

Chapter 27

Configure Flow Collection

You can process and export multiple cflowd records with a flow collector interface. You create a flow collector interface on a Monitoring Services PIC. The flow collector interface combines multiple cflowd records into a compressed ASCII data file and exports the file to an FTP server. To convert a Monitoring Services PIC into a flow collector interface, include the `flow-collector` statement at the [edit chassis fpc *fpc-slot* pic *pic-slot* monitoring-services application] hierarchy level. You can use the Monitoring Services PIC for either flow collection or monitoring, but not for both types of service simultaneously. When converting the PIC between service types, you must configure the `flow-collector` statement, take the PIC offline, and then bring the PIC back online. Restarting the router does not enable the new service type.

A flow collector interface, designated by the `cp-fpc/pic/port` interface name, requires three logical interfaces for correct operation. Units 0 and 1 are used to send the compressed ASCII data files to an FTP server, while Unit 2 is used to receive cflowd records from a monitoring services interface.



NOTE: Unlike conventional interfaces, the address statement at the [edit interfaces `cp-fpc/pic/port` unit *unit-number* family inet] hierarchy level corresponds to the IP address of the Routing Engine. Likewise, the destination statement at the [edit interfaces `cp-fpc/pic/port` unit *unit-number* family inet address *ip-address*] hierarchy level corresponds to the IP address of the flow collector interface. As a result, you must configure the destination statement for Unit 0 and 1 with *local* addresses that can reach the FTP server. Similarly, configure the destination statement for Unit 2 with a *local* IP address so it can reach the monitoring services interface that sends cflowd records.

To activate flow collector services after the Monitoring Services PIC is converted into a flow collector, include the `flow-collector` statement at the [edit services] hierarchy level. After you activate the flow collector, you need to configure the following components:

- Destination of the FTP server

- File specifications

- Input interface-to-flow collector interface mappings

- Transfer log settings

To configure flow collection, you include the following statements at the [edit services] hierarchy level of the configuration:

```

[edit services]
flow-collector {
  analyzer-address address;
  analyzer-id name;
  destinations {
    ftp:url {
      password "password";
    }
  }
  file-specification {
    variant variant-number {
      data-format format;
      name-format format;
      transfer {
        record-level number;
        timeout seconds;
      }
    }
  }
  interface-map {
    collector interface-name;
    file-specification variant-number;
    interface-name {
      file-specification variant-number;
      collector interface-name;
    }
  }
  retry number;
  retry-delay seconds;
  transfer-log {
    destinations {
      ftp:url {
        password "password";
        username username;
      }
    }
    filename "file-name";
    interval minutes;
    maximum-size number;
  }
}

```

This chapter contains the following sections:

Configure Flow Collection Properties on page 335

Send cflowd Records to the Flow Collector Interface on page 338

Enable Flow Collection Mode and Interface on page 338

Example: Flow Collector Interface Configuration on page 339

Configure Flow Collection Properties

This section describes the following tasks for configuring flow collection:

Configure Flow Collector Destinations on page 335

Configure a Packet Analyzer on page 335

Configure File Formats on page 336

Configure Interface Mappings on page 336

Configure Transfer Logs on page 337

Configure Retry Attempts on page 337

Configure Flow Collector Destinations

Flow collection destinations are where the compressed ASCII data files are sent after the cflowd records are collected and processed. To specify the destination FTP server, include the destinations statement at the [edit services flow-collector] hierarchy level. You can specify up to two FTP server destinations and include the password for each configured server. If two FTP servers are configured, the first server in the configuration is the primary server and the second is a backup server.

To configure a destination for flow collection files, include the destinations statement at the [edit services flow-collector] hierarchy level:

```
[edit services flow-collector]
destinations {
  ftp:url {
    password "password";
  }
}
```

To specify the destination FTP server, include the `ftp:url` statement. The `url` is the FTP server address for the primary flow collection destination. To specify the FTP server password, include the `password "password"` statement. The password must be enclosed in quotation marks. You can specify up to two destination FTP servers. The first destination specified is considered the primary destination.

