

2.1.5. Configuring Protocol ACL Options

This section describes how to configure policers per protocol and configure match on IPv4 ToS or IPv6 TC fields for protocol ACLs.

- If control plane security is disabled, this configuration has no effect in the system
- Protocol-specific configuration will take priority over ALL configuration in the control-plane-security protocol

Syntax:

set forwarding-options control-plane-security protocol <protocol-name> <attribute> <value>

Option	Description
protocol <protocol-name>	Name of the protocol. You can configure individual protocols, and/or all protocols using the 'ALL' value keyword.
match-tc <tc-value>	Configure IPv6 TC value. The range is 0 to 248.
match-tos <tos-value>	Configure IPv4 ToS value. The range is 0 to 248.
policer <policer>	Configure policer name.

Example:

```
{
  "ietf-restconf:data": {
    "rtbrick-config:forwarding-options": {
      "class-of-service": {
        "policer": [
          {
            "policer-name": "_DEFAULT_POLICER_BGP_LL",
            "flags": "color-blind",
            "level1-rates": {
              "cir": 1000,
              "cbs": 1000,
              "pir": 1200,
              "pbs": 1000
            },
            "levels": 1,
            "type": "two-rate-three-color"
          }
        ]
      },
      "control-plane-security": {
        "state": "enable",
        "protocol": [
          {
            "protocol": "PIM",
            "policer": "_DEFAULT_POLICER_PIM",
            "match-tos": 192,
            "match-tc": 120
          }
        ]
      }
    }
  }
}
```

2.2. Secure Control Plane Traffic via Route Lookup

This section describes the configuration to secure the control plane for traffic that is redirected to the CPU via route lookup. Any packet sent to one of the router's IP addresses ("my IP") and not matching any ACL, will be redirected to the CPU via route lookup. By default this type of traffic is not restricted or rate-limited. In order

to secure the control plane, you need to apply ACLs by configuration. Please note you do not need to consider and allow any protocol traffic that is already captured by the automatically created protocol ACLs. You only need to explicitly define rules for any other traffic sent to "my IP". In the simplest case, you can deny any other traffic sent to the router. Typically you will want to allow some additional traffic like ICMP, and deny anything else.

2.2.1. Configuring ACLs

This section describes how to configure ACLs to secure the control plane to protect "my IP". In RFBS, ACLs are applied globally, that is, you do not need to attach them by configuration. Besides, for ACLs matching traffic sent to one of the router's IP addresses, the redirect-to-cpu action applies implicitly and does not need to be configured.

Syntax:

set forwarding-options acl <options>

Option	Description
l3v4	ACL configuration for IPv4
l3v6	ACL configuration for IPv6
rule <rule-name>	Name of the ACL rule
ordinal <ordinal-value>	Number of the configuration entry. Please note the order of the configuration entries (ordinals) does not determine the processing.
match <condition>	Supported match conditions are IP source/destination prefix, prefix lists, source/destination Port, IP protocol, and direction.
action <action>	Supported actions are permit, drop, and police.
priority <value>	ACL entry priority. Determines the processing precedence for multiple matching i.e. conflicting rules. A less-specific rule should have a lower priority so that a more-specific rules takes precedence. Default: 10.

Example 1: Denying any Traffic destined to the Router's Loopback Addresses

```
supervisor@rtbrick>LEAF01: cfg> show config forwarding-options acl
{
    "rtbrick-config:acl": {
      "l3v4": {
        "rule": [
          {
            "rule-name": "BLOCK_LOOPBACK_TRAFFIC_v4",
            "ordinal": [
              {
                "ordinal-value": 20,
                "match": {
                  "destination-ipv4-prefix": "192.1.4.3/32",
                  "direction": "ingress"
                },
                "action": {
                  "drop": "true"
                },
                "priority": 20
              }
            ]
          }
        ]
      },
      "l3v6": {
        "rule": [
          {
            "rule-name": "BLOCK_LOOPBACK_TRAFFIC_v6",
            "ordinal": [
              {
                "ordinal-value": 20,
                "match": {
                  "destination-ipv6-prefix": "192:1:4::3/128",
                  "direction": "ingress"
                },
                "action": {
                  "drop": "true"
                },
                "priority": 20
              }
            ]
          }
        ]
      }
    }
  }
supervisor@rtbrick>LEAF01: cfg>
```

