COMMAND REFERENCE GUIDE

Extreme Virtual TAP 2.0.0 Command Line Reference

9035941-00
13 February 2019

Legal Notice

Extreme Networks, Inc. reserves the right to make changes in specifications and other information contained in this document and its website without prior notice. The reader should in all cases consult representatives of Extreme Networks to determine whether any such changes have been made.

The hardware, firmware, software or any specifications described or referred to in this document are subject to change without notice.

Trademarks

Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries.

All other names (including any product names) mentioned in this document are the property of their respective owners and may be trademarks or registered trademarks of their respective companies/owners.

For additional information on Extreme Networks trademarks, please see: www.extremenetworks.com/company/legal/trademarks

Open Source Declarations

Some software files have been licensed under certain open source or third-party licenses. End-user license agreements and open source declarations can be found at: www.extremenetworks.com/support/policies/software-licensing

Contents

Preface

This section discusses the conventions used in this guide, ways to provide feedback, additional help, and other Extreme Networks® publications.

Conventions

This section discusses the conventions used in this guide.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used to highlight specific words or phrases.

Format	Description
bold text	Identifies command names.
	Identifies keywords and operands.
	Identifies the names of GUI elements.
	Identifies text to enter in the GUI.
italic text	Identifies emphasis.
	Identifies variables.
	Identifies document titles.

Format	Description
`Courier font`	Identifies CLI output. Identifies command syntax examples.

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Documentation and Training

To find Extreme Networks product guides, visit our documentation pages at:

Current Product Documentation	www.extremenetworks.com/documentation/
Archived Documentation (for earlier versions and legacy products)	www.extremenetworks.com/support/documentation-archives/
Release Notes	www.extremenetworks.com/support/release-notes
Hardware/Software Compatibility Matrices	https://www.extremenetworks.com/support/compatibility-matrices/
White papers, data sheets, case studies, and other product resources	https://www.extremenetworks.com/resources/

Training

Extreme Networks offers product training courses, both online and in person, as well as specialized certifications. For more information, visit www.extremenetworks.com/education/.

Getting Help

If you require assistance, contact Extreme Networks using one of the following methods:

Extreme Portal	Search the GTAC (Global Technical Assistance Center) knowledge base, manage support cases and service contracts, download software, and obtain product licensing, training, and certifications.
The Hub	A forum for Extreme Networks customers to connect with one another, answer questions, and share ideas and feedback. This community is monitored by Extreme Networks employees, but is not intended to replace specific guidance from GTAC.
Call GTAC	For immediate support: 1-800-998-2408 (toll-free in U.S. and Canada) or +1 408-579-2826. For the support phone number in your country, visit: www.extremenetworks.com/support/contact

Before contacting Extreme Networks for technical support, have the following information ready:

- Your Extreme Networks service contract number and/or serial numbers for all involved Extreme Networks products
- A description of the failure
- A description of any action(s) already taken to resolve the problem
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any related RMA (Return Material Authorization) numbers

Subscribing to Service Notifications

You can subscribe to email notifications for product and software release announcements, Vulnerability Notices, and Service Notifications.

1. Go to www.extremenetworks.com/support/service-notification-form.
2. Complete the form with your information (all fields are required).
3. Select the products for which you would like to receive notifications.

   NOTE
   You can modify your product selections or unsubscribe at any time.

4. Click **Submit**.

Providing Feedback to Us

Quality is our first concern at Extreme Networks, and we have made every effort to ensure the accuracy and completeness of this document. We are always striving to improve our documentation and help you work better, so we want to hear from you! We welcome all feedback but especially want to know about:

- Content errors or confusing or conflicting information.
- Ideas for improvements to our documentation so you can find the information you need faster.
- Broken links or usability issues.

If you would like to provide feedback to the Extreme Networks Information Development team, you can do so in two ways:

- Use our short online feedback form at https://www.extremenetworks.com/documentation-feedback/.
- Email us at documentation@extremenetworks.com.

Please provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

Command-Line interface

Extreme Virtual TAP (vTAP) provides a Command-Line Interface (CLI) that supports several administrative operations. This chapter provides information about the CLI commands supported by vTAP.

Accessing Command-Line Interface

You can access vTAP's Command-Line Interface either through Telnet or SSH. A secure username and password is required for both modes of communication.

NOTE

When using `root` account, use your Linux password as the password.

Accessing Command-Line Interface using SSH

To access Command-Line Interface using SSH:

1. Run the following command to check the status of the CLI service:

   ```
   systemctl status vtap_cli.service
   ```

   If the CLI service is running, the status appears as `Active: active (running)`. If the CLI service not running, the status appears as `Active: inactive (dead)`.

   Example

   ```
   [root@8129 ~]# systemctl status vtap_cli.service
   ● vtap_cli.service - vTAP CLI
      Loaded: loaded (/etc/systemd/system/vtap_cli.service; enabled; vendor preset: disabled)
      Active: active (running) since Thu 2018-10-04 18:28:46 IST; 24h ago
     Process: 10835 ExecStop=/usr/lib/systemd/scripts/vtap_cli.sh stop (code=exited, status=0/SUCCESS)
     Process: 10855 ExecStart=/usr/lib/systemd/scripts/vtap_cli.sh start (code=exited, status=0/SUCCESS)
    Main PID: 10859 (vtap_cli)
      CGroup: /system.slice/vtap_cli.service
              └─10859 /sbin/vtap_cli

   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     registering callback = set_terminal_length
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]: return, priv, Return to previous prompt
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]: action type = cfunction, handler = return_exec
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]: testpcap, priv, start-stop sample pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     0, start, (null), starting sample pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     sub-command: action type = cfunction,
   handler = start_send_pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     registering callback = start_send_pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     0, stop, (null), stopping sample pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     sub-command: action type = cfunction,
   handler = stop_send_pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     registering callback = stop_send_pcap
   ```

2. Run the following command:

   ```
   ssh -p 5000 <vtap_ip_address>
   ```

Accessing Command-Line Interface using Telnet

To access Command-Line Interface using Telnet:

1. Run the following command to check the status of the CLI service:

   ```
   systemctl status vtap_cli.service
   ```

   If the CLI service is running, the status appears as `Active: active (running)`. If the CLI service not running, the status appears as `Active: inactive (dead)`.

   Example

   ```
   [root@8129 ~]# systemctl status vtap_cli.service
   ● vtap_cli.service - vTAP CLI
      Loaded: loaded (/etc/systemd/system/vtap_cli.service; enabled; vendor preset: disabled)
      Active: active (running) since Thu 2018-10-04 18:28:46 IST; 24h ago
     Process: 10835 ExecStop=/usr/lib/systemd/scripts/vtap_cli.sh stop (code=exited, status=0/SUCCESS)
     Process: 10855 ExecStart=/usr/lib/systemd/scripts/vtap_cli.sh start (code=exited, status=0/SUCCESS)
    Main PID: 10859 (vtap_cli)
      CGroup: /system.slice/vtap_cli.service
              └─10859 /sbin/vtap_cli

   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     registering callback = set_terminal_length
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]: return, priv, Return to previous prompt
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]: action type = cfunction, handler = return_exec
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]: testpcap, priv, start-stop sample pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     0, start, (null), starting sample pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     sub-command: action type = cfunction,
   handler = start_send_pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     registering callback = start_send_pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     0, stop, (null), stopping sample pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     sub-command: action type = cfunction,
   handler = stop_send_pcap
   Jan 31 08:58:00 localhost.localdomain vtap_cli[4111]:     registering callback = stop_send_pcap
   ```

2. Run the following command:

   ```
   telnet <vtap_ip_address>
   ```

   NOTE

   If an error is displayed (such as, `-bash: telnet: command not found`), run `yum install telnet` to first install telnet and then perform the steps above.

   Example

   ```
   [root@133131 ~]# telnet 10.37.133.131

   Connected to localhost.

   __   _| |_ __ _ _ __
   \ \ / / __/ _` | '_ \
    \ V /| || (_| | |_) |
     \_/  \__\__,_|.__/
                  |_|

   Module: vTap

   Version:2.0.0-0
   Copyright (c) 2019, by Extreme Networks

   Extreme Networks
   6480 Via Del Oro, San Jose, CA 95119

   Username:
   ```

Online help

Entering a question mark '?' at the system prompt (console prompt) displays a list of commands available for a particular command.

> **NOTE**
> Question mark '?' will not be visible on the screen.

Example

```
vtap>
  quit                Disconnect
  history             Show a list of previously run commands
  help                Show help of available commands
  enable              Turn on privileged commands
  exit                Exit from current mode
  show                Show commands
  clear               clear statistics
  status              Status of a service
```

Auto-complete

The CLI provides an auto-complete feature. To auto-complete a command, type the first few characters of a command, and press **TAB** and then press **ENTER**. For example, typing `int` and pressing **TAB** results in the CLI auto-completing the entry with the command `interface`.

If the partially-entered text matches multiple options, the CLI displays all available matching commands. For example, typing `sa` and pressing **TAB** displays `sampling` and `save`.

Command modes

The Command Line Interface (CLI) is divided into various command modes. Each command mode has its own set of specific commands.

From each mode a specific command is used to navigate from one command mode to another. The standard order to access the modes is as follows: **User EXEC mode** -> **Privileged EXEC mode** -> **Global Configuration mode**. After logging into the device, the user is automatically in Privileged EXEC command mode unless the user is defined as a User EXEC user.

> **NOTE**
> The user modes are set by default and cannot be modified.

The table below lists the main command modes, the prompts visible in each mode, and the exit method for each mode.

TABLE 1 Command Modes

Command Mode	Access Method	Prompt	Exit Method

User EXEC	This is the first level of access for changing terminal settings, performing basic tasks and listing system information.	`vtap>`	Enter one of the following commands:
			`quit`
			`logout`
			`exit`

TABLE 1 Command Modes (continued)

Command Mode	Access Method	Prompt	Exit Method
Privileged EXEC	From User EXEC mode, enter `enable` command. When prompted for the password, enter `secret`, which is the default password.	`vtap#`	Enter `disable` command to exit to User Exec mode.
Global Configuration	From Privileged EXEC mode, enter `configure terminal` command.	`vtap(config)#`	Enter `exit` command to exit to Privileged EXEC mode.

- **User EXEC mode**: The User EXEC mode can be assigned for a user after a user account is created. Only a limited set of commands are available in User EXEC mode. This level is reserved for tasks that do not change the configuration. To enter the next level, the Privileged EXEC mode, a password is required.

  In general, the User EXEC commands allow the user to perform basic tests, and list system information.
- **Privileged EXEC mode**: The Privileged EXEC mode gives access to commands that are restricted on User EXEC mode and provides access to full suite of commands. Privileged users enter directly into the Privileged EXEC mode.

  By default, privileged access is password protected to prevent unauthorized use because many of the privileged commands set operating system parameters. When prompted for the password, enter `secret`, which is the default password. The password is not displayed on the screen and is case-sensitive.

  To enter the Privileged EXEC mode from the User EXEC mode, perform the following steps:

  1. At the prompt enter the `enable` command and press **Enter**. A password prompt is displayed.
  2. Enter the password and press **Enter**. The Privileged EXEC mode prompt is displayed.

  Use the `disable` command to return from the Privileged EXEC mode to the User EXEC mode.
- **Global Configuration mode**: The Global Configuration mode manages the device configuration on a global level. Global Configuration mode commands apply to features that affect the system as a whole, rather than just a specific interface.

  To enter the Global Configuration mode, at the Privileged EXEC mode prompt enter the `configure terminal` command and press **Enter**. The Global Configuration mode prompt is displayed.

  Use the `exit` command to return from the Global Configuration mode to the Privileged EXEC mode.

Configuration modes

Configuration command-line interface (CLI) commands are entered in various modes to configure vTAP. The initial configuration mode is named global configuration mode and all other configuration modes are accessed through this mode.

