

Technology Overview

Lawful Intercept Using Flow-Tap

Release
11.3



Published: 2011-09-22

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Technology Overview Lawful Intercept Using Flow-Tap

Release 11.3

Copyright © 2011, Juniper Networks, Inc.

All rights reserved.

Revision History

September 2011—R1 Junos OS 11.3

The information in this document is current as of the date listed in the revision history.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

Overview of Lawful Intercept by Using Flow-Tap	1
Primary Requirements for Lawful Intercept	1
Flow-Tap Features	2
Components of Flow-Tap	5
Configuring Flow-Tap Services for Lawful Intercept	7
Requirements for Establishing a Flow-Tap Session	7
Flow-Tap Topology	7
Configuring Flow-Tap Services	8
Flow-Tap Configuration on a Juniper Networks Router	10
Flow-Tap Filter Operation	13
Identifying and Capturing Target Packets Using Dynamic Filtering	13
Sample LEA Filter Configuration	13
Sample DTCP Parameter File	14
Provisioning Flow-Tap to a Linux Mediation Device	17
Flow-Tap Script for a Linux Device	17
Invoking a Perl Script from a Linux Device	18
Troubleshooting Flow-Tap	19
General Troubleshooting Steps	19
Troubleshooting No Packets Received by the Analyzer Software	20
Frequently Asked Questions About Using Flow-Tap	23

Overview of Lawful Intercept by Using Flow-Tap

This document explains flow-tap configuration, testing, and basic troubleshooting on Juniper Networks M Series Multiservice Edge Routers and Juniper Networks T Series Core Routers using Juniper Networks Junos® Platform configuration commands and third-party scripts. Flow-tap is the Junos operating system (Junos OS) application used for performing lawful intercept of targeted packet flows.

Lawful intercept (LI) is a process for obtaining communications network data related to an individual (a target), as authorized by a judicial or administrative order. To facilitate the lawful intercept process, certain legislation and regulations require service providers (SPs) and Internet service providers (ISPs) to explicitly support authorized electronic surveillance on their networks to facilitate the interception of telecommunications by law enforcement agencies (LEAs), regulatory or administrative agencies, and intelligence services, in accordance with local law.

Junos OS uses the **flow-tap** application to dynamically capture network flows as required for lawful intercept. Dynamic Tasking Control Protocol (DTCP) is the basis of the dynamic flow capture feature in Junos OS. Dynamic flow capture uses DTCP requests to capture packet flows based on dynamic filtering criteria.

The flow-tap application extends the use of DTCP and dynamic flow capture to intercept IPv4 and IPv6 packets in an active monitoring router, and sends a copy of packets that match filter criteria to one or more content destinations, including mediation devices and traffic analyzers.

Flow-tap is supported on Juniper Networks M Series and T Series routers, with the exception of the M160 routers and the TX Matrix routers.



NOTE: More information regarding DTCP can be found in Internet draft *draft-cavuto-dtcp-00.txt*, *DTCP: Dynamic Tasking Control Protocol* at <http://www.ietf.org/internet-drafts>.

Primary Requirements for Lawful Intercept

The primary requirements for a Juniper Networks (or any vendor's) device to participate in lawful intercept include:

- It must provide an interface and mechanism by which a mediation device can connect to the routing device and request a copy of packets sent to and from an intended target, based on the Layer 3 and Layer 4 parameters of the packet flow.
- It must maintain separation of different LEAs on the device. If multiple agencies are requesting LI action on the same device or for the same target, the agencies must not be aware of each other's presence, and they must not be able to see each other's LI configuration and status.
- The configuration for deciding which packet flow to capture (the identity of the target) and the intercept function on the router must be visible only to authorized personnel.

- The intended target of the LI interception must not be aware of the interception.
- A mediation device is required to simulate and test the interception application (flow-tap). The mediation device can be a Linux server with SSH capabilities.