Example 2: ACL allowing and rate-limiting ICMPv4/v6, and denying any other Traffic

```
supervisor@rtbrick>LEAF01: cfg> show config forwarding-options acl
{
    "rtbrick-config:acl": {
      "l3v4": {
        "rule": [
          {
            "rule-name": "BLOCK_LOOPBACK_TRAFFIC_v4",
            "ordinal": [
              {
                "ordinal-value": 20,
                "match": {
                  "destination-ipv4-prefix": "192.1.4.3/32",
                  "direction": "ingress",
                  "ip-protocol": "ICMP"
                },
                "action": {
                  "permit": "true"
                },
                "priority": 20
              }
            ]
          }
        ]
      },
      "l3v6": {
        "rule": [
          {
            "rule-name": "BLOCK_LOOPBACK_TRAFFIC_v6",
            "ordinal": [
              {
                "ordinal-value": 20,
                "match": {
                  "destination-ipv6-prefix": "192:1:4::3/128",
                  "direction": "ingress",
                  "ip-protocol": "IPv6_ICMP"
                },
                "action": {
                  "permit": "true"
                },
                "priority": 20
              }
            ]
          }
        ]
      }
    }
  }
```

3. Operations

3.1. Show Commands

This section describes operational commands available to verify various control-plane security features.

3.1.1. Verifying ACLs

The show acl command allows to verify protocol ACLs as well as user-defined ACLs.

Syntax:

show acl <options>

Option	Description
detail	Displays all ACL details
<acl-name>	Displays the details for a single ACL

Example 1: Protocol ACL with Control-Plane Security enabled

```
supervisor@rtbrick>LEAF01: op>  show acl detail

Rule: lldp.ifp-0/0/1.trap.rule
  ACL type: l2
  Ordinal: -
    Match:
      Attachment point: ifp-0/0/1
      Direction: ingress
      Destination MAC: 01:80:c2:00:00:0e
    Action:
      Redirect to CPU: True
      Policer profile name: _DEFAULT_POLICER_50_MB
    Result:
      Trap ID: LLDP
<...>
Rule: radius-srv1-v4-auth-trap
  ACL type: l3v4
  Ordinal: -
    Match:
      Source L4 port: 1812
      IP protocol: UDP
    Action:
      Redirect to CPU: True
      Policer profile name: _DEFAULT_POLICER_20_MB
    Result:
      Trap ID: Radius
<...>
```

Example 2: ACL for Inband Management with Source Prefix List

```
supervisor@rtbrick>LEAF01: op>  show acl detail

Rule: ifm.inband.mgmt.lo-0/0/0/1.ssh.client.v4.trap.rule.1
  ACL type: l3v4
  Ordinal: 1
    Match:
      Destination IPv4 address: 10.99.0.1
      Source IPv4 address: 10.99.0.2
      Source L4 port: 22
      IP protocol: TCP
    Action:
      Redirect to CPU: True
    Result:
      Trap ID: INBAND
```

Example 3: User-defined ACL to Protect "my IP"

```
supervisor@rtbrick>LEAF01: op> show acl Protect-CP-v4

Rule: Protect-CP-v4
  ACL type: l3v4
  Ordinal: 1
    Match:
      Direction: ingress
      Destination IPv4 prefix: 10.99.0.1/32
      Source IPv4 prefix: 10.99.0.0/24
      IP protocol: ICMP
    Action:
      Permit: True
    Result:
      Trap ID: User Defined
  Ordinal: 2
    Match:
      Direction: ingress
      Destination IPv4 prefix: 10.99.0.1/32
    Action:
      Drop: True
    Priority: 5
    Result:
      Trap ID: User Defined
```

3.1.2. Verifying ACL Counters

The "show acl statistics" command displays information about the ACL packet counters. The counters are useful to verify if the ACL rules actually match, and if potentially malicious traffic gets dropped.

Syntax:

show acl statistics

Example 1: ACL statistics information

```
supervisor@rtbrick>LEAF01: cfg> show acl statistics
ACL
Units      Total       Accepted    Dropped
lldp.ifp-0/0/12.trap.rule
Packets    -           -           -

Bytes      -           -           -
lldp.ifp-0/0/16.trap.rule
Packets    -           -           -

Bytes      -           -           -
lldp.ifp-0/0/27.trap.rule
Packets    -           -           -

Bytes      -           -           -
lldp.ifp-0/0/53.trap.rule
Packets    -           -           -

Bytes      -           -           -
default_bgp_l4_trap_12::2_12::1_dst
Packets    12          12          0

Bytes      1353        1353        0
default_bgp_l4_trap_12::2_12::1_src
Packets    12          12          0

Bytes      1353        1353        0
default_bgp_l4_trap_12.0.0.2_12.0.0.1_dst
Packets    12          12          0

Bytes      1353        1353        0
default_bgp_l4_trap_12.0.0.2_12.0.0.1_dst
Packets    -           -           -

Bytes      -           -           -
default_bgp_l4_trap_12.0.0.2_12.0.0.1_src
Packets    12          12          0

Bytes      1353        1353        0
default_bgp_l4_trap_12.0.0.2_12.0.0.1_src
Packets    -           -           -

Bytes      -           -           -
supervisor@rtbrick: cfg>
```