To enter a specific Configuration mode, at the Global Configuration mode prompt enter the appropriate configuration command and press **Enter**. For example, at the Global Configuration mode prompt, enter `sampling` to go to Sampling Configuration mode:

```
vtap> User Level EXEC Command
vtap# Privileged Level EXEC Command
vtap(config)# Global Level CONFIG Command
vtap(config-sampling)# Sampling Level CONFIG Command
```

The following table displays a list of the most commonly-used sub-configuration modes. Refer to the command pages for details of the configuration modes applicable to the CLI command and examples of how to access the required mode.

NOTE
Command modes will vary based on the profile type selected.

TABLE 2 Command Modes

Command Mode	Access Method	Prompt	Exit Method
Egress	From Global Configuration mode, enter `egress` command.	`vtap(config)# egress` `vtap(config-egress)#`	Enter `exit`/`return` command to exit to Global Configuration mode.
Flow Exporter	From Global Configuration mode, enter `flow-exporter` command.	`vtap(config)# flow-exporter` `vtap(config-flow-exporter)#`	Enter `exit`/`return` command to exit to Global Configuration mode.
Interface	From Global Configuration mode, enter `interface` command.	`vtap(config)# interface` `vtap(config-interface)#`	Enter `exit`/`return` command to exit to Global Configuration mode.
Sampling	From Global Configuration mode, enter `sampling` command.	`vtap(config)# sampling` `vtap(config-sampling)#`	Enter `exit`/`return` command to exit to Global Configuration mode.
SMARTMatch	From Global Configuration mode, enter `smartmatch` command.	`vtap(config)# smartmatch` `vtap(config-smartmatch)#`	Enter `exit`/`return` command to exit to Global Configuration mode.

Alias	From SMARTMatch Configuration mode, enter `alias` command.	`vtap(config)# smartmatch`
		`vtap(config-smartmatch)#`
		`alias`
		`vtap(config-smatch-alias)#`
			Enter `return` command to exit to SMARTMatch Configuration mode. Enter `exit` command to exit to Global Configuration mode.

InTunnel	From Global Configuration mode, enter `intunnel` command.	`vtap(config)#`
		`smartmatchvtap(config-`
		`smatch)# intunnel`
		`vtap(config-smatch-`
		`intunnel)#`
			Enter `return` command to exit to SMARTMatch Configuration mode. Enter `exit` command to exit to Global Configuration mode.

CLI command structure

vTAP CLI commands may require input as text or number as part of the command. These fields maybe required or optional based on how the information is bracketed.

The following example shows how brackets are used in a CLI command:

add rule { **protocol**=[**any** | **ether** | **ip** | **tcp** | **udp** | **https** | **http** | **ssh** | **sctp**] | **vlan**=*ingress_vlan_id* | **host1-ip**=*host1_ip_address* | **host2-ip**=*host2_ip_address* | **host1-port**=*host1_port* | **host2-port**=*host2_port* | **smatch-alias**=[*alias_name* | *alias_name:egress_action*] | **intunnel**=*ingress_tunnel_name* | **egress**=[**drop** | *egress_action*] | **sampling-policy**=*sampling_policy_name* | **export-ipfix**=[**all** | *sampled-out*] }

An item that is:

- In italics is a variable and required.
- Not bracketed with "{ }" symbols is a required keyword or variable.
- Bracketed with "{ }" symbols with the items separated by a vertical bar "|", then one of the items must be chosen.
- Bracketed with "[ ]" symbols is optional.

Configuration commands

This chapter provides information about the configuration commands for Virtual TAP (vTAP).

Egress settings

After processing the flows/packets, vTAP sends out the packets based on the egress configuration (unless there is a sampling policy configuration that imposes restrictions).

vTAP can also be configured to send out the egress packets in GREv0 or VXLAN tunnels. A maximum of 10 egress tunnels are supported.

NOTE

- For IPv4, the parameters `local-ip` and `nexthop-mac` are optional. For IPv6, the parameter `local-ip` is optional, but `nexthop-mac` is mandatory. If the optional parameters are not configured, vTAP auto-detects the value.
- The destination port for VXLAN is set to 4789 by default and cannot be changed.
- For information about applying an egress alias on the Interface, see the section Setting and clearing interface properties on page 47.

Adding egress settings

Run the following commands to configure egress settings.

Syntax

add egress-alias=*egress_alias_name* { **local-ip**=*local_ip_address* | **destination-ip**=*destination_ip* | **nexthop-mac**=*destination_mac_address* | **vni**=*vni* | tunnel-type={ **GRE** | **VXLAN** } }

Parameters

egress-alias

Alias name for the egress configuration.

local-ip

This is the source IP address of the traffic.

NOTE

This is optional for both IPv4 and IPv6 traffic. If the IP address is not entered, it is auto-detected.

destination-ip

This is the destination IP address of the traffic.

vlan

This is the VLAN Id to be tagged with incoming packet.

NOTE

Do not specify any other parameter when configuring VLAN.

nexthop-mac

This is the destination mac address.

NOTE

For IPv4, if `nexthop-mac` is not provided, it is discovered using ARP. For IPv6, `nexthop-mac` is mandatory.

vni

Set the VNI for configuring VXLAN tunnel.

tunnel-type

Set the tunnel-type as either GRE or VxLAN.

NOTE

- For GRE tunnels: Do not configure VLAN and VNI. Local IP and destination IP are mandatory.
- For VxLAN tunnels: Do not configure VLAN. Local IP, destination IP and VNI are mandatory.

Modes

Egress Configuration mode

Examples

Example - GRE tunnel

```
vtap> enable
Password:
vtap# conf t

vtap(config)# egress

vtap(config-egress)# add egress-alias=egress-3 local-ip=2001::1 destination-ip=2001::2 nexthop-
mac=11:22:33:44:55:66 tunnel-type=gre
Egress Alias Configured successfully.
```

Example - VXLAN tunnel

```
vtap(config-egress)# add egress-alias=egress-4 local-ip=2001::1 destination-ip=2001::2 vni=10234
nexthop-mac=11:22:33:44:55:66
Egress Alias Configured successfully.
```

Example - VLAN

```
vtap(config-egress)# add egress-alias=egress-5 vlan=100
Egress Alias Configured successfully.
```

Example - Display all egress aliases

```
vtap(config-egress)# show egress-alias=all
Egress alias configuration:

            egress alias  : default-egress
                   state  : active
                 vlan-id  : 4095

            egress alias  : egress-4
                   state  : inactive
                local-ip  : 2001::1
          destination-ip  : 2001::2
        destination-port  : 4789
                     vni  : 10234
             tunnel-type  : vxlan
             nexthop-mac  : 11:22:33:44:55:66

            egress alias  : egress-5
                   state  : inactive
                 vlan-id  : 100

            egress alias  : egress-3
                   state  : inactive
                local-ip  : 2001::1
           destination-ip  : 2001::2
              tunnel-type  : gre
              nexthop-mac  : 11:22:33:44:55:66
```

Deleting egress alias

Run the following commands to delete an egress alias and all the settings associated with the alias.

Syntax

del egress-alias=*egress_alias_name*

This command deletes the egress alias.

Parameters

egress-alias

Alias name for the egress tunnel.

Modes

Egress Configuration mode

Usage Guidelines

To delete an egress tunnel that is applied on the Interface, it must first be removed from the interface.

Examples

Example for deleting an egress alias named `egress-1`.

```
vtap> enable
Password:
vtap# conf t

vtap(config)# egress

vtap(config-egress)# del egress-alias=egress-1

vtap(config-egress)#
```

After deleting the egress tunnel, run the **show egress-alias=all** command to verify:

```
vtap(config-egress)# show egress-alias=all
Egress alias configuration:
egress alias  : default-egress
      state  : active
      vlan-id  : 4095

egress-alias: egress-2
      state: active
      protocol: gre
      local-ip: 10.0.0.1
      destination-ip: 10.0.0.2
      nexthop-mac: 11:12:13:14:15:16

egress-alias: egress-3
      state: active
      protocol: gre
      local-ip: 2001::1
      destination-ip: 2001::2
      nexthop-mac: 11:22:33:44:55:66

egress-alias: egress-4
      state: active
      local-ip: 2001::1
      destination-ip: 2001::2
      dst-port : 4789
      nexthop-mac: 11:22:33:44:55:66
      vni: 10234
```

IPFIX

vTAP supports IP Flow Information Export (IPFIX), which is based on the Internet Engineering Task Force (IETF) standard. IPFIX is used to collect IP Flow information from Switches, Routers and other network devices, and analyze the Traffic Flow information. UDP is the supported transport protocol.

vTAP supports two IPFIX collectors. When two collectors are configured, the IPFIX packets are broadcast to both collectors.

Adding Flow Exporter

Run the following command to configure flow exporter.

Syntax

add name=*flow_exporter_name* **collector-ip**=*collector_ip_address***collector-port**= *collector_port* **export-ip**=*export_ip_address* **export-port**=*export_port*

Parameters

name

A string of up to 32 characters to name the collector.

collector-ip

IP address of the IPFIX collector.

collector-port

Remote logical port to be used for UDP transport. Range is 1-65535.

export-ip

Local Interface to be used for export. This IP address must be configured on a vNIC.

export-port

Local logical port to be used for UDP transport.

Modes

Flow Exporter Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t

vtap(config)# flow-exporter
vtap(config-flow-exporter)# add name=flowExp1 collector-ip=10.23.45.109 collector-port=4098 export-
ip=10.37.12.34 export-port=9000
vtap(config-flow-exporter)# add name=flowExp2 collector-ip=10.23.45.109 collector-port=5000 export-
ip=10.37.12.34 export-port=8000

vtap# show flow-exporter collector-info  collector-id=all
   collector : 1
       name       : flowExp1
    col-ip     : 10.23.45.109
    exp-ip     : 10.37.12.34
    col-port   : 4098
    exp-port   : 9000
    transport  : udp
collector : 2
       name       : flowExp2
    col-ip     : 10.23.35.109
    exp-ip     : 10.37.2.34
    col-port   : 5000
    exp-port   : 8000
       transport   : udp
```

Deleting Flow Exporter

Run the following command to delete flow exporter.

Syntax

del name=*collector_name*

Parameters

name

Delete the specified Flow Exporter.

Modes

Flow Exporter Level Configuration mode

Examples

```
vtap(config-flow-exporter)# del name=flowExp1
Flow-export collector deleted successfully.
```

Setting IPFIX export interval

This specifies the interval after which IPFIX metadata transmission to the collector is initiated. The default value is one minute.

Run the following command to configure the IPFIX export interval.

add export-interval=*export_interval_time*

Parameters

export-interval

Enter a value between 1 and 1440 (1440 minutes = 24 hours) to set the new export interval, in minutes.

Modes

Flow Exporter Configuration mode

Examples

Example for configuring export interval to `10` minutes.

```
vtap> en
Password:
vtap# conf t

vtap(config)# flow-exporter
vtap(config-flow-exporter)#


vtap(config-flow-exporter)# set
Parameter          | Type      | Default value | Help
-------------------------------------------------------------------------------------------------
export-interval | Mandatory |     None      | set export-interval=<time 1 to 1440 min>

vtap(config-flow-exporter)# set export-interval=10
Flow-export report interval updated successfully.

vtap(config-flow-exporter)# show export-interval
Time for IPFIX Report Interval : 10 min
```

Clearing IPFIX export interval

Run the following command to reset export interval to one minute, which is the default value.

Syntax

clear export-interval

Parameters

export-interval

Modes

Flow Exporter Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t

vtap(config)# flow-exporter
vtap(config-flow-exporter)# clear export-interval
Flow-export config updated successfully.

vtap(config-flow-exporter)# show export-interval
Time for IPFIX Report Interval : 1 min
```

Sampling policy

vTAP supports a maximum of 10 Sampling policy definitions.

If a SMARTMatch policy and Sampling policy are applied on an interface, the SMARTMatch policy has higher priority. A Sampling policy can be an action in a SMARTMatch rule.

For information about applying a Sampling policy on the Interface, see the section Setting and clearing interface properties on page 47.

Adding sampling policy

Run the following command to add a sampling policy.

