

Chapter 2

Services Interfaces Configuration Statements

This chapter shows the complete configuration statement hierarchies for configuring services interfaces. It lists all the statements that pertain to configuring services and shows their level in the configuration hierarchy. When you are configuring the JUNOS software, your current hierarchy level is shown in the banner on the line preceding the `user@host#` prompt.

For a complete list of the JUNOS configuration statements, see the *JUNOS System Basics Configuration Guide*.

This chapter is organized as follows:

[edit applications] Hierarchy Level on page 6

[edit forwarding-options] Hierarchy Level on page 6

[edit interfaces] Hierarchy Level on page 9

[edit logical-routers] Hierarchy Level on page 11

[edit services] Hierarchy Level on page 12

[edit applications] Hierarchy Level

To configure application protocols, include the following statements at the [edit applications] hierarchy level of the configuration:

```
[edit applications]
application application-name {
  application-protocol protocol-name;
  destination-port port-number;
  icmp-code value;
  icmp-type value;
  inactivity-timeout value;
  protocol type;
  rpc-program-number number;
  snmp-command command;
  source-port port-number;
  ttl-threshold value;
  uuid hex-value;
}
application-set application-set-name {
  [ application application-names ];
}
```

[edit forwarding-options] Hierarchy Level

To configure flow monitoring and accounting properties, include the following statements at the [edit forwarding-options] hierarchy level:

```
[edit forwarding-options]
accounting name {
  output {
    aggregate-export-interval seconds;
    cflowd hostname {
      aggregation {
        autonomous-system;
        destination-prefix;
        protocol-port;
        source-destination-prefix {
          caida-compliant;
        }
        source-prefix;
      }
      autonomous-system-type (origin | peer);
      port port-number;
      version format;
    }
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    interface interface-name {
      engine-id number;
      engine-type number;
      source-address address;
    }
  }
}
```

```

monitoring name;
  family inet {
    output {
      cflowd hostname port port-number;
      export-format format;
      flow-active-timeout seconds;
      flow-export-destination {
        collector-pic;
      }
      flow-inactive-timeout seconds;
      interface interface-name {
        engine-id number;
        engine-type number;
        input-interface-index number;
        output-interface-index number;
        source-address address;
      }
    }
  }
}
next-hop-group [ group-names ] {
  interface interface-name {
    next-hop [ addresses ];
  }
}
port-mirroring {
  input {
    family inet {
      rate rate;
      run-length number;
    }
  }
  output {
    interface interface-name {
      next-hop address;
    }
    no-filter-check;
  }
  traceoptions {
    file filename {
      files number;
      size bytes;
      (world-readable | no-world-readable);
    }
  }
}
sampling {
  disable;
  input {
    family inet {
      max-packets-per-second number;
      rate number;
      run-length number;
    }
  }
  output {
    aggregate-export-interval seconds;
  }
}

```

```

cflowd hostname {
  aggregation {
    autonomous-system;
    destination-prefix;
    protocol-port;
    source-destination-prefix {
      caida-compliant;
    }
    source-prefix;
  }
  autonomous-system-type (origin | peer);
  (local-dump | no-local-dump);
  port port-number;
  source-address address;
  version format;
}
file {
  disable;
  filename filename;
  files number;
  size bytes;
  (stamp | no-stamp);
  (world-readable | no-world-readable);
}
flow-active-timeout seconds;
flow-inactive-timeout seconds;
interface interface-name {
  engine-id number;
  engine-type number;
  source-address address;
}
}
traceroptions {
  file filename {
    files number;
    size bytes;
    (world-readable | no-world-readable);
  }
}
}
}

```



NOTE: For the complete [edit forwarding-options] hierarchy, see the *JUNOS Policy Framework Configuration Guide*. This listing includes only the statements used in flow monitoring and accounting services.