Flow-Tap Features

These are the major features of the Junos OS flow-tap:

- Flow-tap uses DTCP for communication between a mediation device and a network device (router).
- LEA filters installed by one flow-tap user are not visible to another flow-tap user. This maintains separation between LEA users.
- The flow-tap function is applied on all routing instances; therefore, all traffic passing through the instance is subject to monitoring. This intercept function does not create any perceptible delay in forwarding the packets.
- The flow-tap configuration on the device does not reveal the identity of the monitored target. However, the services PIC running the flow-tap service is visible to all users on the routing device.
- Flow-tap and the dynamic flow capture feature cannot both be configured on the router at the same time; if attempted, the configuration will fail to commit.
- Only one instance of flow-tap service is supported per chassis; therefore, only one services PIC in the chassis can be used for flow-tap.
- Junos OS does not support flow-tap for virtual private LAN service (VPLS) and MPLS protocol packets.
- IPv4 and IPv6 intercept filters can coexist on a system, subject to a combined maximum of 100 filters.
- Flow-tap supports up to 20 filters per chassis; therefore, each chassis can monitor a maximum of bidirectional traffic from 10 users.
- For flow-tap, a services PIC supports a total limit of 20 Kpps ingress and 20 Kpps egress traffic, assuming 256 bytes per packet. Carefully select the match conditions in the LEA filter so that these limits are not exceeded when traffic comes from a high-speed interface. Ensure that only necessary traffic is sent by the Packet Forwarding Engine firewall filter to the services PIC.
- Graceful Routing Engine switchover can be configured, but upon switchover, the services PIC and the dynamic flow capture process restart and all LEA filters are lost. If this occurs, the mediation device is expected to replay the LEA filters to the router.
- Target packets are those that match the LEA filters. The target packets are appended with an IP/UDP header and then sent from the services PIC to a content destination. If the target packets need to have high priority for queuing while forwarding, an input filter should be configured in CLI and be visible to all users on the logical interface of the services PIC on which flow-tap is configured.
- Client tools running on the mediation device are not provided by Juniper Networks.

**Related
Documentation**

- [Components of Flow-Tap on page 5](#)
- [Configuring Flow-Tap Services for Lawful Intercept on page 7](#)
- [Flow-Tap Filter Operation on page 13](#)
- [Frequently Asked Questions About Using Flow-Tap on page 23](#)
- [Provisioning Flow-Tap to a Linux Mediation Device on page 17](#)
- [Troubleshooting Flow-Tap on page 19](#)

Components of Flow-Tap

This section describes the major components used in lawful intercept (LI) applications, including **flow-tap**.

Analyzer Device

An analyzer device is used for reporting and analyzing the captured data. The analyzer can be a hardware device separate from the mediation device, or it can be a single hardware device that integrates mediation and analyzer software. Analyzer devices are provided by outside vendors.

For testing and simulation, you can use Wireshark open source software for multiplatform protocol analysis, or a similar software.

Content Destination

The content destination is the recipient of the matched packets from the monitoring device. Typically the matched packets are sent using an IP Security (IPsec) tunnel from the monitoring device to another router connected to the content destination. The content destination and the mediation device can be physically located on the same host.

Dynamic Filters

When flow-tap is configured and the law enforcement agency (LEA) filters have been provisioned, the Packet Forwarding Engine automatically generates a firewall filter that is applied to all routing instances. Each term in the filter includes a **flow-tap** action; when at least one of the filter terms matches an incoming packet, the router copies the packet and forwards it to the services PIC that is configured for flow-tap. The services PIC runs the packet through the client filters and sends a copy to each matching content destination.

Dynamic Flow Capture

The dynamic flow capture process parses dynamic flow capture configurations and performs related tasks. The dynamic flow capture process (**dfcd**) on Juniper Networks devices can be used either for dynamic flow capture or for flow-tap, but not for both at the same time.

Flow

For dynamic flow capture, a flow is a stream of IP packets that flow in a direction with identical Layer 3 and Layer 4 information. It is possible to use wildcards in any of the Layer 3 and Layer 4 fields. MPLS packets are not considered to be a flow and cannot be captured by dynamic flow capture.

Mediation Device

A mediation device is a client that monitors electronic data or voice transfer over the network. It is supplied by a third-party vendor to handle the majority of the processing for LI, including providing the interface for the LI, generating requests to network devices for flow-tap applications, receiving packets that match filter criteria from a router,

converting intercepted traffic into the format required by the requesting LEAs, and forwarding copies of intercepted traffic to requesting LEAs. All of this activity is unknown to the target.