Configure a Packet Analyzer

You can specify values for the IP address and identifier of a packet analyzer to which the flow collector interface sends traffic for analysis. The values you specify here override any default values configured elsewhere.

To configure an IP address and identifier for the packet analyzer, include the `analyzer-address` and `analyzer-id` statements at the [edit services flow-collector] hierarchy level:

```
[edit services flow-collector]
analyzer-address address;
analyzer-id name;
```

Configure File Formats

You configure data file formats, name formats, and transfer characteristics for the flow collection files. File records are sent to the destination FTP server when the timer expires or when a preset number of records are received, whichever comes first.

To configure the flow collection file format, include the following statements at the [edit services flow-collector] hierarchy level:

```
[edit services flow-collector]
file-specification {
  variant variant-number {
    data-format format;
    name-format format;
    transfer {
      record-level number;
      timeout seconds;
    }
  }
}
```

To set the data file format, include the data-format statement. To set the name file format, include the name-format statement. To set the export timer and file size thresholds, include the transfer statement and specify values for the timeout and record-level options.

Specify the name format as follows:

```
[edit services flow-collector file-specification variant variant-number]
name-format "cFlowd-py69Ni69-0-%D_%T_%l_%N.bcp.bi.gz";
```

where cFlowd-py69Ni69-0 is the static portion used verbatim, %D is the date in YYYYMMDD format, %T is the time in HHMMSS format, %l is the value of ifAlias, %N is the generation number, and bcp.bi.gz is a user-configured string.

Configure Interface Mappings

You can match an input interface with a flow collector interface and apply the preset file specifications to the input interface.

To configure an interface mapping, include the interface-map statement at the [edit services flow-collector] hierarchy level:

```
[edit services flow-collector]
interface-map {
  collector interface-name;
  file-specification variant-number;
  interface-name {
    file-specification variant-number;
    collector interface-name;
  }
}
```

To configure the default flow collector and file specifications for all input interfaces, include the file-specification and collector statements at the [edit services flow-collector interface-map] hierarchy level. To override the default settings and apply flow collector and file specifications to a specific input interface, include the file-specification and collector statements at the [edit services flow-collector interface-map *interface-name*] hierarchy level.

Configure Transfer Logs

You can configure the filename, export interval, maximum size, and destination FTP server for log files containing the transfer activity history for a flow collector interface.

To configure a transfer log, include the transfer-log statement at the [edit services flow-collector] hierarchy level:

```
[edit services flow-collector]
transfer-log {
  destinations {
    ftp:url {
      password "password";
      username username;
    }
  }
  filename "file-name";
  interval minutes;
  maximum-size number;
}
```

Include the filename, interval, maximum-size, and destinations statements.

Specify the filename as follows:

```
[edit services flow-collector transfer-log]
file-name "cFlowd-py69Ni69-0-%D_%T";
```

where cFlowd-py69Ni69-0 is the static portion used verbatim, %D is the date in YYYYMMDD format, and %T is the time in HHMMSS format.

Configure Retry Attempts

You can specify values for situations in which the flow collector interface needs more than one attempt to transfer log files to the FTP server:

Maximum number of retry attempts

Amount of time the flow collector interface waits between successive retries

To configure, include the retry and retry-delay statements at the [edit services flow-collector] hierarchy level:

```
[edit services flow-collector]
retry number;
retry-delay seconds;
```

The retry value can be from 0 through 10. The retry-delay value can be from 0 through 60 seconds.

Send cflowd Records to the Flow Collector Interface

To specify a flow collector interface as the destination for cflowd records coming from a Monitoring Services PIC, include the collector-pic statement at the [edit forwarding-options monitoring *group-name* family inet output flow-export-destination] hierarchy level:

```
[edit forwarding-options monitoring group-name family inet output
flow-export-destination]
collector-pic;
```

You can select either the flow collector interface or a cflowd server as the destination for cflowd records, but not both at the same time.

Enable Flow Collection Mode and Interface

You can select the Monitoring Services PIC to run in either flow collection mode or monitoring mode, but not both.