[edit interfaces] Hierarchy Level

To configure services interfaces, include the following statements at the [edit interfaces] hierarchy level of the configuration. The statements can also be configured at the [edit logical-routers *logical-router-name*] hierarchy level.

```
[edit interfaces]
interface-name {
  atm-options {
    mpls {
      pop-all-labels {
        required-depth number;
      }
    }
  }
  encapsulation type;
  mlfr-uni-nni-bundle-options {
    acknowledge-retries number;
    acknowledge-timer milliseconds;
    action-red-differential-delay (disable-tx | remove-link);
    drop-timeout milliseconds;
    fragment-threshold bytes;
    hello-timer milliseconds;
    lmi-type (ansi | itu);
    minimum-links number;
    mrru bytes;
    n391 number;
    n392 number;
    n393 number;
    red-differential-delay milliseconds;
    t391 number;
    t392 number;
    yellow-differential-delay milliseconds;
    encapsulation type;
  }
  unit logical-unit-number {
    clear-dont-fragment-bit;
    compression {
      rtp {
        f-max-period number;
        queues [ queue-numbers ];
        port {
          minimum port-number;
          maximum port-number;
        }
      }
    }
  }
  dial-options {
    l2tp-interface-id name;
    (dedicated | shared);
  }
}
```

```

family family {
  accounting {
    destination-class-usage;
    source-class-usage direction;
  }
  address address {
    destination address;
  }
  bundle (ml-fpc/pic/port | ls-fpc/pic/port | vsp-fpc/pic/port);
  ipsec-sa ipsec-sa;
  receive-options-packets;
  receive-ttl-exceeded;
  sampling direction;
  service {
    input {
      [ service-set service-set-names <service-filter filter-name> ];
      post-service-filter filter-name;
    }
    output {
      [ service-set service-set-names <service-filter filter-name> ];
    }
  }
}
passive-monitor-mode;
peer-unit unit-number;
service-domain (inside | outside);
tunnel {
  backup-destination address;
  destination destination-address;
  key number;
  routing-instance {
    destination routing-instance-name;
  }
  source-address address;
  ttl number;
}
dlci dlci-identifier;
drop-timeout milliseconds;
encapsulation type;
fragment-threshold bytes;
interleave-fragments {
  queues [ queue-numbers ];
}
minimum-links number;
mrru bytes;
multicast-dlci dlci-identifier;
short-sequence;
}
multiservice-options {
  boot-command filename;
  (core-dump | no-core-dump);
  (syslog | no-syslog);
}

```

```

services-options {
  inactivity-timeout seconds;
  open-timeout seconds;
  syslog {
    host hostname {
      services severity-level;
      facility-override facility-name;
      log-prefix prefix-number;
    }
  }
}
so-fpc/pic/port {
  unit logical-unit-number {
    passive-monitor-mode;
  }
}

```



NOTE: For the complete [edit interfaces] hierarchy, see the *JUNOS Network Interfaces and Class of Service Configuration Guide*. This listing includes only the statements used in configuring services.

[edit logical-routers] Hierarchy Level

The following lists the statements that can be configured at the [edit logical-routers] hierarchy level that are documented in this manual. For more information about logical routers, see the *JUNOS Routing Protocols Configuration Guide*.

```

[edit logical-routers]
logical-router-name {
  interfaces interface-name {
    interface-configuration;
  }
}

```

[edit services] Hierarchy Level

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

```
[edit services]
adaptive-services-pics {
  traceoptions {
    flag flag;
  }
}
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;
}
}
```

```

ids {
  rule rule-name {
    match-direction (input | output | input-output);
    term term-name {
      from {
        applications [ application-names ];
        application-sets [ set-names ];
        destination-address address;
        source-address address;
      }
      then {
        aggregation {
          destination-prefix prefix-value;
          source-prefix prefix-value;
        }
        (force-entry | ignore-entry);
        logging {
          syslog;
          threshold rate;
        }
        syn-cookie {
          mss value;
          threshold rate;
        }
      }
    }
  }
  rule-set rule-set-name {
    [ rule rule-names ];
  }
}
ipsec-vpn {
  ike {
    proposal proposal-name {
      authentication-algorithm (md5 | sha1);
      authentication-method (dsa-signatures | pre-shared-keys | rsa-signatures);
      description description;
      dh-group (group1 | group2);
      encryption-algorithm (3des-cbc | des-cbc);
      lifetime-seconds seconds;
    }
    policy policy-name {
      description description;
      local-id {
        ipv4_addr [ values ];
        key_id [ values ];
      }
      mode (aggressive | main);
      pre-shared-key (ascii-text key | hexadecimal key);
      proposals [ proposal-names ];
      remote-id {
        ipv4_addr [ values ];
        key_id [ values ];
      }
    }
  }
}