Authorized personnel of an Internet service provider (ISP) communicate with a Juniper Networks routing device from a mediation device using Dynamic Tasking Control Protocol (DTCP) over a secure channel (SSH). A user ID that has a flow-tap permission bit explicitly configured in the command-line interface or through RADIUS is available on the router to the authorized personnel.

Multiple mediation devices can communicate with a single Juniper device over multiple SSH sessions. The client software running at the mediation device is not provided by Juniper Networks.

Monitoring Platform

The monitoring platform processes the requests from the mediation devices, applies the dynamic filters, monitors incoming data flows, and sends the matched packets to the appropriate content destinations. The Juniper Networks monitoring platform for LI is an M Series or T Series router (except M160 routers and TX Matrix routers) with one or more Adaptive Services (AS) or Multiservices PICs configured to support the flow-tap application.

Parameter File

When the filters for flow-tap are provisioned by a Linux device, a parameter file must be supplied containing details about the flow, the traffic analyzer device location, and the new Layer 4 header details of the resultant packet. This information is interpreted by the dynamic flow capture process and passed to the services PIC. For more information about the parameter file, see [“Flow-Tap Filter Operation” on page 13](#).

Relay Agent

A DTCP relay agent (RA) is spawned by the SSH process whenever a mediation device opens an SSH connection on the flow-tap port. The RA is responsible for authenticating the user using a RADIUS or CLI-configured list of flow-tap users. Upon successful authentication, the RA establishes a UNIX domain socket connection with the dynamic flow capture process and hands over its UNIX socket side to the SSH process. At this point, the RA process ends.

Related Documentation

- [Configuring Flow-Tap Services for Lawful Intercept on page 7](#)
- [Flow-Tap Filter Operation on page 13](#)
- [Frequently Asked Questions About Using Flow-Tap on page 23](#)
- [Overview of Lawful Intercept by Using Flow-Tap on page 1](#)
- [Provisioning Flow-Tap to a Linux Mediation Device on page 17](#)
- [Troubleshooting Flow-Tap on page 19](#)

Configuring Flow-Tap Services for Lawful Intercept

This section includes the following topics:

- [Requirements for Establishing a Flow-Tap Session on page 7](#)
- [Flow-Tap Topology on page 7](#)
- [Configuring Flow-Tap Services on page 8](#)
- [Flow-Tap Configuration on a Juniper Networks Router on page 10](#)

Requirements for Establishing a Flow-Tap Session

The following steps are required to establish a flow-tap session for lawful intercept:

1. A law enforcement agency (LEA) obtains a court warrant to capture packets to and from the intended target and provides it to the authorized user at the Internet service provider (ISP).
2. Using a mediation device administered by an LEA, an authorized user issues a Dynamic Tasking Control Protocol (DTCP) request over SSH to the routing device to add a flow-tap filter for the targeted flow. The selected flow is known only to the LEA administrator of the mediation device.
3. At the router, the DTCP request is handed over to the dynamic flow capture process on the Routing Engine. The dynamic flow capture process takes appropriate actions, which typically include:
 - Issuing a request to the firewall process on the Routing Engine to add the filter to the Packet Forwarding Engine.
 - Issuing a request to the services PIC to run a flow-tap service and add the filter to its software state.
4. The services PIC appends an IP/UDP header onto matched packets and sends the appended packets to any mediation devices that have matching filters for that traffic.

