

Classifying Interfaces (SRC CLI)

Use the following configuration statements to define interface classification scripts:

```
shared network device name interface-classifier rule name {  
    target target;  
}
```

```
shared network device name interface-classifier rule name condition name ...
```

```
shared network device name interface-classifier rule name script {  
    script-value;  
    include include;  
}
```

A classification script can contain either a target and a condition or a script. If you do not define a script, the classifier must have both a target and a condition.

To define interface classification scripts:

1. From configuration mode, enter the interface classifier configuration for a device.

```
user@host# edit shared network device erx-node1 interface-classifier
```

2. Create a rule for the classifier. You can create multiple rules for the classifier.

```
[edit shared network device erx-node1 interface-classifier]  
user@host# edit rule rule-3
```

3. Configure either a target or a script for the rule.

- Configure the target for the rule.

```
[edit shared network device erx-node1 interface-classifier rule rule-3]  
user@host# set target target
```

If you configured a target for the rule, you must configure a match condition for the rule. You can create multiple conditions for the rule. See “Interface Classification Conditions” on page 3.

```
[edit shared network device erx-node1 interface-classifier rule rule-3]  
user@host# set condition name
```

- Configure the script for the rule.

```
[edit shared network device erx-node1 interface-classifier rule rule-3]  
user@host# edit script
```

(Optional) You can specify a script target.

```
[edit shared network device erx-node1 interface-classifier rule rule-3 script]  
user@host# set script-value
```

(Optional) You can include a script that has already been created.

```
[edit shared network device erx-node1 interface-classifier rule rule-3 script]
user@host# set include include
```

where *include* is a reference to an existing script that is included in the script you are configuring.

4. (Optional) Change the order of rules.

```
[edit shared network device erx-node1 interface-classifier]
user@host# insert rule rule-5 before rule-4
```

5. (Optional) Rename a rule.

```
[edit shared network device erx-node1 interface-classifier]
user@host# rename rule rule-5 to DHCP
```

6. (Optional) Verify the classifier rule configuration.

```
[edit shared network device erx-node1 interface-classifier rule rule-3]
user@host# show
target /sample/junose/PPP-special;
condition {
    "pppLoginName=\"*@special.com\"";
}
```

7. (Optional) Verify the interface classifier configuration.

```
[edit shared network device erx-node1 interface-classifier]
user@host# show
rule rule-1 {
    script "
# Use the following syntax:
#
# descr-file ::= [script] section*
# section   ::= ('[' type ']' nl conditions) | ('[*]' nl script)
# type      ::= 'a-zA-Z0-9-_*'
# nl        ::= '\\n'
# conditions ::= ((('#'|';') comment) |
#                 (['&'|'|'] field-name ( '='|'=='|'!=') match) nl)*
# field-name ::= member of InterfaceObject
# match      ::= UNIX style filename matching
# script     ::= regular python script, defined functions need to be
#                 included in the list \"classify\"
#
# the section-names correspond to a PolicyList object below
# o=Policies, o=umc:
# [name] => DN: \"policyGroupName=name, o=Policies, o=umc\"
#
# Use one of the following \"field names\":
#   pppLoginName      - set to \"user@realm\", if interface is PPP
#   interfaceName     - name of the ERX Interface in CLI syntax
#   virtualRouterName - name of the VR the interface is connected to
";
}
```

```

rule rule-2 {
    script "
    # apply different default policies for PPP subscribers in realm
    \"special.com\"
    def log(obj):
        from net.juniper.smgmt.sae import Main
        icc = Main.theComponentRegistry.get(\"icc.component\")
        if icc is None:
            Main.theComponentRegistry.put(\"icc.component\", [])
        else:
            icc.append(obj)
    classify.append(log)
    ";
}
rule rule-3 {
    target /sample/junose/PPP-special;
    condition {
        "pppLoginName=\"*@special.com\"";
    }
}
rule rule-4 {
    target /sample/junose/PPP;
    condition {
        "pppLoginName!=\"\"";
    }
}
rule rule-5 {
    target /sample/junose/DHCP;
    condition {
        "interfaceName=\"fastEthernet*\"";
        "interfaceName=\"atm*/*. *\"";
    }
}

```

- Related Topics**
- Classifying Interfaces (C-Web Interface)
 - Reloading Interface Classification Scripts
 - Example: Managing Specific Interfaces
 - Example: Managing Interfaces by Using the Interface Description
 - Overview of Configuring Classification Scripts

Interface Classification Conditions

Use the fields in this section to define interface classification conditions.

broadcastAddr

- Interface broadcast address.
- Value—Valid broadcast address format
- Example—broadcastAddr.hostAddress = “ 255.255.255.255”

ifAlias

- Description of an interface.
- Value—Interface description that is configured on the router. For JUNOSe routers, it is the description configured with the **interface description** command.
- Example—ifAlias = “ 1st pppoe int”

ifDesc

- Alternate name of the interface that is used by SNMP. This name is a system-generated name.
- Value
 - On a JUNOSe router, the format of the description is
`ip<slot>/<port>.<subinterface>`
 - On the JUNOS routing platform, ifDesc is the same as interfaceName.
- Example—ifDesc = “ IP3/1.1 ”

interfaceName

- Name of the interface.
- Value
 - Name of the interface in your router CLI syntax
 - FORWARDING_INTERFACE for routing instance (used by traffic mirroring)
- Example—For JUNOSe routers: interfaceName = “ fastethernet6/0.1”

For JUNOS routing platforms: interfaceName = “ fe-0/1/0.0”

For forwarding interface: interfaceName = “ FORWARDING_INTERFACE”

ipAddress

- Interface IP address.
- Value—Valid IPv4 IP address format
- Example—ipAddress = “ 10.10.30.1”

ipMask

- Interface network mask.
- Value—Valid IPv4 IP network mask format
- Example—ipMask = “ 255.255.255.255”

mtu

- Maximum transfer unit configured on the interface.
- Value—32-integer value
- Example—mtu = “ 1492”

nasPortId

- Port identifier of an interface.
- Value—Includes interface name and additional layer 2 information
- Example—nasPortId = “ fastEthernet 3/1 ” (There is a space between fastEthernet and slot number 3/1 in the nasPortId.)

pppLoginName

- Login name for PPP subscribers.
- Value—Login name in the format username@domain
- Example—pppLoginName = “ pebbles@virneo.net”

radiusClass

- RADIUS class attribute.
- Value—RADIUS class name
- Example—radiusClass = “ Premium”

serviceBundle

- Content of the vendor-specific RADIUS attribute for the service bundle.
- Value—Name of a service bundle

userIpAddress

- Subscriber IP address (PPP only).
- Value—valid IPv4 address
- Example—userIpAddress = “ 192.168.30.15”

virtualRouterName

- Name of the virtual router or routing instance.
- Value—For JUNOS routers: name of the virtual router in the format vrname@hostname
For JUNOS routing platforms: name of the routing instance
- Example—virtualRouterName = “ default@erx5”