Syntax

add sampling-policy= *sampling_policy_name*

Parameters

sampling_policy_name

Name of the sampling policy. The name is case-sensitive.

Modes

Sampling Policy Configuration mode

Usage Guidelines

A sampling policy can exist only after its properties have been set by running the **set** command.

The **sampling** command takes you to Sampling Configuration mode.

Examples

Example for adding a new sampling policy named `sp2`.

```
vtap> enable
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# add sampling-policy=sp2

vtap(config-sampling-sp2)#
```

Setting and clearing sampling policy properties

After adding a sampling policy, set its properties. The following are the commands for setting and clearing sampling policy properties.

Syntax

set sample *sampling_rate* | **preserve-pkts**

clear sample *sampling_rate* | **preserve-pkts**

Parameters

sample

This is a mandatory parameter.
Enter a value between 1 and 100 to set the sampling drop percentage. The default value is 1.

NOTE
If the sampling rate for a sampling policy is cleared (using `clear sample-rate`), the sampling rate for that policy is set to 1.

preserve-pkts

This is an optional parameter.
Enter a value between 1 and 1000.
When a session is selected to be sampled out, the `preserve-pkts` parameter specifies the number of packets to be forwarded for that session before being dropped. If this parameter is not set, the default value is set to `0` and the flow is dropped immediately.

Modes

Sampling Policy Configuration mode

Usage Guidelines

To set the properties for a sampling policy, it must first be added using the **add sampling-policy** command.

If you have just added the sampling policy, you will already be at sampling policy level to set its properties. If you are not at sampling policy level, at the Sampling Policy Configuration mode, run the **edit sampling-policy=<sampling_policy_name>** command.

After setting sampling policy properties, run the **return/exit** command to go one level back from sampling policy level to Sampling Policy Configuration mode.

After clearing the sampling rate (using `clear sample` command), do not exit the command mode without setting a new sampling rate.

If you clear all the properties of a sampling policy and then run the **return** command, the sampling policy is deleted.

A sampling policy cannot be deleted if it is already applied on the interface or in a SMARTMatch rule. To delete a sampling policy, remove it from the interface and all the SMARTMatch rules and delete the policy.

Examples

Example - Setting properties

Example for setting properties for the sampling policy `sp1`. In this example, sample drop percentage is 50 and Preserve Packets is set to 100.

> NOTE
> When a new Sampling policy is added, its **State** is `inactive` (as shown below). The **State** changes to `active` after the policy is added to an interface.

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# add sampling-policy=sp2

vtap(config-sampling-sp2)# set sample-rate=10
sampling policy "sp2" parameter "sampling " configured successfully.

vtap(config-sampling-sp2)# set preserve-pkts=10
sampling policy "sp2" parameter "preserve-pkts" configured successfully.
```

Run the **show sampling-policy=sp1** command to verify:

```
vtap(config-sampling-sp2)# show sampling-policy=all
policy name : sp2
           state : inactive
     sample-rate : 10
   preserve-pkts : 10

policy name : sp1
           state : inactive
     sample-rate : 10
   preserve-pkts : 10
```

Example - Clearing properties

Example for clearing properties for the sampling policy `sp1`. In this example, Preserve Packets is cleared.

```
vtap(config-sampling-sp1)# clear preserve-pkts
```

Run the **show sampling-policy=sp1** command to verify:

```
vtap(config-sampling-sp2)# show sampling-policy=all
policy name : sp2
           state : inactive
     sample-rate : 10
   preserve-pkts : 10

policy name : sp1
           state : inactive
     sample-rate : 10
   preserve-pkts : 0
```

Editing and deleting sampling policy

Run the following commands to edit or delete a sampling policy.

Syntax

edit sampling-policy= *sampling_policy_name*

This command takes you to the sampling policy level, where you can edit the properties of a sampling policy.

del sampling-policy= *sampling_policy_name*

This command deletes the sampling policy.

Parameters

policy_name

Name of the sampling policy to edit or delete.

Modes

Sampling Policy Configuration mode

Usage Guidelines

The **edit sampling-policy** command takes you to the sampling policy level.

The following restrictions apply for managing a sampling policy:

- A Sampling policy cannot be deleted if it is already applied on the Interface or it is in use within a SMARTMatch rule. To delete a Sampling policy, it must be removed from the Interface and all the SMARTMatch rules it is used in.
- A Sampling policy cannot be renamed. It has to be deleted and a new one created.

Examples

Example - Edit

Example for editing a sampling policy named `sp1`.

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# edit sampling-policy=sp1

vtap(config-sampling-sp1)#
```

Example - Delete

Example for deleting sampling policy named `sp1`.

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# del sampling-policy=sp1
sampling policy sp1 deleted successfully.
```

Run the **show sampling-policy=all** command to verify:

```
vtap(config-sampling)# show sampling-policy=all
No sampling policy configured
```

SMARTMatch policy

A SMARTMatch policy is useful for collating multiple generic packet match rules, each of which specify the following:

- Filtering rules based on n-tuple parameters.
- Configure up to four flex-match aliases (optional).
- Configure a set of actions - forward to an egress destination or drop, Sampling Flows, and Export IPFIX metadata (optional).

Only one SMARTMatch policy can be configured in vTAP.

A SMARTMatch rule enables vTAP to identify tunnels, flows, or packets based on n-tuples, and configure actions for tunnel/flow/packets matching the n-tuples.

A SMARTMatch rule also supports flex-match capability to detect one or more regex/hex patterns anywhere within the L2 packet boundary and configure action to either drop the traffic or forward the traffic to an egress path. If no action is configured for a flex-match, the respective counters are incremented for the detected pattern . If no forwarding or drop action is configured for the flex-match, the SMARTMatch rule actions are effective.

A SMARTMatch rule must include at least one Match criteria.

For information about applying a SMARTMatch policy to an interface, see the section Setting and clearing interface properties on page 47.

Adding alias

An alias must include at least one flex-match. Multiple flex-matches can be configured .

vTAP supports regex and hex format for search pattern. The maximum search pattern length is 512 characters.

Run the following command to add alias.

Syntax

add alias = *alias_name* **flex-match** = *offset:length:hex pattern*

Parameters

alias

Name of the alias. The name is case-sensitive.

flex-match

This parameter indicates the pattern (regex or hex) at a specific offset. The maximum search pattern length is 512 characters.

NOTE

All strings must be added between two backtick symbols (`).

Example

- Adding string **abc*.com**:

```
add alias=a1 flex-match=0:0:`abc*.com`
```

- Adding string **extremenet**:

```
add alias=a1 flex-match=0:0:`extremenet`
```

- Adding **?** for string matching:

```
add alias=a1 flex-match=0:0:`(?i)pattern`
```

Modes

Alias Level Configuration mode

Usage Guidelines

The **alias** command takes you from SMARTMatch Configuration mode to Alias Configuration mode.

Examples

Example 1

Example for adding an alias named `a1`.

The flex-match **10:10:pattern1** indicates that the start offset is `10` and the end offset is `20` (start offset + length = 10 + 10 = 20) for the search pattern `pattern1`. The offset values are calculated based on the n-tuple match that uses it. Length value of `0` indicates end of packet as the end offset for the pattern match.

```
vtap> enable
Password:
vtap# conf t

vtap(config)# smartmatch
vtap(config-smartmatch)# alias

vtap(config-smatch-alias)# add alias=a1 flex-
match=10:10:pattern1,20:20:pattern2,30:30:pattern3,40:40:pattern4
Match criteria updated successfully.
```

Example 2

```
vtap> enable
Password:
vtap# conf t

vtap(config)# smartmatch
vtap(config-smartmatch)# alias

vtap(config-smatch-alias)# add alias=test_case_1 flex-match=10:100:cloud
Match criteria updated successfully.
```

Run the **show alias=all** command to display the aliases:

```
vtap(config-smatch-alias)# show alias=all

 Flex match combination 1
        alias name                    : a1
        Number of flex matches        : 4
          Flex match1 byte offset     : 10
          Flex match1 length to search: 10
          Flex match1 pattern         : pattern1
          Flex match2 byte offset     : 20
          Flex match2 length to search: 20
          Flex match2 pattern         : pattern2
          Flex match3 byte offset     : 30
          Flex match3 length to search: 30
          Flex match3 pattern         : pattern3
          Flex match4 byte offset     : 40
          Flex match4 length to search: 40
          Flex match4 pattern         : pattern4
 Flex match combination 2
        alias name                    : test_case_1
        Number of flex matches        : 1
          Flex match1 byte offset     : 10
          Flex match1 length to search: 100
          Flex match1 pattern         : cloud
```

Deleting alias

Run the following command to delete an alias.

Syntax

del alias =*alias_name*

Parameters

alias

Delete the specified alias.

Modes

Alias Level Configuration mode

Usage Guidelines

The **alias** command takes you from SMARTMatch Configuration mode to Alias Configuration mode.

A n alias cannot be deleted if it is in use in a SMARTMatch rule. In such cases, first delete the rule and then delete the alias.

Examples

Example

```
vtap(config-smartmatch-smp1)# del alias=a1
Alias deleted successfully
```

Ingress tunnel

A SMARTMatch rule can be configured for a specific tunnel on the network. A maximum of 100 ingress tunnels can be configured.

An ingress tunnel can be VLAN, Q-in-Q, GRE, ERSPAN Type II, IPIP, or VXLAN.

NOTE

Nested tunnels are not supported. Therefore, GRE or IPIP or VXLAN cannot be configured along with VLAN or Q-in-Q.

Adding ingress tunnel

Run the following command to add an ingress tunnel.

Syntax

add tunnel-name=*tunnel-name* **svlan**=*svlan-id* [**cvlan** =*cvlan-id*]

add tunnel-name=*tunnel-name* **host1-ip**=*host1-ip-address* **host2-ip**=*host2-ip-address* [**erspanII-vlan**=*vlan_id*] **tunnel-type**= [**GRE** | **IPIP**] | [**vni**=*vxlan-vni*]

Parameters

tunnel-name

Name of the ingress tunnel. The name is case-sensitive.

svlan

Specifies the Service VLAN (svlan) of the packet.

NOTE

`svlan` must be specified for VLAN.

cvlan

Specifies the Customer VLAN (cvlan) of the packet.

NOTE

Both `cvlan` and `svlan` must be specified for Q-in-Q.

host1-ip

Specifies the IP address of one host in the IP header.

host2-ip

Specifies the IP address of the other host in the IP header.

tunnel-type

Specifies the tunnel as GRE, VXLAN or IPIP.

NOTE

For ERSPAN Type II, specify `erspanII-vlan` with `tunnel-type` as `gre`.

erspanII-vlan

Specifies VLAN Id for ERSPAN Type II tunnel.

NOTE

For ERSPAN Type II, tunnel-type as GRE and ERSPAN VLAN is mandatory, while ERSPAN Session ID is optional.

vni

Specifies the VNI for configuring VXLAN tunnel.

NOTE

Specifying the VNI value implies the tunnel type as VXLAN. The destination port for VXLAN is 4789. This value is fixed and cannot be changed.

Modes

Ingress Tunnel Configuration mode

Usage Guidelines

The **intunnel** command takes you from SMARTMatch Configuration mode to Ingress Tunnel Configuration mode.

Examples

Example - VLAN

Example for adding an ingress tunnel named `int1`.