Flow-Tap Topology

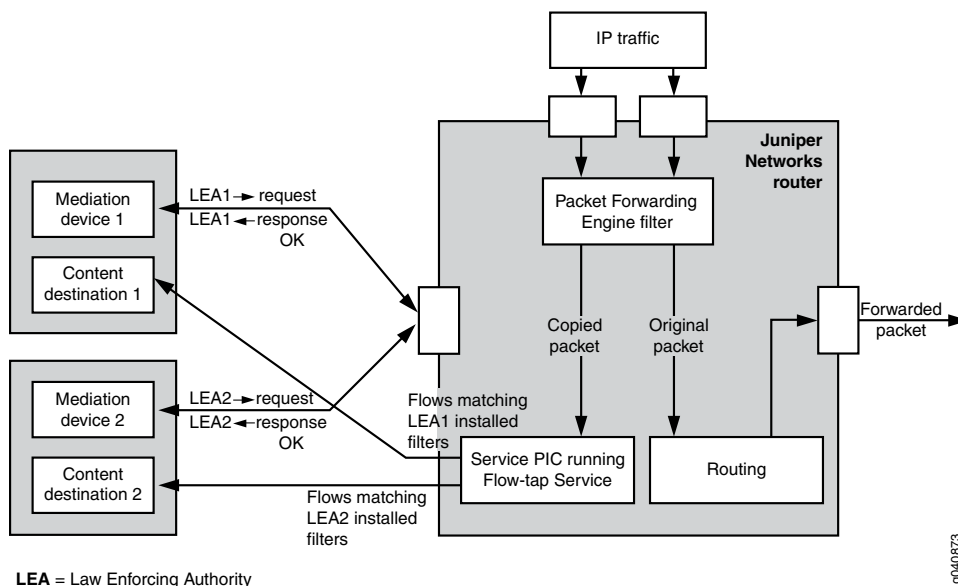
An example of flow-tap topology is shown in [Figure 1 on page 8](#). The architecture in the example consists of two mediation devices, each from a different agency, that send requests to a Juniper Networks router to monitor incoming data and forward any packets matching specific filter criteria to one or more content destinations.

A services PIC runs the flow-tap service and conveys flow-tap filters from the mediation devices to the router over an SSH channel. The filters are automatically installed in the services PIC and in the Packet Forwarding Engine, and are applied on all traffic. The services PIC receives packets matching the filters from the Packet Forwarding Engine, appends an IP/UDP header, and sends the filtered packets to the content destination specified for each LEA.

The original packets are forwarded to their intended destination, with no perceptible delay from the flow-tap interception, and with no knowledge of the interception by the

intended target or by any other clients that may also be applying flow-tap filters to this or any other targeted flow.

Figure 1: Flow-Tap Topology



Configuring Flow-Tap Services

In the Junos operating system (Junos OS), the flow-tap application creates firewall filters and pushes them to all the active Packet Forwarding Engines. This method captures the matching flows occurring in any interface and any Packet Forwarding Engine and sends the flow to the services PIC. The services PIC constructs a new IP header, using the details in the parameter file.

Use the following steps to perform basic configuration of flow-tap:

1. Configure flow-tap services by including the **flow-tap** statement at the **[edit services]** hierarchy level.



NOTE: Other statements are configured at the **[edit interfaces]** and **[edit system]** hierarchy levels.

```
flow-tap {
  interface interface-name;
}
```

2. Configure a services PIC interface by including the **interface** statement at the **[edit services flow-tap]** hierarchy level. You can assign any Adaptive Services (AS) or Multiservices PIC in the active monitoring router for flow-tap service, and use any logical unit on the PIC.

```
sp-fpc/pic/port.unit-number;
```

3. You must also configure the logical interface at the **[edit interfaces]** hierarchy level:

```

sp-fpc/pic/port {
  unit logical-unit-number {
    family inet;
    family inet6;
  }
}

```



NOTE: If you do not include the **family inet6** statement in the configuration, IPv6 flows will not be intercepted.

4. Configure flow-tap Dynamic Tasking Control Protocol (DTCP) by including the **flow-tap-dtcp** statement at the **[edit system services]** hierarchy level. This enables DTCP sessions on top of the SSH layer, providing authentication and privacy for lawful intercept (LI) flow-tap users. The **flow-tap-dtcp** service can be used only by those who have the flow-tap operation permission bit set in their login class and in the RADIUS server.



NOTE: You cannot configure dynamic flow capture and flow-tap features on the same router simultaneously.

In the following DTCP configuration, **connection-limit** is the maximum number of allowed connections (default = 75) and **rate-limit** is the maximum number of connections per minute (default = 150).

```

system {
  services {
    flow-tap-dtcp {
      ssh {
        connection-limit 5;
        rate-limit 5;
      }
    }
  }
}

```

5. Configure client permissions for viewing and modifying flow-tap configurations and for receiving targeted traffic by including the **permissions** statement at the **[edit system login class class-name]** hierarchy level.

```
permissions [permissions];
```

The permissions options available to use flow-tap features are:

```

flow-tap—Can view flow-tap configuration
flow-tap-control—Can modify flow-tap configuration
flow-tap-operation—Can tap flows

```

6. Specify RADIUS server user permissions for flow-tap by using the defined attribute **Juniper-User-Permissions**. This attribute lets the RADIUS server match flow-tap permission bits to those specified in RADIUS or in a local user's login class. The permission bits are read by the Junos OS on the router.