Example 2: Display ACL statistics information for the specified ACL

```
supervisor@rtbrick>LEAF01: cfg> show acl
default_bgp_l4_trap_12.0.0.2_12.0.0.1_dst statistics
ACL
Units      Total       Accepted    Dropped
default_bgp_l4_trap_12.0.0.2_12.0.0.1_dst
Packets    20          20          0

Bytes      1917        1917        0
default_bgp_l4_trap_12.0.0.2_12.0.0.1_dst
Packets    -           -           -

Bytes      -           -           -
supervisor@rtbrick>LEAF01: cfg>
```

3.1.3. Verifying Control Plane Policers

This command allows to view the policers created by the control-plane security feature.

Syntax:

show qos policer <options>

Option	Description
-	Displays all policers created by the control-plane security feature
<policer-name>	Displays information about the specified policer
counter	Displays all policer counters

Example 1: Display information of all policers created by the control-plane security feature

```
supervisor@rtbrick>LEAF01: cfg> show qos policer
Policer: _DEFAULT_POLICER_100_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        100000         100000         33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
Policer: _DEFAULT_POLICER_1_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
```

```
  1        1000           1000           33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
Policer: _DEFAULT_POLICER_20_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        20000          20000          33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
Policer: _DEFAULT_POLICER_250_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        250000         250000         33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
Policer: _DEFAULT_POLICER_500_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        500000         500000         33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
Policer: _DEFAULT_POLICER_50_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        50000          50000          33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
Policer: _DEFAULT_POLICER_5_MB
Active: True, Type: two-rate-three-color, Levels: 1, Flags: -
```

```
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        5000           5000           33000          33000          -
-
  2        -              -              -              -              -
-
  3        -              -              -              -              -
-
  4        -              -              -              -              -
-
supervisor@rtbrick>LEAF01: cfg>
```

Example 2: Display information of a specific policer

```
supervisor@rtbrick>LEAF01: cfg> show qos policer
Premium_Upstream_Hierarchical_Policer
Policer: Premium_Upstream_Hierarchical_Policer
Active: False, Type: two-rate-three-color, Levels: 4, Flags: color-blind
  Level    CIR(Kbps)      PIR(Kbps)      CBS(KB)        PBS(KB)        Max
CIR(Kbps)  Max PIR(Kbps)
  1        1000           1200           1000           1000           -
-
  2        900            1000           1000           1000           -
-
  3        5000           5200           1000           1000           -
-
  4        6000           6200           1000           1000           -
-
```

Example 3: Display information of policer counter

```
supervisor@rtbrick>LEAF01: cfg> show qos policer counter
Interface                          Level  Units      Total            Received
Dropped
ipv6_ll_prefix_acl                 1      Packets    48               48
0
                                          Bytes      6383             6383
0
ipv6_mcast_ff01_prefix_acl         1      Packets    48               48
0
                                          Bytes      6383             6383
0
ipv6_mcast_ff02_prefix_acl         1      Packets    48               48
0
                                          Bytes      6383             6383
0
ppp-0/1/28/72339069014638594       1      Packets    0                0
0
                                          Bytes      0                0
0
ppp-0/1/28/72339069014638594       2      Packets    0                0
0
                                          Bytes      0                0
0
ppp-0/1/28/72339069014638594       3      Packets    0                0
0
                                          Bytes      0                0
0
ppp-0/1/28/72339069014638594       4      Packets    0                0
0
                                          Bytes      0                0
0
pppoed_ifp-0/1/28_1-3500-1-35      1      Packets    48               48
0
                                          Bytes      6383             6383
0
pppoed_ifp-0/1/28_1-3500-1-35      1      Packets    48               48
0
                                          Bytes      6383             6383
0
pppoed_ifp-0/1/30_1-3500-1-35      1      Packets    48               48
0
                                          Bytes      6383             6383
0
pppoed_ifp-0/1/30_1-3500-1-35      1      Packets    48               48
0
                                          Bytes      6383             6383
0
```

The show qos policer counter command displays the policer-level counters for the subscribers. The packets that get dropped after the RPF check, are currently updated in the local.bcm.q2c.trap.stats table in FIBD.