To set the Monitoring Services PIC to run in flow collection mode, include the flow-collector statement at the [edit chassis fpc *slot-number* pic *pic-number* monitoring-services application] hierarchy level:

```
[edit chassis fpc slot-number pic pic-number monitoring-services application]
flow-collector;
```

For further information on configuring chassis properties, see the *JUNOS Internet Software System Basics Configuration Guide*.

To specify flow collection interfaces, you configure the cp interface at the [edit interfaces] hierarchy level:

```
[edit interfaces]
cp-fpc/pic/port {
...
}
```

Example: Flow Collector Interface Configuration

Figure 6: Flow Collector Interface Topology Diagram

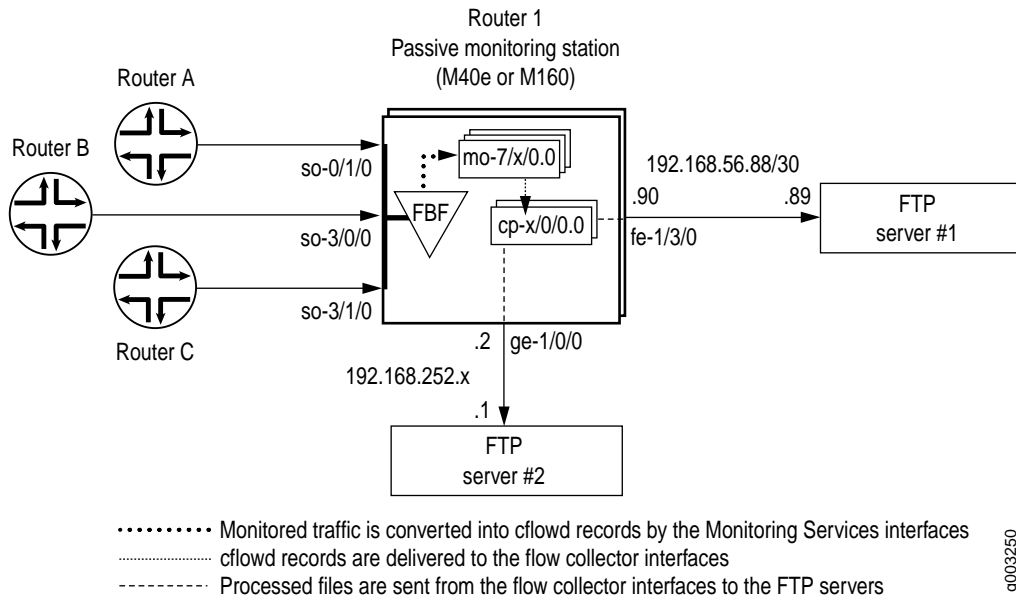


Figure 6 shows the path travelled by monitored traffic as it passes through the router. Packets arrive at input interfaces so-0/1/0, so-3/0/0, and so-3/1/0. The raw packets are directed into a filter-based forwarding routing instance and processed into cflowd records by the monitoring services interfaces mo-7/1/0, mo-7/2/0, and mo-7/3/0. The cflowd records are compressed into files at the flow collector interfaces cp-6/0/0 and cp-7/0/0 and sent to the FTP server for analysis. Finally, a mandatory class-of-service (CoS) configuration is applied to export channels 0 and 1 on the flow collector interfaces to manage the outgoing processed files.

```

Router 1 [edit]
chassis {
  fpc 6 {
    pic 0 {
      monitoring-services {
        application flow-collector; # This converts a Monitoring Services II PIC
                                   # into a flow collector interface.
      }
    }
  }
  fpc 7 {
    pic 0 {
      monitoring-services {
        application flow-collector; # This converts a Monitoring Services II PIC
                                   # into a flow collector interface.
      }
    }
  }
}