```
vtap> en
Password:
vtap# conf t

vtap(config)# smartmatch
vtap(config-smartmatch)# intunnel
vtap(config-smartmatch-intunnel)#add tunnel-name=t1 svlan=100
Ingress tunnel configured successfully.
```

Example - QinQ

```
vtap(config-smartmatch-intunnel)#add tunnel-name=t2 svlan=120 cvlan=1055
Ingress tunnel configured successfully.
```

Example - GRE tunnel

```
vtap(config-smartmatch-intunnel)# add tunnel-name=t3 host1-ip=2001::1 host2-ip=2002::2 tunnel-type=gre
Ingress tunnel configured successfully.
```

Example - IPIP tunnel

```
vtap(config-smartmatch-intunnel)# add tunnel-name=t4 host1-ip=2001::1 host2-ip=2002::2 tunnel-type=ipip
Ingress tunnel configured successfully.
```

Example - VxLAN tunnel

```
vtap(config-smartmatch-intunnel) add tunnel-name=t5 host1-ip=10.0.0.1 host2-ip=10.0.0.2 vni=100
Ingress tunnel configured successfully.
```

Example - ERSPAN Type II tunnel

```
vtap(config-smartmatch-intunnel)# add tunnel-name=t6 host1-ip=2001::1 host2-ip=2002::2 erspanII-
vlan=100 tunnel-type=gre
Ingress tunnel configured successfully.
```

Example - Display all ingress tunnels

```
vtap(config-smartmatch-intunnel)# show tunnel-name=all

             tunnel-name  : t5
                host1-ip  : 10.0.0.1
                host2-ip  : 10.0.0.2
        destination-port  : 4789
             tunnel-type  : vxlan
                     vni  : 100

             tunnel-name  : t2
                   svlan  : 120
                   cvlan  : 1055

             tunnel-name  : t4
                host1-ip  : 2001::1
                host2-ip  : 2002::2
             tunnel-type  : ipip

             tunnel-name  : t1
                    vlan  : 100

             tunnel-name  : t6
                host1-ip  : 2001::1
                host2-ip  : 2002::2
           erspanII-vlan  : 100
             tunnel-type  : gre-erspanII

             tunnel-name  : t3
                host1-ip  : 2001::1
                host2-ip  : 2002::2
             tunnel-type  : gre
```

Deleting ingress tunnel

Run the following command to delete ingress tunnel.

Syntax

del tunnel-name =*tunnel_name*

Parameters

tunnel-name

Delete the specified ingress tunnel.

Modes

Ingress Tunnel Level Configuration mode

Usage Guidelines

The **intunnel** command takes you from SMARTMatch Configuration mode to Ingress Tunnel level Configuration mode.

An ingress tunnel cannot be deleted if it is in use in a SMARTMatch rule. In such cases, first delete the rule and then delete the tunnel.

Examples

```
vtap> en
Password:
vtap# conf t

vtap(config)# smartmatch
vtap(config-smartmatch)# intunnel
vtap(config-smartmatch-intunnel)# del tunnel-name=int1
Tunnel deleted successfully
```

Adding SMARTMatch policy

Run the following command to add a SMARTMatch policy.

Syntax

add smartmatch-policy= *SMARTMatch_policy_name*

Parameters

SMARTMatch_policy_name

Name of the SMARTMatch policy you want to add.

Modes

SMARTMatch Policy Configuration mode

Usage Guidelines

A SMARTMatch policy can exist only if at least one rule is added to the policy.

The **add smartmatch-policy** command takes you to SMARTMatch policy level.

Examples

Example for adding a new SMARTMatch policy named `smp1`.

```
vtap> enable
Password:
vtap# conf t
vtap(config)# smartmatch

vtap(config-smartmatch)# add smatch-policy=smp1

vtap(config-smartmatch-smp1)#
```

Editing and deleting SMARTMatch policy

Run the following commands to edit or delete SMARTMatch policy.

Syntax

edit smartmatch-policy=*SMARTMatch_policy_name*

Use this command to edit the properties of a SMARTMatch policy.

del smartmatch-policy=*SMARTMatch_policy_name*

Use this command to delete a SMARTMatch policy.

Parameters

SMARTMatch_policy_name

Name of the SMARTMatch policy you want to edit or delete.

Modes

SMARTMatch Policy Configuration mode

Usage Guidelines

The **edit smartmatch-policy** command takes you to the SMARTMatch policy level.

The following restrictions apply for managing a SMARTMatch policy:

- A SMARTMatch policy cannot be deleted if it is already applied on an interface. To delete a SMARTMatch policy, it must first be removed from the interface.
- When a SMARTMatch policy is removed from the interface, all the rules associated with the policy are also removed.
- A SMARTMatch policy cannot be renamed. It must be deleted and new one created.

Examples

Example 1 - Edit SMARTMatch policy

Example for editing a SMARTMatch policy named `smp1`.

```
vtap(config-smartmatch)# edit smatch-policy=smp1
vtap(config-smartmatch-smp1)#
```

Example 2 - Delete SMARTMatch policy

Example for deleting SMARTMatch policy named `smp1`.

```
vtap(config-smartmatch)# del smatch-policy=smp1
SMARTMatch policy smp1 deleted successfully.
```

After deleting the SMARTMAtch policy, run the **show smartmatch-policy=all** command to verify:

```
vtap(config-smartmatch)# show smatch-policy=all
No Policy created.
```

Adding rules

Run the following commands to add rules. A rule must include at least one of the parameters listed below.

Syntax

add rule { **protocol**=[**any** | **ether** | **ip** | **tcp** | **udp** | **https** | **http** | **ssh** | **sctp**] | **vlan**=*ingress_vlan_id* | **host1-ip**=*host1_ip_address* | **host2-ip**=*host2_ip_address* | **host1-port**=*host1_port* | **host2-port**=*host2_port* | **smatch-alias**=[*alias_name* | *alias_name:egress_action*] | **tunnel-name**=*ingress_tunnel_name* | **egress**=[**drop** | *egress_action*] | **sampling-policy**=*sampling_policy_name* | **export-ipfix**=[**all** | *sampled-out*] | **header-strip**=[**vxlan** | **nvgre** | **mpls** | **802.1br_vntag** | **erspanII** | **gtpu**] **packet-slice**= *offset_value*

Parameters

The following parameters are optional. However at least one parameter is required for adding a SMARTMatch rule:

protocol

This is the protocol of the IP traffic. It may also specify the protocol within a particular type of payload. The supported values are `any`, `ether`, `ip`, `tcp`, `udp`, `https`, `http`, `ssh`, `sctp`.

vlan

This is the VLAN of the packet irrespective of the traffic type.

host1-ip

This is the source IP address of the IP traffic.

host2-ip

This is the destination IP address of the IP traffic.

host1-port

This is the source port of the IP traffic.

host2-port

This is the destination port of the IP traffic.

smatch-alias

This is the name of the SMARTMatch alias.

NOTE

For information about adding a SMARTMatch alias, see the section Adding alias on page 31.

tunnel-name

This is the name of the ingress tunnel.

NOTE

For information about adding an ingress tunnel, see the section Adding ingress tunnel on page 35.

egress

This is the egress path on which a flow/packet is to be sent out after processing. The egress may be set to `drop` to drop packets post-processing.

sampling-policy

This is the name of the sampling policy.

export-ipfix

Set the option to export IPFIX for all:

- Flows/packets sent to egress

  or

- Dropped packets post-processing
  This parameter is applicable only on 'session'. It enables IPFIX metadata generation for flow sessions on which the SMARTMatch policy is not applied. When this parameter is set to `sampled-out`, the metadata is exported for the sampled-out flow sessions for which packets are being dropped by vTAP.

header-strip

Set the header to be stripped.
One of the following values can be specified:

- `802.1br_vntag`
- `vxlan`
- `nvgre`
- `mpls`
- `erspan`
- `gtpu`

packet-slice

Set the offset value for packet slicing. A value between 1 and 1000 can be specified.

Modes

SMARTMatch Policy Configuration mode

Usage Guidelines

The following restrictions apply for managing a SMARTMatch rule:

- A SMARTMatch rule cannot be modified. It has to be deleted and a new one added.
- A SMARTMatch rule is applied on a Flow if its source/destination IP address/ports match the rule parameters. The rules are applied for both directions of a Flow.
- If a SMARTMatch rule specifies an Ingress Tunnel name (`tunnel-name`) of type VLAN or Q-in-Q, the flow tuples cannot contain VLAN with an absolute value.

  For example, `add rule tunnel-name=t1 vlan=10` is not allowed if `t1` is Q-in-Q or VLAN. However, the VLAN value above can be `any`.
- SMARTMatch rules do not have any priority. The rules are selected based on Longest Prefix Match criteria. Addition of a new rule with longer prefix match results existing rules being overridden. However, this does not result in re-evaluation of existing rules already applied across the system.

Examples

```
vtap(config-smartmatch)# add smatch-policy=smatch-1
vtap(config-smartmatch-smp1)# add rule vlan=400 protocol=tcp host1-ip=10.0.0.1 host2-ip=10.0.0.2 smatch-
alias=alias1:egress-1,alias2:egress-2,alias3:drop,alias4:drop sampling-policy=sampling-1 export-
ipfix=sampled-out header-strip=vxlan,nvgre packet-slice=72
Rule with Rule ID 1 configured successfully for policy smp1

vtap(config-smartmatch-smp1)#add rule vlan=any host1-ip=10.0.0.1 host2-ip=100.1.1.1 egress=drop export-
ipfix=all
Rule with Rule ID 2 configured successfully for policy smp1

vtap(config-smartmatch-smp1)#add rule vlan=any host1-ip=2001:43:12::1 host2-ip=2600:32:11::1
egress=egress-2 export-ipfix=disable
Rule with Rule ID 3 configured successfully for policy smp1

vtap(config-smartmatch-smp1)# show smatch-policy=all
smart-match policy 1
    name : smp1
   state : inactive
   rules :
         rule id : 1
         vlan id : any
        protocol : any
        host1 ip : 10.0.0.1
        host2 ip : 100.1.1.1
      host1 port : any
      host2 port : any
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : drop
    header-strip : -
    packet-slice : -

         rule id : 2
         vlan id : 400
        protocol : tcp
        host1 ip : 10.0.0.1
        host2 ip : 10.0.0.2
      host1 port : any
      host2 port : any
     no of alias : 1
            alias name 1 : a1
 sampling policy : sp1
     tunnel name : --
    export ipfix : sampled-out
          egress : --
    header-strip : vxlan,nvgre
    packet-slice : 72

         rule id : 3
         vlan id : any
        protocol : any
        host1 ip : 2001:43:12::1
        host2 ip : 2600:32:11::1
      host1 port : any
      host2 port : any
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : disable
          egress : egress-1
    header-strip : -
    packet-slice : -
```

Deleting rules

Run the following command to delete rules.

Syntax

del { **rule rule-id**=*rule_id* | **all** }

Parameters

rule-id

Delete specific rules.

all

Delete all rules.

Modes

SMARTMatch Policy Configuration mode

Usage Guidelines

The **edit smartmatch-policy** command takes you to the SMARTMatch policy level.

The following restrictions apply on managing a SMARTMatch policy when vTAP is in service:

- A SMARTMatch rule cannot be modified. It has to be deleted and a new one added.
- SMARTMatch rule deletion takes effect immediately.

Examples

Example 1

```
vtap(config-smartmatch-smp1)# del rule rule-id=1
Rules deleted successfully
```

Interface

An Interface is a logical entity that represents a collection of ingress ports as well as a traffic type. vTAP includes a single Interface with the traffic type factory-set to IP. An Interface is always of type Ingress.

The interface logically maps to the Rx vNIC.

Editing interface

Run the following commands to edit the IP interface.

Syntax

edit interface *interface_name*

Use this command to edit the properties of an interface.

Parameters

interface_name

Name of the interface.

Modes

Interface Configuration mode

Usage Guidelines

The **edit interface** command takes you to the Interface level.

Examples