When using the RADIUS method of specifying permission bits, the bit granting permission to LI does not need to be on the Juniper device, so anyone looking at the configuration on the Juniper device can neither see nor guess who the LI users are.

```
Bob Auth-Type := Local, User-Password = = "abc123"  
Juniper-User-Permissions = "flow-tap-operation"
```

Then, if the RADIUS server returns:

```
Juniper-User-Permissions = "flow-tap-operation configure"
```

and the user's login class in the configuration specifies permission for:

```
[admin system]
```

then the user effectively has this permission:

```
[flow-tap-operation configure admin system]
```

Flow-Tap Configuration on a Juniper Networks Router

The following shows an example flow-tap configuration on a Juniper Networks router:

```
system {  
  login {  
    class VERINT {  
      idle-timeout 30;  
      permissions flow-tap-operation;  
    }  
    user verint {  
      uid 2008;  
      class VERINT;  
      authentication {  
        encrypted-password "$1$cUeV8XKs$28nf0JiRoDeYdE7lj4/9q.";  
      }  
    }  
  }  
  services {  
    flow-tap-dtcp {  
      ssh {  
        connection-limit 5;  
        rate-limit 5;  
      }  
    }  
  }  
}  
chassis {  
  fpc 0 {  
    pic 2 {  
      adaptive-services {  
        service-package layer-3;  
      }  
    }  
  }  
}  
interfaces {  
  sp-0/2/0 {  
    unit 100 {
```



```
        family inet;  
        family inet6;  
    }  
}  
services {  
    flow-tap {  
        interface sp-0/2/0.100;  
    }  
}
```

**Related
Documentation**

- [Components of Flow-Tap on page 5](#)
- [Flow-Tap Filter Operation on page 13](#)
- [Frequently Asked Questions About Using Flow-Tap on page 23](#)
- [Overview of Lawful Intercept by Using Flow-Tap on page 1](#)
- [Provisioning Flow-Tap to a Linux Mediation Device on page 17](#)
- [Troubleshooting Flow-Tap on page 19](#)

Flow-Tap Filter Operation

This section includes the following topics:

- [Identifying and Capturing Target Packets Using Dynamic Filtering on page 13](#)
- [Sample LEA Filter Configuration on page 13](#)
- [Sample DTCP Parameter File on page 14](#)

Identifying and Capturing Target Packets Using Dynamic Filtering

These are the steps used by dynamic filtering in flow-tap for identifying and capturing target packets:

1. If one of the filter terms matches an incoming packet, a copy of the packet is made and sent to the services PIC.
2. The services PIC receives the packet and runs it through all law enforcement agency (LEA) filters again. Then it sends a copy of the packet to each matching LEA, after adding a corresponding IP/UDP header. The Internet service provider can tunnel these packets using an IPsec tunnel to the mediation device.
3. The mediation device receives the packet and stores it or forwards it to each LEA, in a format specified by the receiving LEA.

Sample LEA Filter Configuration

The following is an example law enforcement agency (LEA) filter configuration as it would be on the router. However, the LEA filter configuration is not visible in the router configuration. It is dynamically generated by the router and no user configuration is required.

```
filter combined_LEA_filter {
  term LEA1_filter {
    from {
      source-address 1.2.3.4;
      destination-address 3.4.5.6;
    }
    then {
      flow-tap;
    }
  }
  term LEA2_filter {
    from {
      source-address 10.1.1.1;
      source-port 23;
    }
    then {
      flow-tap;
    }
  }
}
```

Sample DTCP Parameter File

Table 1 on page 14 describes each line of a sample Dynamic Tasking Control Protocol (DTCP) parameter file that would be sent from the mediation device to the router. The parameters program the router to capture packets sent to the router, then send the packet with a new law enforcement agency (LEA) header of source address = 10.209.75.199, destination address = 192.168.3.2, source port = 65534 and destination port = 1814.