```

```

interfaces {
  cp-6/0/0 {
    unit 0 {
      # Logical interface .0 on a flow collector interface is export
      family inet {
        # channel 0 and sends records to the FTP server.
        filter {
          output cp-ftp; # Apply the CoS filter here.
        }
        address 10.0.0.1/32 {
          destination 10.0.0.2;
        }
      }
    }
    unit 1 {
      # Logical interface .1 on a flow collector interface is export
      family inet {
        # channel 1 and sends records to the FTP server.
        filter {
          output cp-ftp; # Apply the CoS filter here.
        }
        address 10.1.1.1/32 {
          destination 10.1.1.2;
        }
      }
    }
    unit 2 {
      # Logical interface .2 on a flow collector interface is the flow
      family inet {
        # receive channel that communicates with the Routing Engine.
        address 10.2.2.1/32 { # Do not apply a CoS filter on logical interface .2.
          destination 10.2.2.2;
        }
      }
    }
  }
}

```

```

cp-7/0/0 {
  unit 0 {
    # Logical interface .0 on a flow collector interface is export
    family inet {
      # channel 0 and sends records to the FTP server.
      filter {
        output cp-ftp; # Apply the CoS filter here.
      }
      address 10.3.3.1/32 {
        destination 10.3.3.2;
      }
    }
  }
  unit 1 {
    # Logical interface .1 on a flow collector interface is export
    family inet {
      # channel 1 and sends records to the FTP server.
      filter {
        output cp-ftp; # Apply the CoS filter here.
      }
      address 10.4.4.1/32 {
        destination 10.4.4.2;
      }
    }
  }
  unit 2 {
    # Logical interface .2 on a flow collector interface is the flow
    family inet {
      # receive channel that communicates with the Routing Engine.
      address 10.5.5.1/32 { # Do not apply a CoS filter on logical interface .2.
        destination 10.5.5.2;
      }
    }
  }
}
fe-1/3/0 {
  # This is the exit interface leading to the first FTP server.
  unit 0 {
    family inet {
      address 192.168.56.90/30;
    }
  }
}
ge-1/0/0 {
  # This is the exit interface leading to the second FTP server.
  unit 0 {
    family inet {
      address 192.168.252.2/24;
    }
  }
}
mo-7/1/0 { # This is the first interface that creates cflowd records.
  unit 0 {
    family inet;
  }
}
mo-7/2/0 { # This is the second interface that creates cflowd records.
  unit 0 {
    family inet;
  }
}

```

```

mo-7/3/0 { # This is the third interface that creates cflowd records.
  unit 0 {
    family inet;
  }
}
so-0/1/0 { # This is the first input interface that receives traffic to be monitored.
  encapsulation ppp;
  unit 0 {
    passive-monitor-mode; # This allows the interface to be passively monitored.
    family inet {
      filter {
        input catch;      # The filter-based forwarding filter is applied here.
      }
    }
  }
}
so-3/0/0 { # This is the second interface that receives traffic to be monitored.
  encapsulation ppp;
  unit 0 {
    passive-monitor-mode; # This allows the interface to be passively monitored.
    family inet {
      filter {
        input catch;      # The filter-based forwarding filter is applied here.
      }
    }
  }
}
so-3/1/0 { # This is the third interface that receives traffic to be monitored.
  encapsulation ppp;
  unit 0 {
    passive-monitor-mode; # This allows the interface to be passively monitored.
    family inet {
      filter {
        input catch;      # The filter-based forwarding filter is applied here.
      }
    }
  }
}
}

```

```

forwarding-options {
  monitoring group1 {                                     # Always define your monitoring group here.
    family inet {
      output {
        export-format cflowd-version-5;
        flow-active-timeout 60;
        flow-inactive-timeout 15;
        flow-export-destination collector-pic;# Sends records to the flow collector.
        interface mo-7/1/0.0 {
          source-address 192.168.252.2;
        }
        interface mo-7/2/0.0 {
          source-address 192.168.252.2;
        }
        interface mo-7/3/0.0 {
          source-address 192.168.252.2;
        }
      }
    }
  }
}
firewall {
  family inet {
    filter cp-ftp { # This filter provides CoS for flow collector interface traffic.
      term t1 {
        then forwarding-class expedited-forwarding;
      }
    }
    filter catch { # This firewall filter sends incoming traffic into the
      interface-specific; # filter-based forwarding routing instance.
      term def {
        then {
          count counter;
          routing-instance fbf_instance;
        }
      }
    }
  }
}
routing-options {
  interface-routes {
    rib-group inet common;
  }
  rib-groups {
    common {
      import-rib [inet.0 fbf_instance.inet.0];
    }
  }
  forwarding-table {
    export pplb;
  }
}