```
vtap> en
Password:
vtap# conf t

vtap(config)# interface
vtap(config-interface)# show interface=all

                     name  : ip
             traffic type  : ip
                   egress  : eAlias1
       vtap ingress ports  : ens34,ens35
                     mode  : session
             export IPFIX  : all
             header-strip  : -
             packet-slice  : -

vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)#
```

Setting and clearing interface properties

The following are the commands for setting and editing interface properties.

Syntax

set { **sampling-policy**=*sampling_policy_name* | **smatch-policy**= *smatch_policy_name* | **mode**= {*session* | *packet* } |**egress**= *egress_alias_name* | **export-ipfix**= {**all** | *sampled-out* } | **header-strip** = {*vxlan* | *nvgre* | *mpls* | *802.1br_vntag* } | **packet-slice**=*offset*}

clear { **sampling-policy** | **smatch-policy** | **mode** | **egress**| **export-ipfix**|**header-strip** | **packet-slice**}

Parameters

sampling-policy

Specifies the sampling policy to be applied on the interface. The sampling policy is applied only on session.

NOTE

For information about adding a Sampling policy, see the section Adding sampling policy on page 26.

smatch-policy

Specifies the SMARTMatch policy to be applied on the interface. The SMARTMatch policy is applied on both session and packet.

NOTE

For information about adding a SMARTMatch policy, see the section Adding SMARTMatch policy on page 39.

mode

Specifies whether the flow session management needs to be enabled for the interface by default. It can be set to either `session` or `packet`. The default mode is session.

When the mode is set to `session`, the flow session management is enabled for any packet received with the supported protocols. If the mode is set to `packet`, all the actions are applied on a per-packet basis, and flow session management is not enabled by default.

If a SMARTMatch rule is configured with Sampling or IPFIX, and the mode is set to `packet`, then flow sessions are maintained for the traffic on which such a rule is applied. If the mode is dynamically modified from `session` to `packet`:

- `sampling-policy` and `export-ipfix` properties of the interface are cleared dynamically.
- All flow sessions are cleared except for the ones on which SMARTMatch rules with Sampling or IPFIX actions are applied.

NOTE

Run the `clear mode` command to set mode to its default setting (`session`).

egress

Specifies the egress path on which a flow/packet is to be sent out after processing. Egress is set to drop by default, to drop the packets after processing.

NOTE

- Run the `clear egress` command to set egress to its default setting (`drop`).

- For information about adding an egress alias and to set its properties, see the section Adding egress settings on page 18.

export-ipfix

Set the option to either export IPFIX for all flows/packets that are sent to egress or to drop the flow/packets after processing.

This parameter is applied only on session. It enables IPFIX metadata generation for flow sessions on which the SMARTMatch policy is not applied. The value `sampled-out` specifies that the metadata must be exported for the sampled-out flow sessions for which packets are dropped by vTAP.

NOTE

IPFIX export can be set on the interface even if a collector is not configured. If a collector is not configured, IPFIX export does not occur.

header-strip

Set the header to be stripped.

One of the following values can be specified:

- `802.1br_vntag`
- `vxlan`
- `nvgre`
- `mpls`
- `erspan`
- `gtpu`

packet-slice

Set the offset value for packet slicing. A value between 1 and 1000 can be specified.

Modes

Interface Configuration mode

Usage Guidelines

To get to the Interface level, at the Interface Configuration mode, run the **edit interface** command.

A Sampling/SMARTMatch policy cannot be deleted if it is already applied on the Interface. To delete a policy, it must first be removed from the interface.

When a SMARTMatch policy is removed from an interface, the rules associated with that SMARTMatch policy are also removed immediately.

When a SMARTMatch policy and a Sampling policy is applied on the interface, the SMARTMatch policy takes precedence over the Sampling policy.

Examples

Example 1 - Set properties

```
vtap> en
Password:
vtap# conf t

vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# set sampling-policy=sp1
Interface property updated successfully.

vtap(config-interface-ip)# set smatch-policy=smp1
Interface property updated successfully.
```

After setting a interface's properties, run the **show interface=all** command to verify:

```
vtap(config-interface)# show interface=all

                     name  : ip
             traffic type  : ip
                   egress  : default-egress
    sampling policy-name   : sp1
  smartmatch policy-name   : smp1
                     mode  : session
             export-ipfix  : disabled
             header-strip  : -
             packet-slice  : -
```

Example 2 - Clear properties

Example for clearing properties for the interface.

```
vtap> enable
Password:
vtap# conf t

vtap(config)# interface

vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear sampling-policy
Interface "ip"  parameters "Sampling policy sp1" cleared successfully.
```

After clearing an interface's properties, run the **show interface=all** command to verify:

```
vtap(config-interface)# show interface=all

                     name  : ip
             traffic type  : ip
                   egress  : default-egress
  smartmatch policy-name   : smp1
```

```
                mode  : session
        export-ipfix  : disabled
        header-strip  : -
        packet-slice  : -
```

Other configuration commands

load config

Use this command to load the saved configuration in vTAP.

NOTE

- If the saved configuration has 3k rules, vTAP takes up to 8 minutes to load all the rules in the file.
- When a saved configuration is loaded, all existing policies and rules are deleted.

Syntax

load config

Modes

Global Configuration mode

Examples

```
vtap> enable
Password:
vtap# conf t

vtap(config)# load config
Loading new configuration will clear all existing configuration
Do you want to continue with load configuration (yes/y/no/n)?y


Clearing egress alias

Clearing intunnel

Clearing flexmatch alias


Clearing flow-exporter config

Clearing log modules
Logging cleared successfully for tunnel-flow module
Logging cleared successfully for smatch-sample module
Logging cleared successfully for ipfix module
Logging cleared successfully for mgmt module
Logging cleared successfully for L7app module
Logging cleared successfully for kni module
```

save config

Use this command to save the current running configuration to the startup configuration file. The file, named `saved_config` by default, can be saved to a user-defined location. If no location is provided, it is saved to the default location, `/etc/vtap/config/`.

Syntax

save config

Modes

Global Configuration mode

Examples

```
vtap> enable
Password:
vtap# conf t

vtap(config)# save config
This command may take maximum time of 30 seconds
Configuration file saved successfully
```

set logging

Use this command to set Warning, Info, and Debug log levels for various modules.

NOTE

Critical and Error log levels are always on and enabled by default.

set logging={ **tunnel-flow** | **smatch** | **sampling** |**ipfix**| **mgmt** | **kni** | **all** } **level**={**info**| **warning** | **debug** }

Modes

Global Configuration mode

Examples

```
vtap> enable
Password:
vtap# conf t

vtap(config)# set logging
enter module=<tunnel-flow|smatch-sample|ipfix|mgmt|L7app|kni|all>

vtap(config)# set logging module=all
enter level=<info|warning|debug>

vtap(config)# set logging module=all level=info
info Log level enabled for module all successfully

vtap(config)# set logging module=tunnel-flow level=warning
warning Log level enabled for module tunnel-flow successfully

vtap(config)# set logging module=smatch-sample level=all
debug Log level enabled for module smatch-sample successfully
```

terminal length

Use this command to set the number of lines of output to be displayed on the terminal screen before pausing. Enter `0` to disable paging.

Syntax

terminal length

Modes

Privileged EXEC mode

Examples

```
vtap> enable
Password:

vtap# terminal length 10
Terminal length updated
```

Show commands

This chapter provides information about the vTAP `show` commands. All commands are listed in alphabetical order.

show alias

Use this command to display all the configured aliases.

Syntax

show alias = { **all** | *alias_name* }

Modes

SmartMatch Configuration mode

Examples

Example

```
vtap> en
Password:
vtap# conf t

vtap(config)# smartmatch

vtap(config-smartmatch)# alias

vtap(config-smartmatch-alias)# show alias=all

 Flex match combination 1
        alias name                    : a1
        Number of flex matches        : 4
          Flex match1 byte offset     : 10
          Flex match1 length to search: 10
          Flex match1 pattern         : pattern1
          Flex match2 byte offset     : 20
          Flex match2 length to search: 20
          Flex match2 pattern         : pattern2
          Flex match3 byte offset     : 30
          Flex match3 length to search: 30
          Flex match3 pattern         : pattern3
          Flex match4 byte offset     : 40
          Flex match4 length to search: 40
          Flex match4 pattern         : pattern4
 Flex match combination 2
        alias name                    : test_case_1
        Number of flex matches        : 1
          Flex match1 byte offset     : 10
          Flex match1 length to search: 100
          Flex match1 pattern         : cloud
```

Example 2

```
vtap(config-smartmatch-alias)# show alias=a1

        alias name                    : a1
        Number of flex matches        : 4
          Flex match1 byte offset     : 10
          Flex match1 length to search: 10
          Flex match1 pattern         : pattern1
          Flex match2 byte offset     : 20
          Flex match2 length to search: 20
          Flex match2 pattern         : pattern2
          Flex match3 byte offset     : 30
          Flex match3 length to search: 30
          Flex match3 pattern         : pattern3
          Flex match4 byte offset     : 40
          Flex match4 length to search: 40
          Flex match4 pattern         : pattern4
```

show collector

Use this command to display all the configured flow exporters.

Syntax

show collector = { **all** | *collector_name* }

Modes

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# show collector=all
Collector ID         : 1
name                 : test
Collector Address    : 10.9.9.186
Export Address       : 10.9.9.184
Transport Protocol   : UDP
Collector Port       : 8888
Export Port          : 9999
```

show config

Use this command to display all vTAP-related configuration information in the running configuration.

Syntax

show config { running | all | alias | sampling-policy | smatch-policy | interface | egress-alias | ingress-tunnel | collector | template | export-interval }

