

Chapter 1

Services Interfaces Overview

Interfaces used in router networks fall into two categories:

Networking interfaces, such as Ethernet and SONET interfaces, that primarily provide traffic connectivity. For more information on these interfaces, see the *JUNOS Internet Software Configuration Guide: Network Interfaces and Class of Service* .

Services interfaces that provide specific capabilities for manipulating traffic before it is delivered to its destination.

Service PIC Types

Services interfaces enable you to add services to your network incrementally. The JUNOS software supports the following services PICs:

Adaptive Services PIC (AS PIC)—Enables you to perform multiple services on the same PIC by configuring a set of services and applications. The AS PIC offers a special range of services you configure as a service set: stateful firewalls, network address translation (NAT), and intrusion detection services (IDS). For more information about these services, see “Adaptive Services PIC Features” on page 4.



Note

To take advantage of all the capabilities available on the AS PIC, you must install it in an Enhanced FPC in an M-series router equipped with an Internet Processor II ASIC. To find out whether your router hardware is suitably equipped, you can use the `show chassis` command. For more information, see the *JUNOS Internet Software Operational Mode Command Reference: Protocols, Class of Service, Chassis, and Management* .

ES PIC—Provides a security suite for the IPv4 and IPv6 network layers. The suite provides functionality such as authentication of origin, data integrity, confidentiality, replay protection, and non-repudiation of source. It also defines mechanisms for key generation and exchange, management of security associations, and support for digital certificates. For more information about encryption interfaces, see “Configure Encryption Interfaces” on page 75.

Monitoring Services PIC—Enables you to monitor traffic flow and export the monitored traffic. Monitoring traffic allows you to perform the following tasks:

Gather and export detailed information about IPv4 traffic flows between source and destination nodes in your network.

Sample all incoming IPv4 traffic on the monitoring interface and present the data in cflowd record format.

Perform discard accounting on an incoming traffic flow.

Encrypt or tunnel outgoing cflowd records, intercepted IPv4 traffic, or both.

Direct filtered traffic to different packet analyzers and present the data in its original format.

For more information about flow monitoring interfaces, see “Configure Flow Monitoring and Discard Accounting” on page 101.

Multilink Services and Link Services PICs—Enable you to split, recombine, and sequence datagrams across multiple logical data links. The goal of multilink operation is to coordinate multiple independent links between a fixed pair of systems, providing a virtual link with greater bandwidth than any of the members. The JUNOS software supports two services PICs based on the Multilink Protocol: the Multilink Services PIC and the Link Services PIC. For more information about encryption interfaces, see “Configure Multilink and Link Services Interfaces” on page 179.

Tunnel Services PIC—By encapsulating arbitrary packets inside a transport protocol, provides a private, secure path through an otherwise public network. Tunnels connect discontinuous subnetworks and enable encryption interfaces, virtual private networks (VPNs), and Multiprotocol Label Switching (MPLS). For more information about encryption interfaces, see “Configure Tunnel Interfaces” on page 263.

Adaptive Services PIC Features

The following services are configured as a service set and are available only on the Adaptive Services PIC:

Stateful firewall—A type of firewall filter that considers state information derived from previous communications and other applications when evaluating traffic.

Network Address Translation (NAT)—A security procedure for concealing host addresses on a private network behind a pool of public addresses.

Intrusion detection services (IDS)—A set of tools for detecting, redirecting, and preventing certain kinds of network attack and intrusion.

The configuration for these three services comprises a series of rules that you can arrange in order of precedence as a *rule set*. Each rule follows the structure of a firewall filter, with a from statement containing input or match conditions and a then statement containing actions to be taken if the match conditions are met.

In addition, JUNOS software includes the following tools for configuring services:

Application protocols definition—Allows you to configure properties of application protocols that are subject to processing by router services, and group the application definitions into application sets.

Service-set definition—Allows you to configure combinations of directional rules and default settings that control the behavior of each service in the service set.

Services Configuration Flow

The general flow of services configuration is as follows:

1. Define application objects by configuring statements at the [edit applications] hierarchy level.
2. Define service rules by configuring statements at the [edit services (ids | nat | stateful-firewall) rule] hierarchy level.
3. Group the service rules by configuring the rule-set statement at the [edit services (ids | nat | stateful-firewall)] hierarchy level.
4. Group service rule sets under a service-set definition by configuring the service-set statement at the [edit services] hierarchy level.
5. Apply the service set on an interface by including the service-set statement at the [edit interfaces *interface-name* unit *logical-unit-number* family inet service (input | output)] hierarchy level. Alternatively, you can configure AS PIC logical interfaces as a next-hop destination by including the next-hop-service statement at the [edit services service-set *service-set-name*] hierarchy level.

Example: Complete Services Interfaces Configuration

The following configuration includes all the items necessary to configure services on an interface.

```
[edit]
interfaces {
  fe-0/1/0 {
    unit 0 {
      family inet {
        service {
          input {
            service-set Firewall-Set;
          }
          output {
            service-set Firewall-Set;
          }
        }
        address 10.1.3.2/24;
      }
    }
  }
  fe-0/1/1 {
    unit 0 {
      family inet {
        filter {
          input Sample;
        }
        address 112.148.1.2/24;
      }
    }
  }
}
```

```
sp-1/0/0 {
  unit 0 {
    family inet {
      address 10.1.3.2/32 {
        destination 10.1.3.50;
      }
    }
  }
}
forwarding-options {
  sampling {
    input {
      family inet {
        rate 1;
      }
    }
    output {
      cflowd 10.1.3.1 {
        port 2055;
        version 5;
      }
      flow-inactive-timeout 15;
      flow-active-timeout 60;
      interface sp-1/0/0 {
        engine-id 1;
        engine-type 136;
        source-address 10.1.3.2;
      }
    }
  }
}
firewall {
  filter Sample {
    term Sample {
      then {
        count Sample;
        sample;
        accept;
      }
    }
  }
}
```

```
services {
  stateful-firewall {
    rule Rule1 {
      match-direction input;
      term 1 {
        from {
          application-sets Applications;
        }
        then {
          accept;
        }
      }
      term accept {
        then {
          accept;
        }
      }
    }
    rule Rule2 {
      match-direction output;
      term Local {
        from {
          source-address {
            10.1.3.2/32;
          }
        }
        then {
          accept;
        }
      }
    }
  }
}
ids {
  rule Attacks {
    match-direction output;
    term Match {
      from {
        application-sets Applications;
      }
      then {
        logging {
          syslog;
        }
      }
    }
  }
}
```

```
nat {
  pool public {
    address-range low 112.148.2.1 high 112.148.2.32;
    port automatic;
  }
  rule Private-Public {
    match-direction input;
    term Translate {
      then {
        translated {
          source-pool public;
          translation-type source dynamic;
        }
      }
    }
  }
}
service-set Firewall-Set {
  stateful-firewall-rules Rule1;
  stateful-firewall-rules Rule2;
  nat-rules Private-Public;
  ids-rules Attacks;
  interface-service {
    service-interface sp-1/0/0;
  }
}
applications {
  application ICMP {
    application-protocol icmp;
  }
  application FTP {
    application-protocol ftp;
    destination-port ftp;
  }
  application-set Applications {
    application ICMP;
    application FTP;
  }
}
```