It is assumed that the destination is a packet analyzer device whose functionality is separate from that of the mediation device.



NOTE: Ensure that the parameters selected will not cause the captured data to overwhelm the packet analyzer device beyond its processing capacity.

Table 1: Lines of Sample DTCP Parameter File

Line	Command	Description
1.	ADD DTCP/0.6	This indicates the DTCP version to be used. DTCP/0.6 should be used for all versions of Junos OS up to and including Junos OS 8.5. DTCP/0.7 should be used for Junos OS 9.0 and later. However, Junos OS 9.5R2 and later also accept previous versions of DTCP. If any unsupported parameters are received for a particular DTCP version, the request is rejected.
2.	Csource-ID: verint	This line specifies the username of the owner of the filter (verint in this example). This should be the same username specified under the system login.
3.	Cdest-ID: cd1	This line identifies the mediation device (cd1 in this example). This is only for administrator reference.
4.	Source-Address:	Lines 4 through 8 identify the desired target flow to be captured. The identifiers can be wildcards or absolute IP addresses or port numbers.
5.	Dest-Address:	
6.	Source-Port:	
7.	Dest-Port:	
8.	Protocol: 6	
9.	Flags: STATIC	

Table 1: Lines of Sample DTCP Parameter File (*continued*)

Line	Command	Description
10.	X-JTap-Cdest-Dest-Address: 192.168.3.2	Lines 10 through 14 define the LEA IP header fields. A captured target packet is appended with a header with these values before it is sent to the content destination.
11.	X-JTap-Cdest-Dest-Port: 1814	
12.	X-JTap-Cdest-Source-Address: 10.209.75.199	The source address is the operational IP address.
13.	X-JTap-Cdest-Source-Port: 65534	
14.	X-JTap-Cdest-TTL: 255	
15.	Seq:10	The sequence number identifies the "version" of flow-tap parameters being used. It is incremented each time the LEA reprograms the parameters and is tracked by the router. The router looks for a newer sequence number before accepting and implementing new parameters. Any configuration attempt with an older sequence number is rejected by the dynamic flow capture process.

**Related
Documentation**

- [Components of Flow-Tap on page 5](#)
- [Configuring Flow-Tap Services for Lawful Intercept on page 7](#)
- [Frequently Asked Questions About Using Flow-Tap on page 23](#)
- [Overview of Lawful Intercept by Using Flow-Tap on page 1](#)
- [Provisioning Flow-Tap to a Linux Mediation Device on page 17](#)
- [Troubleshooting Flow-Tap on page 19](#)

Provisioning Flow-Tap to a Linux Mediation Device

This section includes the following topics:

- [Flow-Tap Script for a Linux Device on page 17](#)
- [Invoking a Perl Script from a Linux Device on page 18](#)

Flow-Tap Script for a Linux Device

This section presents an example Linux Expect script for sending flow-tap parameters to a Linux server. The content and distribution of this script are subject to terms and conditions of Juniper Networks, Inc.

```
use Expect;
use Digest::HMAC_SHA1 qw(hmac_sha1 hmac_sha1_hex);

if ($#ARGV != 3) {
    die("Usage: dfcclient.pl <router> <user_name> <password> <input_file>\n");
}
my $exp = new Expect;

my $router = $ARGV[0];
my $user = $ARGV[1];
my $password = $ARGV[2];
my $input_file = $ARGV[3];
my $command = "ssh -l $user -p 32001 $router -s flow-tap-dtcp";
my $key = "Juniper";
my $digest;
my $hexdata;
my $dtcp_cmd = "";

print "$command\n";
$exp->raw_pty(1);
$exp->spawn($command) or die "Cannot spawn $command: $!\n";

$exp->expect(15, '-re', "password:");
$exp->send("$password\n");
sleep 3;
print "\n";

open(DAT, $input_file) || die("Could not open file!");
@raw_data=<DAT>;

foreach $line (@raw_data)
{
    chomp($line);
    if ($line eq "") {
        $digest = hmac_sha1($dtcp_cmd, $key);

        # converts binary to hex
        $hexdata = unpack("H*", $digest);

        $dtcp_cmd = $dtcp_cmd . "Authentication-Info: " . $hexdata . "\r\n\r\n";
        print "Sending DTCP cmd:\n" . $dtcp_cmd;
```

```

$exp->send($dtcp_cmd);
$dtcp_cmd = "";
sleep 1;
} else {
    $dtcp_cmd = $dtcp_cmd . $line . "\r\n";
}
}

$exp->interact();
=====