Modes

User EXEC mode

Privileged EXEC mode

Global Configuration mode

Examples

```
vtap> show config running

_____________________________________________________________________________


_____________________________________________________________________________


Interface information

                    name  : ip
            traffic type  : ip
                  egress  : drop
    vtap ingress ports  : ens224,ens256
                    mode  : session
            export IPFIX  : disabled


_____________________________________________________________________________


        IPFIX Collector Info
No element configured

_____________________________________________________________________________


        IPFIX Template Info
Template ID     : 300
        Set ID                          : 3
        Number of Scope Fields          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 0
Template ID     : 257
        Set ID                          : 2
        Number of Fields                : 24
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 256
        Set ID                          : 2
        Number of Fields                : 15
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 258
        Set ID                          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 272
        Set ID                          : 2
        Number of Fields                : 4
        Number of Dependent Templates   : 0
Template ID     : 271
        Set ID                          : 2
        Number of Fields                : 5
        Number of Dependent Templates   : 0
Template ID     : 273
        Set ID                          : 2
        Number of Fields                : 11
        Number of Dependent Templates   : 0

_____________________________________________________________________________


        IPFIX Export Interval
Time for IPFIX Report Interval : 1 min

_____________________________________________________________________________
```

```
Ingress port information
-------------------------------------------------------
| Type            | Name            | Jumbo Frame     |
-------------------------------------------------------
| IP-PORT-RX      | ens224          | enabled         |
| IP-PORT-TX      | ens256          | enabled         |
-------------------------------------------------------
```

```
vtap> show config all
```

```
Sampling information

policy name : sp1
          state : active
     sample-rate : 10
   preserve-pkts : 0
```

```
Smartmatch alias information
Flex match combination 1
        alias name                    : alias1
        Number of flex matches        : 3
          Flex match1 byte offset     : 32
          Flex match1 length to search: 10
          Flex match1 pattern         : cloud
          Flex match2 byte offset     : 0
          Flex match2 length to search: 0
          Flex match2 pattern         : 0x3456
          Flex match3 byte offset     : 0
          Flex match3 length to search: 0
          Flex match3 pattern         : c1.*.d
```

```
Smartmatch information
smart-match policy 1
    name : smp1
   state : inactive
   rules :
         rule id : 1
         vlan id : 100
        protocol : any
          src ip : any
          dst ip : any
        src port : any
        dst port : any
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : disable
          egress : --

         rule id : 2
         vlan id : 100
        protocol : http
          src ip : any
          dst ip : any
        src port : 80/any
        dst port : any/80
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : --

         rule id : 3
```

```
         vlan id : 100
        protocol : https
          src ip : any
          dst ip : any
        src port : 443/any
        dst port : any/443
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : --

         rule id : 4
         vlan id : 100
        protocol : https
          src ip : any
          dst ip : any
        src port : 45
        dst port : 443
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : --

         rule id : 5
         vlan id : any
        protocol : tcp
          src ip : 10.0.0.1
          dst ip : 10.0.0.2
        src port : any
        dst port : any
     no of alias : 0
 sampling policy : sp1
     tunnel name : t1
    export ipfix : sampled-out
          egress : --

         rule id : 6
         vlan id : any
        protocol : any
          src ip : 10.0.0.1
          dst ip : 100.1.1.1
        src port : any
        dst port : any
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : drop



_____________________________________________________________________________

Interface information


                    name  : ip
            traffic type  : ip
                  egress  : drop
   vtap ingress ports  : ens224,ens256
                    mode  : session
            export IPFIX  : disabled


_____________________________________________________________________________

Ingress port information
-------------------------------------------------------
| Type            | Name            | Jumbo Frame    |
-------------------------------------------------------
```

```
| IP-PORT-RX      | ens224          | enabled        |
| IP-PORT-TX      | ens256          | enabled        |
-----------------------------------------------------

_____________________________________________________________________________


Egress alias configuration:
No egress alias configured

_____________________________________________________________________________


        IPFIX Collector Info
No element configured

_____________________________________________________________________________


        IPFIX Template Info
Template ID     : 300
        Set ID                          : 3
        Number of Scope Fields          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 0
Template ID     : 257
        Set ID                          : 2
        Number of Fields                : 24
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 256
        Set ID                          : 2
        Number of Fields                : 15
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 258
        Set ID                          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 272
        Set ID                          : 2
        Number of Fields                : 4
        Number of Dependent Templates   : 0
Template ID     : 271
        Set ID                          : 2
        Number of Fields                : 5
        Number of Dependent Templates   : 0
Template ID     : 273
        Set ID                          : 2
        Number of Fields                : 11
        Number of Dependent Templates   : 0

_____________________________________________________________________________


        IPFIX Export Interval
Time for IPFIX Report Interval : 1 min

_____________________________________________________________________________


Ingress tunnel

              tunnel-name  : t1
                     vlan  : 100
_____________________________________________________________________________



_____________________________________________________________________________
```

```
Logging information
         module : level
    tunnel-flow : critical
         smatch : critical
       sampling : critical
          ipfix : critical
           mgmt : critical
            kni : critical
```

```
vtap> show config alias=all

Flex match combination 1
        alias name                   : alias1
        Number of flex matches       : 3
          Flex match1 byte offset    : 32
          Flex match1 length to search: 10
          Flex match1 pattern        : cloud
          Flex match2 byte offset    : 0
          Flex match2 length to search: 0
          Flex match2 pattern        : 0x3456
          Flex match3 byte offset    : 0
          Flex match3 length to search: 0
          Flex match3 pattern        : c1.*.d

vtap> show config sampling-policy=all
policy name : sp1
           state : active
     sample-rate : 10
   preserve-pkts : 0

vtap> show config smatch-policy=all
smart-match policy 1
    name : smp1
   state : inactive
   rules :
         rule id : 1
         vlan id : 100
        protocol : any
          src ip : any
          dst ip : any
        src port : any
        dst port : any
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : disable
          egress : --

         rule id : 2
         vlan id : 100
        protocol : http
          src ip : any
          dst ip : any
        src port : 80/any
        dst port : any/80
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : --

         rule id : 3
         vlan id : 100
        protocol : https
          src ip : any
          dst ip : any
        src port : 443/any
        dst port : any/443
     no of alias : 0
```

```
 sampling policy : --
     tunnel name : --
    export ipfix : all
           egress : --

          rule id : 4
          vlan id : 100
         protocol : https
           src ip : any
           dst ip : any
         src port : 45
         dst port : 443
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
           egress : --

          rule id : 5
          vlan id : any
         protocol : tcp
           src ip : 10.0.0.1
           dst ip : 10.0.0.2
         src port : any
         dst port : any
     no of alias : 0
 sampling policy : sp1
     tunnel name : t1
    export ipfix : sampled-out
           egress : --

          rule id : 6
          vlan id : any
         protocol : any
           src ip : 10.0.0.1
           dst ip : 100.1.1.1
         src port : any
         dst port : any
     no of alias : 0
 sampling policy : --
     tunnel name : --
    export ipfix : all
           egress : drop

vtap> show config interface=all

interfaces information

                     name  : ip
             traffic type  : ip
                   egress  : drop
    vtap ingress ports  : ens224,ens256
                     mode  : session
             export IPFIX  : disabled

vtap> show config egress-alias=all

Egress alias configuration:
No egress alias configured

vtap> show config collector=all
No element configured

vtap> show config template=all
Template ID     : 300
        Set ID                          : 3
        Number of Scope Fields          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 0
Template ID     : 257
        Set ID                          : 2
        Number of Fields                : 24
```

```
        Number of Dependent Templates    : 1
                Dependent Template ID    : 300
Template ID     : 256
        Set ID                           : 2
        Number of Fields                 : 15
        Number of Dependent Templates    : 1
                Dependent Template ID    : 300
Template ID     : 258
        Set ID                           : 2
        Number of Fields                 : 8
        Number of Dependent Templates    : 1
                Dependent Template ID    : 300
Template ID     : 272
        Set ID                           : 2
        Number of Fields                 : 4
        Number of Dependent Templates    : 0
Template ID     : 271
        Set ID                           : 2
        Number of Fields                 : 5
        Number of Dependent Templates    : 0
Template ID     : 273
        Set ID                           : 2
        Number of Fields                 : 11
        Number of Dependent Templates    : 0

vtap> show config export-interval
Time for IPFIX Report Interval : 1 min
```

show device-info

Syntax

show device-info

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show device-info
Device Information
         NIC port  :  0
                 PCI Bus Location                                :     0000:13:00.00
                 Vendor id                                       :     15ad
                 Device id                                       :     7b0
                 Sub System Vendor id                            :     15ad
                 Sub System Device id                            :     7b0
                 Driver Name                                     :     net_vmxnet3
                 if_index                                        :     0
                 Interface Name                                  :
                 Minimum RX Buffer size                          :     1646
                 Minimum configurable length of RX Packet        :     16384
                 Maximum number of RX Queues                     :     16
                 Maximum number of TX Queues                     :     8
                 Maximum number of VFs                           :     0
                 Maximum number of VMDq pools                    :     0
                 RX offload capabilities                         :     29
                 TX offload capabilities                         :     45
                 Redirection table size                          :     0
                 Hash key size                                   :     0
                 RSS offloads                                    :     1300
                 First queue ID for VMDQ pools                   :     0
                 Queue number for VMDQ pools                     :     0
                 First ID of VMDQ pools                          :     0
                 Maximum number of RX allowed descriptors        :     4096
                 Minimum number of RX allowed descriptors        :     128
                 Number of RX descriptors should be aligned      :     1
                 Maximum number of TX allowed descriptors        :     4096
                 Minimum number of TX allowed descriptors        :     512
                 Number of TX descriptors should be aligned      :     1
```

show egress-alias

Use this command to display all the egress aliases and the settings configured for each egress alias.

Syntax

show egress-alias = { **all** | *egress_alias_value* }

Modes

Egress Configuration mode

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

```
vtap> en
Password:

vtap# configure t

vtap(config)# egress

vtap(config-egress)# show egress-alias=all
Egress alias configuration:

             egress alias  : default-egress
                    state  : active
                  vlan-id  : 4095

             egress alias  : egress-4
                    state  : inactive
                 local-ip  : 2001::1
           destination-ip  : 2001::2
         destination-port  : 4789
                      vni  : 10234
              tunnel-type  : vxlan
              nexthop-mac  : 11:22:33:44:55:66

             egress alias  : egress-5
                    state  : inactive
                  vlan-id  : 100

             egress alias  : egress-3
                    state  : inactive
                 local-ip  : 2001::1
           destination-ip  : 2001::2
              tunnel-type  : gre
              nexthop-mac  : 11:22:33:44:55:66

             egress alias  : egress-1
                    state  : active
                  vlan-id  : 100
```

show export-interval

Use this command to display the export interval timeout value.

Syntax

show export-interval

Modes

Flow Exporter Configuration mode

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# flow-exporter

vtap(config-flow-exporter)# show export-interval
Time for IPFIX Report Interval : 1 min
```

show flow-exporter config

Use this command to display collector and template configuration for flow exporter.

Syntax

show flow-exporter config { **collector** = { *collector_name* | **all** } | **template** = { *template_id* | **all** } }

Modes

Flow Exporter Configuration mode

Examples

Example 1

```
vtap> en
Password:
vtap# conf t
vtap(config)# flow-exporter

vtap-flow-exporter> show flow-exporter config collector=all
Collector ID         : 1
name                 : test
Collector Address    : 10.9.9.186
Export Address       : 10.9.9.184
Transport Protocol   : UDP
Collector Port       : 8888
Export Port          : 9999
```

Example 2

```
vtap> en
Password:
vtap# conf t
vtap(config)# flow-exporter

vtap-flow-exporter> show flow-exporter config template=all
Template ID     : 300
        Set ID                          : 3
        Number of Scope Fields          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 0
Template ID     : 257
        Set ID                          : 2
        Number of Fields                : 24
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 256
        Set ID                          : 2
        Number of Fields                : 15
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 258
        Set ID                          : 2
        Number of Fields                : 8
        Number of Dependent Templates   : 1
                Dependent Template ID   : 300
Template ID     : 272
        Set ID                          : 2
        Number of Fields                : 4
        Number of Dependent Templates   : 0
Template ID     : 271
        Set ID                          : 2
        Number of Fields                : 5
        Number of Dependent Templates   : 0
Template ID     : 273
        Set ID                          : 2
        Number of Fields                : 11
        Number of Dependent Templates   : 0
```

show flow-exporter stats

Use this command to display collector and template statistics for flow exporter.

Syntax

show flow-exporter stats { **collector** = { *collector_name* | **all** } | **template** = { *template_id* | **all** } }

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show flow-exporter stats template=all

Template ID          : 300
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 10

Template ID          : 257
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 0

Template ID          : 256
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 0

Template ID          : 258
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 5

Template ID          : 272
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 0

Template ID          : 271
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 0

Template ID          : 273
        Number of Template Sent        : 1
        Number of DataRecord Sent      : 0

vtap> show flow-exporter stats collector=all
Collector ID         : 1
name                 : test
Number Message Sent  : 7
Max Record Sent      : 0
Number Template Sent : 7
Number of Data Record Sent : 0
```

show flow-info

Use this command to display information for all the flows.

Syntax

show flow-info

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap# show flow-info
flow information:
 flow-id : 55
                parent id : 2
             parent-type : vlan
                host1-ip : fe80::5054:ff:fea8:fd43
                host2-ip : ff02::16
              host1-port : 0
              host2-port : 0
                protocol : icmpv6
          dir1 rx packets : 4284
            dir1 rx bytes : 3470040
          dir2 rx packets : 0
            dir2 rx bytes : 0
          dir1 tx packets : 4284
            dir1 tx bytes : 3470040
          dir2 tx packets : 0
            dir2 tx bytes : 0
             export-ipfix : disable
              sampled-out : no
      sampled out packets : 0
          dropped packets : 0
           smatch rule id : -
            egress-action : default-egress
                  l7-type : other

 flow-id : 24
                parent id : 2
             parent-type : vlan
                host1-ip : fe80::5054:ff:fe4f:e50a
                host2-ip : ff02::16
              host1-port : 0
              host2-port : 0
                protocol : icmpv6
          dir1 rx packets : 4284
            dir1 rx bytes : 3470040
          dir2 rx packets : 0
            dir2 rx bytes : 0
          dir1 tx packets : 4284
            dir1 tx bytes : 3470040
          dir2 tx packets : 0
            dir2 tx bytes : 0
             export-ipfix : disable
              sampled-out : no
      sampled out packets : 0
          dropped packets : 0
           smatch rule id : -
            egress-action : default-egress
                  l7-type : other
```

show ingress-ports

Use this command to display all the ingress ports.

NOTE

- Ingress ports are pre-configured and cannot be modified.
- Jumbo Frame for Rx (ens224) and Tx (ens256) ports is enabled by default.