```

```

ipsec {
  proposal proposal-name {
    authentication-algorithm (hmac-md5-96 | hmac-sha1-96);
    description description;
    encryption-algorithm (3des-cbc | des-cbc);
    lifetime-seconds seconds;
    protocol (ah | esp | bundle);
  }
  policy policy-name {
    description description;
    perfect-forward-secrecy {
      keys (group1 | group2);
    }
    proposals [ proposal-names ];
  }
}
rule rule-name {
  match-direction (input | output);
  term term-name {
    from {
      destination-address address;
      source-address address;
    }
    then {
      backup-remote-gateway address;
      clear-dont-fragment-bit;
      dynamic {
        ike-policy policy-name;
        ipsec-policy policy-name;
      }
      manual {
        direction (inbound | outbound | bidirectional) {
          authentication {
            algorithm (hmac-md5-96 | hmac-sha1-96);
            key (ascii-text key | hexadecimal key);
          }
          auxiliary-spi spi-value;
          encryption {
            algorithm (des-cbc | 3des-cbc);
            key (ascii-text key | hexadecimal key);
          }
          protocol (ah | bundle | esp);
          spi spi-value;
        }
      }
      no-anti-replay;
      remote-gateway address;
      syslog;
    }
  }
}
rule-set rule-set-name {
  [ rule rule-names ];
}
}

```

```

l2tp {
  tunnel-group group-name {
    hello-interval seconds;
    hide-avps;
    l2tp-access-profile profile-name;
    local-gateway address address;
    maximum-send-window packets;
    ppp-access-profile profile-name;
    receive-window packets;
    retransmit-interval seconds;
    service-interface interface-name;
    syslog {
      host hostname {
        services severity-level;
        facility-override facility-name;
        log-prefix prefix-number;
      }
    }
    tunnel-timeout seconds;
  }
  traceoptions {
    debug-level level;
    filter {
      protocol name;
    }
    flag flag;
    interfaces interface-name {
      debug-level level;
      flag flag;
    }
  }
}
nat {
  pool nat-pool-name {
    address (address | address-range low minimum-value high maximum-value);
    port (automatic | range low minimum-value high maximum-value);
  }
  rule rule-name {
    match-direction (input | output);
    term term-name {
      from {
        applications [ application-names ];
        application-sets [ set-names ];
        destination-address address;
        source-address address;
      }
      then {
        translated {
          destination-pool nat-pool-name;
          source-pool nat-pool-name;
          translation-type (destination type | source type);
        }
        syslog;
      }
    }
  }
}

```

```

    rule-set rule-set-name {
        [ rule rule-names ];
    }
}
service-set service-set-name {
    ([ ids-rules rule-names ] | ids-rule-sets rule-set-name);
    ([ ipsec-vpn-rules rule-names ] | ipsec-vpn-rule-sets rule-set-name);
    ([ nat-rules rule-names ] | nat-rule-sets rule-set-name);
    ([ stateful-firewall-rules rule-names ] | stateful-firewall-rule-sets rule-set-name);
    interface-service {
        service-interface interface-name;
    }
    ipsec-vpn-options {
        local-gateway address;
    }
    next-hop-service {
        inside-service-interface name.number;
        outside-service-interface name.number;
    }
    syslog {
        host hostname {
            services severity-level;
            facility-override facility-name;
            log-prefix prefix-number;
        }
    }
}
}
adaptive-services-pics {
    traceoptions {
        flag flag;
    }
}
stateful-firewall {
    rule rule-name {
        match-direction (input | output | input-output);
        term term-name {
            from {
                applications [ application-names ];
                application-sets [ set-names ];
                destination-address address;
                source-address address;
            }
            then {
                (accept | discard | reject);
                allow-ip-option [ values ];
                syslog;
            }
        }
    }
    rule-set rule-set-name {
        [ rule rule-names ];
    }
}
}

```