```

Invoking a Perl Script from a Linux Device

The following example shows the syntax to invoke the Perl script from a Linux device:

1. Invoke the Perl script:

```

[root@blr-e flowtap]# ./dfcclient.pl
Usage: dfcclient.pl <router> <user_name> <password> <input_file>
[root@blr-e flowtap]#

```

2. Use the following line to push the parameter file **lea1_tcp.flowtap** to the router. In this example, 10.209.75.199 is the IP address of the router, and **verint verint123** is the username and password that has permission to implement **flow-tap-operation**. Any firewall that is between the mediation device and the routing device should allow **ssh** and port 32001.

```

[root@blr-e flowtap]# ./dfcclient.pl 10.209.75.199 verint verint123 lea1_tcp.flowtap

```

3. Use the **show policer | match flow** statement to verify that the flow-tap filter is present on the router:

```

user@host-M320> show policer | match flow

Flowtap-Internal_IFL-ID-77          183364          2575
Flowtap-Internal_IFL-ID-76          150038          1531
Flowtap-Internal_IFL-ID-75          244504           626

```

Related Documentation

- [Components of Flow-Tap on page 5](#)
- [Configuring Flow-Tap Services for Lawful Intercept on page 7](#)
- [Flow-Tap Filter Operation on page 13](#)
- [Frequently Asked Questions About Using Flow-Tap on page 23](#)
- [Overview of Lawful Intercept by Using Flow-Tap on page 1](#)
- [Troubleshooting Flow-Tap on page 19](#)

Troubleshooting Flow-Tap

This section includes the following topics:

- [General Troubleshooting Steps on page 19](#)
- [Troubleshooting No Packets Received by the Analyzer Software on page 20](#)

General Troubleshooting Steps

If your flow-tap application does not seem to be operating as expected, use the following steps to troubleshoot.

1. Check the configuration:
 - Ensure that the configuration matches the configuration in the “Flow-Tap Configuration on Juniper Networks Router” section within “[Configuring Flow-Tap Services for Lawful Intercept](#)” on page 7 of this document; all components must be present.
 - Ensure that Layer 3 services have been enabled for the services PIC (MS-400 PIC).
 - Ensure that the **sp-x/y/z** interface is present.
 - Ensure that **policer** is present. Verify this by using the **show policer** CLI command.
2. Log in to the services PIC and issue the **show flow-tap services summary** statement. Verify that flow-tap actions have occurred by observing the packet count for packets processed to date.
3. Review information for the analyzer device and ensure that packets have been received by that device.
4. Issue the **monitor inter sp-0/2/0.100** command. From the output, determine whether packets have been received and processed by the services PIC interface.
5. Issue the **show log dfcd** command to see details from the parameter file that is being accepted by the router. If any of the information reported is inaccurate, this indicates a possible source of the problem. Example output follows:

show log dfcd

```
Aug 25 11:16:49 dfc_proto_pkt_handler: packet handling begins
Aug 25 11:16:49 Msg:
ADD DTCP/0.6
Csource-ID: verint
Cdest-ID: cd1
Source-Address: *
Dest-Address: *
Source-Port: *
Dest-Port: *
Protocol: 17
Flags: STATIC
X-JTap-Cdest-Dest-Address: 192.168.3.2
X-JTap-Cdest-Dest-Port: 1814
X-JTap-Cdest-Source-Address: 10.209.74.183
X-JTap-Cdest-Source-Port: 65534
```

```
X-JTap-Cdest-TTL: 255
Seq: 1
Authentication-Info: 3db78d10deb83f8934f021e40dc2b49f45bc6dc7
```

Troubleshooting No Packets Received by the Analyzer Software

Use the following troubleshooting steps if you determine that no packets are being received by the analyzer software:

1. Ensure that the target flow is occurring in the Juniper Networks router by installing a firewall filter counter on the interface toward the intended destination. If the counter increments, this indicates that packets are being sent, and the problem is with the flow-tap.
2. Next, log in to each of the Flexible PIC Concentrators (FPCs) and look for the installation of the firewall filters. You will see information similar to this example:

```
user@FFPC1# show filter
```

```
Program Filters:
```

Index	Dir	Cnt	Text	Bss	Name
3	96	0	20	0	__default_bpdu_filter__
17000	48	0	4	20	__default_arp_policer__
57023	240	576	40	96	__flowtap_inet__
65280	48	0	4	0	__auto_policer_template__
65281	96	0	16	0	__auto_policer_template_1__

```
user@FFPC1# show filter index 57023 counter
```

```
Filter Counters/Policers:
```

Index	Packets	Bytes	Name
57023	0	0	Flowtap-Internal_IFL-ID-77
57023	0	0	Flowtap-Internal_IFL-ID-76
57023	0	0	Flowtap-Internal_IFL-ID-75
57023	0	0	
Csource-verint__Cdest-cd2__ID-2			

3. If you do not see information for the filters, try restarting the **dynamic-flow-capture** process, then reapply the parameter file from the mediation device. Configure the syslog logging level to **any** and observe the messages file for the login activity. The following is a sample output:

```
Oct  2 13:48:44 FFPC1 sshd[27410]: Accepted password for verint from
172.24.18.168 port 2902 ssh2
Oct  2 13:48:45 FFPC1 ssh-relay[27414]: user 'verint' requesting
'flow-tap-dtcp' service
Oct  2 13:48:45 FFPC1 ssh-relay[27414]: connected as user 'verint' to
'flow-tap-dtcp' server
Oct  2 13:48:45 FFPC1 dfcd[27182]: New 'flow-tap-dtcp' relay connection
from user: verint host: 172.24.18.168 auth: 1
```

- Related Documentation**
- [Components of Flow-Tap on page 5](#)
 - [Configuring Flow-Tap Services for Lawful Intercept on page 7](#)

- [Flow-Tap Filter Operation on page 13](#)
- [Frequently Asked Questions About Using Flow-Tap on page 23](#)
- [Overview of Lawful Intercept by Using Flow-Tap on page 1](#)
- [Provisioning Flow-Tap to a Linux Mediation Device on page 17](#)

Frequently Asked Questions About Using Flow-Tap

Can port-mirroring be used when flow-tap is configured?

When flow-tap is configured, port-mirroring may not work for certain interfaces, due to sampling hardware limitations. On these interfaces, if a packet matches more than one sampling class, with each having a next hop programmed, then only one of the next hops can be chosen. Both flow-tap and port-mirroring use next-hop sampling, so any traffic through these interfaces that is marked for both flow-tap and port-mirroring will default to flow-tap, and no port-mirroring will be used for these packets.

If port-mirroring is configured for other types of interfaces that do not have sampling limitations, it will continue to work as expected.

Can syslog be used when flow-tap is configured?

The filter action **then syslog** (to forward packets to the syslog server) cannot be configured for any firewall filter if flow-tap is configured on the same platform. The flow-tap configuration commit will fail if the **then syslog** filter action is configured on any filter. This is true for all platforms on which flow-tap is supported. This helps ensure the security of the target packets.

What is flow-tap-lite?

Flow-tap-lite is a version of flow-tap in which all of the functionality is performed on the Packet Forwarding Engine, instead of on the services PIC as in the full-featured flow-tap. This is the only version of flow-tap supported on MX Series platforms, M120 routers, and M320 routers with Enhanced III FPCs only.

For **flow-tap-lite**, everything is specified under the flow-tap hierarchy only. Under **[edit services flow-tap]**, use the **tunnel-interface** option to specify the **flow-tap-lite** feature. Use the **interface** option if you want to use the full-featured flow-tap based on the services PIC.

Related Documentation

- [Components of Flow-Tap on page 5](#)
- [Configuring Flow-Tap Services for Lawful Intercept on page 7](#)
- [Flow-Tap Filter Operation on page 13](#)
- [Overview of Lawful Intercept by Using Flow-Tap on page 1](#)
- [Provisioning Flow-Tap to a Linux Mediation Device on page 17](#)
- [Troubleshooting Flow-Tap on page 19](#)