Syntax

show ingress-ports

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap# show ingress-ports
-------------------------------------------------------
| Type            | Name            | Jumbo Frame     |
-------------------------------------------------------
| IP-PORT-RX      | ens1f0          | enabled         |
| IP-PORT-TX      | ens1f1          | enabled         |
-------------------------------------------------------
```

show interface

Use this command to display the interface configuration.

Syntax

show interface = { **all** | *interface_name* }

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface

vtap(config-interface)# show interface=all

                     name  : ip
             traffic type  : ip
                   egress  : default-egress
    sampling policy-name   : sp1
  smartmatch policy-name   : smp1
                     mode  : session
             export-ipfix  : disabled
             header-strip  : -
             packet-slice  : -
```

show interface-stats

Use this command to display interface statistics.

Syntax

show interface-stats

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show interface-stats
interface stats
    sampling policy-name : s1
                    sample-rate : 0
                     flow count : 0
                flow drop count : 0
               preserve packets : 0

      smatch policy-name : -

      egress alias name  : e1
               forwarded packets : 11

            header-strip : vxlan,802.1br_vntag,mpls,vlan
                   vxlan packets : 5
                     vxlan bytes : 258
                   nvgre packets : 0
                     nvgre bytes : 0
                 erspan2 packets : 0
                   erspan2 bytes : 0
                    gtpu packets : 0
                      gtpu bytes : 0
                    mpls packets : 0
                      mpls bytes : 0
                    vlan packets : 0
                      vlan bytes : 0
              802.1br_vntag packets : 22
                802.1br_vntag bytes : 88
               packet-slice : -
```

show link-status

Use this command to display the link status for all the ports.

Syntax

show link-status

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show link-status
Link Information
  NIC Port : 0 (Rx-Port)
        Link Status                                    :     UP
        Link Speed                                     :     10000
        Link Duplex                                    :     FULL
  NIC Port : 1 (Tx-Port)
        Link Status                                    :     UP
        Link Speed                                     :     10000
        Link Duplex                                    :     FULL
```

show logging

Use this command to display the log levels for various modules.

Syntax

show logging

Modes

Global Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t

vtap(config)# show logging
         module : level

    tunnel-flow : critical
  smatch-sample : critical
          ipfix : critical
           mgmt : critical
          L7app : critical
            kni : critical
```

show nic-stats

Use this command to display DPDK NIC statistics.

NOTE

The `show nic-stats` command does not display byte count details for Rx and Tx for e1000 vNIC driver.

Syntax

show nic-stats

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show nic-stats
NIC Stats
        NIC Port : 0 (Rx-Port)
                Total number of received packets: 4316941
                Total number of transmitted packets: 0
                Total number of received bytes: 3454113950
                Total number of transmitted bytes: 0
                Total number of RX packets dropped by hardware: 0
                Total number of erroneous received packets: 0
                Total number of failed transmitted packets: 0
                Total number of RX mbuf allocation failures: 0
        NIC Port : 1 (Tx-Port)
                Total number of received packets: 12334
                Total number of transmitted packets: 4329262
                Total number of received bytes: 800032
                Total number of transmitted bytes: 3472230228
                Total number of RX packets dropped by hardware: 0
                Total number of erroneous received packets: 0
                Total number of failed transmitted packets: 0
                Total number of RX mbuf allocation failures: 0
```

show packet-stats

Use this command to display packet statistics for Rx, Tx and egress.

NOTE

- In the output, values for `Total Packet Rate` and `Total Throughput` are updated every five seconds.
- The `show packet-stats all` command displays output without any pause. To add a pause after a set number of lines, run the `terminal length` command and set the value to 20. For more information, see the section terminal length on page 54.

Syntax

Syntax

show packet-stats={ **all** | **rx** | **tx** | **egress** | **smartmatch** }

Parameters

all

rx

tx

egress

smartmatch

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show packet-stats all

Rx Packet Statistics:
        Packet Stats for Port : ingress
                No Of Packets Received : 86312904
                No Of Packets Filtered : 0

Tx Packet Statistics:
        Packet Stats for Port : egress
                No of Packets Received : 86219114

total egress stats :
         dropped packets : 86312903
            dropped rate : 14
      dropped throughput : 0.009875
           total packets : 86312903
              total rate : 14
        total throughput : 0.009875

smartmatch stats:
                    packets received : 0
                      bytes received : 0
                   packets forwarded : 0
                     bytes forwarded : 0
                     packets dropped : 0
                       bytes dropped : 0
 packets not matched smartmatch rule : 0
      packets not matched flex match : 0

vtap> show packet-stats rx
Rx Packet Statistics:
        Packet Stats for Port : ingress
                No Of Packets Received : 86314157
                No Of Packets Filtered : 0

vtap> show packet-stats tx
Tx Packet Statistics:
        Packet Stats for Port : egress
                No of Packets Received : 86220399

vtap> show packet-stats egress
Egress Statistics:
egress name : default-egress
             egress type : vlan
            packet count : 4308870
             bytes count : 3455926660
             packet rate : 1
              throughput : 0.000513

Total Egress Statistics:
         dropped packets : 0
            dropped rate : 0
      dropped throughput : 0.000000
           total packets : 4308870
              total rate : 1
        total throughput : 0.000513

vtap> show packet-stats smartmatch

smartmatch stats:
                    packets received : 54181
                      bytes received : 40696207
                   packets forwarded : 54181
                     bytes forwarded : 40696207
                     packets dropped : 0
                       bytes dropped : 0
 packets not matched smartmatch rule : 357
      packets not matched flex match : 0
          packets matched flex match : 0
```

show rule stats

Use this command to display statistics for SmartMatch rules.

Syntax

show rule stats smatch-policy= *smartmatch_policy_name*

Modes

User EXEC mode

Privileged EXEC mode

Examples

Example

```
vtap# show rule stats smatch-policy=smatch-1
rule id : 1
              rx packets : 0
                rx bytes : 0
              tx packets : 0
                tx bytes : 0
         dropped packets : 0
         sampled packets : 0
        preserve packets : 0
         number of flows : 0

      header-strip stats :
                   vxlan packets : 0
                     vxlan bytes : 0
                   nvgre packets : 7
                     nvgre bytes : 364
                  erspan2 packets : 0
                    erspan2 bytes : 0
                     gtpu packets : 0
                       gtpu bytes : 0
                     mpls packets : 4
                       mpls bytes : 20
                     vlan packets : 0
                       vlan bytes : 0
           802.1br_vntag packets : 0
             802.1br_vntag bytes : 0

    packet-slice packets : 20
      packet-slice bytes : 486
```

show sampling-policy

Use this command to display all the configured Sampling policies.

Syntax

show sampling-policy = { **all** | *policy_name* }

Modes

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# show sampling-policy=all
policy name : sp1
           state : active
     sample-rate : 10
   preserve-pkts : 0
```

show sampling-stats

Use this command to display statistics for sampling policies.

Syntax

show sampling-stats *sampling_policy_name* | **all**}

Modes

User EXEC mode

Privileged EXEC mode

Examples

Example 1

```
vtap# show sampling-stats sampling-policy=all
    policy name : sp1
         smatch-policy: sm1
                 r    ule-id : 10
             sample-rate : 100
              total-flow : 1
               drop-flow : 1
         drop-pkts-count : 0
    preserve-pkts-count: 1


     smatch-policy : sm1
                 rule-id : 20
             sample-rate : 100
              total-flow : 1
               drop-flow : 1
         drop-pkts-count : 0
      preserve-pkts-count: 0


     smatch-policy : sm2
                 rule-id : 30
             sample-rate : 100
              total-flow : 1
               drop-flow : 1
         drop-pkts-count : 0
       preserve-pkts-count: 0



  policy name          : sp2
           sampling applied on interface
             sample-rate : 100
              total-flow : 1
               drop-flow : 1
         drop-pkts-count : 0
    preserve-pkts-count: 0


    smatch-policy: sm2
        rule-id : 11
sample-rate : 100
              total-flow : 1
               drop-flow : 1
         drop-pkts-count : 0
       preserve-pkts-count: 0
```

Example 2

```
vtap# show sampling-stats sampling-policy=sp1

policy name : sp1
smatch-policy : sm1
                   rule-id : 10
               sample-rate : 100
                total-flow : 1
                 drop-flow : 1
          drop-pkts-count : 0

smatch-policy : sm1
                   rule-id : 20
               sample-rate : 100
                total-flow : 1
                 drop-flow : 1
          drop-pkts-count : 0
     preserve-pkts-count: 10


smatch-policy : sm2

                   rule-id : 30
               sample-rate : 100
                total-flow : 1000
                 drop-flow : 1000
          drop-pkts-count : 12443
     preserve-pkts-count: 100
```

show smartmatch-stats

Use this command to display statistics for SmartMatch aliases and rules.

Syntax

show smartmatch-stats { **smatch-alias** | **rule rule-id**={*rule_id* | **all**}

Parameters

smatch-alias

Display SmartMatch alias-specific statistics.

rule

Display SmartMatch rule-specific statistics for a policy.

Modes

User EXEC mode

Privileged EXEC mode

Examples

Example 1

```
vtap> show smartmatch-stats smatch-alias
        flexmatch alias : smatch-alias-1
        smatch-policy : smatch-1
        rule id : 1
       packets matched : 13
       packets masked : 10
       packets forwarded/dropped : 13
        egress action : egress-1
```

Example 2

```
vtap# show smartmatch-stats rule rule-id=all
  smatch policy name : SM

rule id : 1
              rx packets : 90
                rx bytes : 10257
              tx packets : 90
                tx bytes : 14757
         dropped packets : 0
         sampled packets : 0
        preserve packets : 0
         number of flows : 1

      header-strip stats :
                   vxlan packets : 10
                     vxlan bytes : 224
                   nvgre packets : 0
                     nvgre bytes : 0
                  erspan2 packets : 0
                    erspan2 bytes : 0
                     gtpu packets : 0
                       gtpu bytes : 0
                     mpls packets : 0
                       mpls bytes : 0
                     vlan packets : 5
                       vlan bytes : 20
           802.1br_vntag packets : 0
             802.1br_vntag bytes : 0

    packet-slice packets : 10
      packet-slice bytes : 356
```

show smatch-policy

Use this command to display all the configured SMARTMatch policies.

Syntax

show smatch-policy = { **all** | *policy_name* }

Modes

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

Example

```
vtap> en
Password:
vtap# conf t
vtap(config)# smartmatch

vtap(config-smartmatch)# show smatch-policy=all

 smart-match policy 1
     name : smp1
    state : inactive
    rules :
          rule id : 1
          vlan id : any
         protocol : any
         host1 ip : 10.0.0.1
         host2 ip : 100.1.1.1
       host1 port : any
       host2 port : any
      no of alias : 0
  sampling policy : --
     tunnel name : --
    export ipfix : all
          egress : drop
    header-strip : -
    packet-slice : -

          rule id : 2
          vlan id : 400
         protocol : tcp
         host1 ip : 10.0.0.1
         host2 ip : 10.0.0.2
       host1 port : any
       host2 port : any
      no of alias : 1
            alias name 1 : a1
  sampling policy : sp1
     tunnel name : --
    export ipfix : sampled-out
          egress : --
    header-strip : vxlan,nvgre
    packet-slice : 72

          rule id : 3
          vlan id : any
         protocol : any
         host1 ip : 2001:43:12::1
         host2 ip : 2600:32:11::1
       host1 port : any
       host2 port : any
      no of alias : 0
  sampling policy : --
     tunnel name : --
    export ipfix : disable
          egress : egress-1
    header-strip : -
    packet-slice : -
```

show system-stats

Use this command to display vTAP system statistics.