```

```

policy-options {
  policy-statement pplb {
    then {
      load-balance per-packet;
    }
  }
}
routing-instances {
  fbf_instance { # This instance sends traffic to the monitoring services interface.
    instance-type forwarding;
    routing-options {
      static {
        route 0.0.0.0/0 next-hop mo-7/1/0.0;
      }
    }
  }
}
class-of-service { # A class-of-service configuration for the flow collector interface
  interfaces { # is required for flow collector services.
    cp-6/0/0 {
      scheduler-map cp-map;
    }
    cp-7/0/0 {
      scheduler-map cp-map;
    }
  }
  scheduler-maps {
    cp-map {
      forwarding-class best-effort scheduler Q0;
      forwarding-class expedited-forwarding scheduler Q1;
      forwarding-class network-control scheduler Q3;
    }
  }
  schedulers {
    Q0 {
      transmit-rate remainder;
      buffer-size percent 90;
    }
    Q1 {
      transmit-rate percent 5;
      buffer-size percent 5;
      priority strict-high;
    }
    Q3 {
      transmit-rate percent 5;
      buffer-size percent 5;
    }
  }
}

```

```

services {
  flow-collector { # Define properties for flow collector interfaces here.
    analyzer-address 10.10.10.1; # This is the IP address of the analyzer.
    analyzer-id server1; # This helps to identify the analyzer.
    retry 3; # Maximum number of attempts by the PIC to send a file transfer log.
    retry-delay 30; # The time interval between attempts to send a file transfer log.
    destinations { # This defines the FTP servers that receive flow collector output.
      "ftp://user@192.168.56.89//tmp/collect1/" {# The primary FTP server.
        password "$9$IXJK8xN-w2oZdbZDHmF3001";# SECRET-DATA
      }
      "ftp://user@192.168.252.1//tmp/collect2/" {# The secondary FTP server.
        password "$9$elbvL7-dsgaGVwGjkP3nOBI"; # SECRET-DATA
      }
    }
  }
  file-specification {# Define sets of flow collector characteristics here.
    def-spec {
      name-format "default-allInt-0-%D_%T-%I_%N.bcp.bi.gz";
      data-format flow-compressed; # The default compressed output format.
    } # When no overrides are specified, a collector uses default transfer values.
    f1 {
      name-format "cFlowd-py69Ni69-0-%D_%T-%I_%N.bcp.bi.gz";
      data-format flow-compressed; # The default compressed output format.
      transfer timeout 1800 record-level 1000000; # Here are configured values.
    }
  }
  interface-map {# Allows you to map interfaces to flow collector interfaces.
    file-specification def-spec; # Flows generated for default traffic are sent to the
    collector cp-7/0/0; # default flow collector interface "cp-7/0/0".
    so-0/1/0.0 { # Flows generated for the so-0/1/0 interface are sent
      collector cp-6/0/0; # to cp-6/0/0, and the file-specification used is
      # "default."
    }
    so-3/0/0.0 { # Flows generated for the so-3/0/0 interface are sent
      file-specification f1; # to cp-6/0/0, and the file-specification used is "f1."
      collector cp-6/0/0;
    }
    so-3/1/0.0; # Because no settings are defined, flows generated for this
  } # interface use interface cp-7/0/0 and the default file specification.
  transfer-log-archive {# Sends flow collector interface log files to an FTP server.
    filename-prefix so_3_0_0_log;
    maximum-age 15;
    archive-sites {
      "ftp://user@192.168.56.89//tmp/transfers/" {
        password "$9$IFaEyevMXNVsWLsgaU.m6/C";
      }
    }
  }
}
}

```