Syntax

show system-stats

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap# show system-stats
                     Rx Packets  : 136
                       Rx Bytes  : 9864
                   Drop Packets  : 136
                     Drop Bytes  : 9864
                          Flows  : 25
                        Tunnels  : 0
                Not Sent Flows   : 0
            Tunnels Terminated   : 0
                Kni Rx Packets   : 136
                Kni Tx Packets   : 0
          IP Fragmented Packets  : 128
          IP Reassembly Packets  : 32
          Total Deleted Tunnels  : 0
            Total Deleted Flows  : 0
Total Packets With Decode Error  : 40
```

show tech

When you run this command, a tar file containing various log files and consolidated output of various `show commands` is created, which is useful for monitoring and troubleshooting vTAP. The tar file is created in the `/var/log` directory. Use this command only in conjunction with Extreme Networks Technical support.

Syntax

show tech { **brief** | **detail** }

Modes

User EXEC mode

Privileged EXEC mode

Examples

Example

```
vtap> show tech detail
................................................
Logs tar file copied to /var/log/LOG_127.0.0.1_190121_113131.tar

vtap show tech brief
.....................................
Logs tar file copied to /var/log/LOG_127.0.0.1_190121_113536.tar
```

show template

Use this command to display information about the flow exporter templates.

Syntax

show template = { **all** | *template_id* }

Modes

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

Example

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface

vtap(config-interface)# show template=all
Template ID          : 300
        Set ID               : 3
        Number of Scope Fields : 2
        Number of Fields     : 8
        Number of Dependent Templates : 0
Template ID          : 257
        Set ID               : 2
        Number of Fields     : 24
        Number of Dependent Templates : 1
                Dependent Template ID : 300
Template ID          : 256
        Set ID               : 2
        Number of Fields     : 15
        Number of Dependent Templates : 1
                Dependent Template ID : 300
Template ID          : 258
        Set ID               : 2
        Number of Fields     : 8
        Number of Dependent Templates : 1
                Dependent Template ID : 300
Template ID          : 272
        Set ID               : 2
        Number of Fields     : 4
        Number of Dependent Templates : 0
Template ID          : 271
        Set ID               : 2
        Number of Fields     : 5
        Number of Dependent Templates : 0
Template ID          : 273
        Set ID               : 2
        Number of Fields     : 11
        Number of Dependent Templates : 0
```

show terminal info

Use this command to display the value set for terminal length.

Syntax

show terminal info

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show terminal info
Terminal configuration
Terminal length              :24
```

show tunnel-info

Use this command to display information about the tunnels received by vTAP.

Syntax

show tunnel-info

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap# sh tunnel-info
tunnel information:
 tunnel-id : 1
        parent-tunnel-id : 3
             tunnel-type : gre
                   host1-ip : 1.1.1.1
                   host2-ip : 2.2.2.2
        dir-1 rx packets : 5
          dir-1 rx bytes : 0
        dir-2 rx packets : 5
          dir-2 rx bytes : 0
           child tunnels : 0
                   flows : 10
           egress-action : default-egress
```

show tunnel-name

Use this command to display all the configured ingress tunnels.

Syntax

show tunnel-name= { **all** | *tunnel_name* }

Modes

Interface Configuration mode

Sampling Configuration mode

SMARTMatch Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)#
vtap(config-smartmatch)# show tunnel-name=all
    tunnel-name: t1
       svlan: 100

    tunnel-name: t2
    svlan: 120
    cvlan: 1055

    tunnel-name: t3
    host1-ip: 2001::1
    host2-ip: 2002::2
    tunnel-type: gre

    tunnel-name: t4
    host1-ip: 2001::1
    host2-ip: 2002::2
    tunnel-type: ipip
```

show version

Use this command to display vTAP release version.

Syntax

show version

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> show version
2.0-0
```

Clear commands

This chapter provides information about the vTAP `clear` commands. All commands are listed in alphabetical order.

clear all

Use this command to clear all statistics.

Syntax

clear all

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> clear all
Cleared packet Tx Stats
Cleared packet Rx Stats
Cleared packet egress Stats
Cleared packet smartmatch-stats
Cleared all smartmatch-stats successfully
```

clear config

Use this command to clear running configuration.

Syntax

clear config

Modes

Global Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t

vtap(config)# clear config
Do you want to clear configuration (yes/y/no/n)?y
This command may take maximum time of 10 seconds
Clearing interface properties for ip
Info: mode is already set to session

Failed: Policy sp1 is still active

SMARTMatch policy  deleted successfully.

Clearing egress alias
Egress alias deleted successfully.
Egress alias deleted successfully.
Egress alias deleted successfully.
Egress alias deleted successfully.

Clearing intunnel
Ingress tunnel deleted successfully.

Clearing flexmatch alias
match criteria deleted successfully.


Flow-export collector deleted successfully.
Flow-export collector deleted successfully.
Flow-export report interval updated successfully.

Clearing log modules
Logging cleared successfully for tunnel-flow module
Logging cleared successfully for smatch module
Logging cleared successfully for sampling module
Logging cleared successfully for ipfix module
Logging cleared successfully for mgmt module
Logging cleared successfully for L7app module
Logging cleared successfully for kni module
Configuration cleared successfully
```

clear egress

Use this command to clear the egress settings configured on the interface.

Syntax

clear egress

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear egress
Interface "ip"  parameters "egress " cleared successfully.
```

clear export-ipfix

Use this command to clear the export-ipfix settings configured on the interface.

Syntax

clear export-ipfix

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear export-ipfix
Interface "ip"  parameters "export ipfix" cleared successfully.
```

clear header-strip

Use this command to remove the header stripping property that is configured on the interface.

Syntax

clear header-strip

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear header-strip
Interface "ip"  parameters "header-strip" cleared successfully.
```

clear interface-stats

Use this command to clear interface statistics for header stripping and packet slicing.

Syntax

clear interface-stats { header-stripping | packet-slicing }

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> clear interface-stats packet-slice
Cleared packet-slicing stats in interface-stats

vtap> clear interface-stats header-stripping
Cleared header-stripping stats in interface-stats
```

clear logging

Use this command to clear Warning, Info, and Debug log levels for various modules.

Syntax

clear logging { tunnel-flow | smatch | sampling | ipfix | mgmt | L7app | kni | all }

Modes

Global Configuration mode

Examples

```
vtap> enable
Password:
vtap# conf t

vtap(config)# clear logging
enter module=<tunnel-flow|smatch-sample|ipfix|mgmt|L7app|kni|all>

vtap(config)# clear logging module=all
vtap(config)# clear logging module=all
Logging cleared successfully for all module

vtap(config)# clear logging module=tunnel-flow
vtap(config)# clear logging module=tunnel-flow
Logging cleared successfully for tunnel-flow module
```

clear mode

Use this command to clear the vTAP mode and set it to the default value (session).

Syntax

clear mode

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear mode
Info: mode is already set to session

vtap(config-interface-ip)# set mode=packet
Sampling Policy and IPFIX settings on ip Interface are now disabled with Packet mode

vtap(config-interface-ip)# clear mode
Sampling Policy and IPFIX settings on ip Interface can now be applied with Session mode
```

clear packet-slice

Use this command to remove the packet slicing property that is configured on the interface.

Syntax

clear packet-slice

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear packet-slice
Interface "ip"  parameters "packet-slice" cleared successfully.
```

clear packet-stats

Use this command to clear all packet statistics.

Syntax

clear packet-stats { all | tx | rx | egress | smartmatch}

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> clear packet-stats all
Cleared Tx Stats
Cleared Rx Stats
Cleared Egress Stats

vtap> clear packet-stats tx
Cleared Tx Stats

vtap> clear packet-stats rx
Cleared Rx Stats

vtap> clear packet-stats egress
Cleared Egress Stats
```

clear preserve-pkts

Use this command to clear preserve packets settings configured for a Sampling policy.

Syntax

clear preserve-pkts

Modes

Sampling Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# edit sampling-policy=sp1
vtap(config-sampling-sp1)# clear preserve-pkts
sampling policy "sp1" with parameter "preserve-pkts" cleared successfully
```

clear sample-rate

Use this command to clear the sampling rate configured for a Sampling policy and set it to the default value of 1.

Syntax

clear sample-rate

Modes

Sampling Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# sampling

vtap(config-sampling)# edit sampling-policy=sp1
vtap(config-sampling-sp1)# clear sample-rate
sampling policy updated successfully.
```

clear sampling-policy

Use this command to remove the Sampling policy configured on the interface.

Syntax

clear sampling-policy

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear sampling-policy
Interface "ip"  parameters "Sampling policy sp1" cleared successfully.
```

clear smartmatch-stats

Use this command to clear SmartMatch statistics.

Syntax

clear smartmatch-stats { rule | all }

Modes

User EXEC mode

Privileged EXEC mode

Parameters

rule

Clear all SmartMatch rule statistics.

all

Clear all SmartMatch statistics.

Examples

```
vtap> clear smartmatch-stats rule
Cleared smartmatch rule stats successfully

vtap> clear smartmatch-stats all
Cleared all smartmatch- stats successfully
```

clear smatch-policy

Use this command to remove the SmartMatch policy configured on the interface.

Syntax

clear smatch-policy

Modes

Interface Configuration mode

Examples

```
vtap> en
Password:
vtap# conf t
vtap(config)# interface
vtap(config-interface)# edit interface=ip

vtap(config-interface-ip)# clear smatch-policy
Interface "ip"  parameters "Smartmatch policy smp1" cleared successfully.
```

Service commands

This chapter provides information about the vTAP `service` commands. All commands are listed in alphabetical order.

restart

Use this command to restart vTAP services.

Syntax

restart { vtap | vtap_snmp | vtap_nginx | all }

Parameters

vtap

vtap_snmp

vtap_nginx

all

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> en
Password:

vtap# restart all
vtap service restarted successfully
vtap-snmp service restarted successfully
vtap-nginx service restarted successfully

vtap# restart vtap
vtap service restarted successfully

vtap# restart vtap-snmp
vtap-snmp service restarted successfully

vtap# restart vtap-nginx
vtap-nginx service restarted successfully
```

status

Use this command to check the status of vTAP services.

Syntax

restart { vtap | vtap_snmp | vtap_nginx | all }

Parameters

vtap

vtap_snmp

vtap_nginx

all

Modes

User EXEC mode

Privileged EXEC mode

Examples

```
vtap> status all
vtap service status:
   vtap is initialized
        Physical contiguous memory is initialized
        Mempool and hash are initialized
        vtap is active now
   Active: active (running) since Wed 2018-09-26 18:51:05 IST; 1 weeks 1 days ago


vtap-snmp service status:
   Active: active (running) since Wed 2018-09-26 18:49:57 IST; 1 weeks 1 days ago

vtap-nginx service status:
   Active: active (running) since Wed 2018-09-26 18:49:57 IST; 1 weeks 1 days ago


vtap> status vtap
vtap service status:
   vtap is initialized
        Physical contiguous memory is initialized
        Mempool and hash are initialized
        vtap is active now
   Active: active (running) since Wed 2018-09-26 18:51:05 IST; 1 weeks 1 days ago


vtap> status vtap-snmp
vtap-snmp service status:
   Active: active (running) since Wed 2018-09-26 18:49:57 IST; 1 weeks 1 days ago


vtap> status vtap-nginx
vtap-nginx service status:
   Active: active (running) since Wed 2018-09-26 18:49:57 IST; 1 weeks 1 days ago
```

start

Use this command to start vTAP services.

NOTE

vTAP services take about three to four minutes to start. If the services do not start, check the logs in `/var/log/vtap.log`.

Syntax

restart { vtap | vtap_snmp | vtap_nginx | all }

Parameters

vtap

vtap_snmp

vtap_nginx

all

Modes

Privileged EXEC mode

Examples

```
vtap> en
Password:

vtap# start all
vtap service started successfully
vtap-snmp service started successfully
vtap-nginx service started successfully

vtap# start vtap
vtap service started successfully

vtap# start vtap-snmp
vtap-snmp service started successfully

vtap# start vtap-nginx
vtap-nginx service started successfully
```

stop

Use this command to stop vTAP services.

Syntax

restart { vtap | vtap_snmp | vtap_nginx | all }

Parameters

vtap

vtap_snmp

vtap_nginx

all

Modes

Privileged EXEC mode

Examples

```
vtap> en
Password:

vtap# stop all
vtap service stopped successfully
vtap-snmp service stopped successfully
vtap-nginx service stopped successfully

vtap# stop v
vtap vtap-snmp vtap-nginx
vtap# stop vtap
vtap service stopped successfully

vtap# stop vtap-snmp
vtap-snmp service stopped successfully

vtap# stop vtap-nginx
vtap-nginx service stopped successfully
```