128 TECHNOLOGY

ROUTING FILTERS AND POLICIES

April 2, 2019

ROUTING FILTERS AND POLICIES

Routing filters and policies allow you to refine and match routes and traffic and apply rules & actions to achieve Policy based Routing. They are handy in configuring route maps for Routing Protocols like BGP and OSPF.
The routing filters and Policies elements are global data. They are available authority-wide and can be applied to individual routers' Routing Instances.

ROUTING FILTERS:

A routing filter is a rule or set of rules that will return a result of accept or reject. The main function of a route filter is to *match* traffic (objects). The rules specified under a filter element define how to process the objects passing the filter. For each rule, based on the filter:

- Accept: Indicates elements matching the rule should not be filtered by the calling construct.
- Reject: Indicates elements matching the rule should be filtered by the calling construct.

You can find the Routing Filters configuration element under the Authority > Authority wide Routing Filters and Policies.
On a PCLI, configure > authority > routing > filter.

ROUTING FILTER TYPES:

There are primarily 4 types of filters available:

- **prefix-filter:** based on prefixes within a given range. Allows you to give a match criterion.
- **as-path-filter:** based on AS path of a BGP route.
- **community-filter:** Based on BGP community string value
- **extended-community- filter:** Based on extended community value for BGP routes

Based on the filter type, the fields for the filter rules are generated. E.g. when *prefix-filter* is selected it generates the required fields of *prefix, greater than or equal, less than or equal*, etc. as opposed to the *community* field generated for *community-filter* type.

ROUTING POLICIES:

A routing policy is the action or set of actions you want to be made to routes that meet the conditions of your filter. A policy consists of a set of statements that are executed in a sequence. A statement is executed by first running the *conditions*. If all the conditions *match* or if no conditions are specified, the *policy* (accept or reject) is checked. An *accept* means to execute the *actions* in the statement and then terminate the policy returning accept. A *reject* means not to execute the actions and terminate the policy returning reject.
On a PCLI, configure > authority > routing > policy

Note: You need to mention an explicit ***accept_all*** statement to allow all other traffic to pass.

CONDITION TYPES:

The condition type is configured under the *Conditions* element. Based on the *condition type*, *filters* of matching type can be selected. E.g. an *address-prefix-filter-condition* type allows you to select a *filter* of the type *prefix-filter*. Hence, the filter type must be selected according to the Routing Policy Condition Type. Following are the various types of condition available:

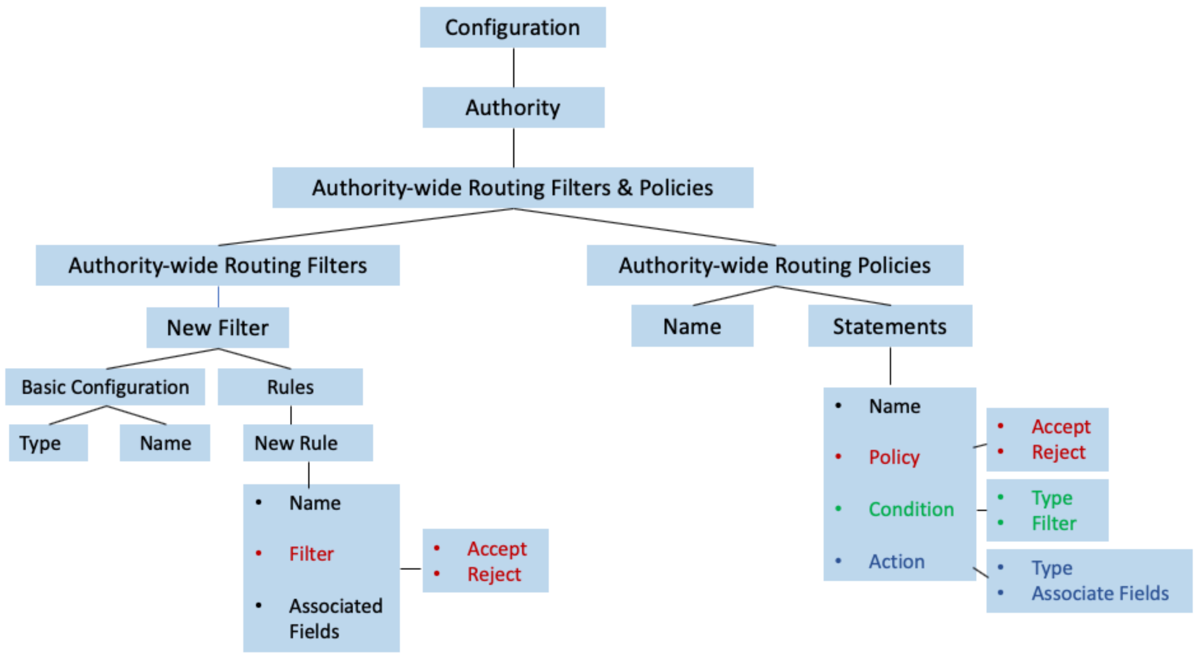
- **address-prefix-filter-condition:** A prefix filter condition on address. Filter-type is *prefix-filter.*
- **next-hop-prefix-filter-condition:** A prefix filter condition on next hop. Filter-type is *prefix-filter*.
- **source-prefix-filter-condition:** A prefix filter condition on route source. Filter-type is *prefix-filter.*
- **as-path-filter-condition:** An autonomous path filter condition. Filter-type is *as-path-filter.*
- **community-filter-condition:** A community filter condition. Filter-type is *community-filter.*
- **extended-community-filter-condition:** An extended community filter condition*.* Filter-type is *extended-community-filter.*
- **next-hop-interface-condition:** A next hop interface condition. *Next-hop-node* field is the name of the node the next hop interface resides on. The *next-hop-interface* name of the next hop interface to match on.
- **metric-condition:** A metric condition. Specify the metric value to match on.
- **origin-condition:** An origin condition. Specify the BGP origin to match on
- **peer-condition:** A peer condition. Specify the peer-address-type as
    - peer-address: The peer address to match
    - peer-local: Match local addresses (static or redistributed routes)
- **probability-condition:** A probability condition. Specify the probability to match.
- **tag-condition:** A tag condition. Specify the tag to match.

ACTION TYPES:

An action is executed when all the conditions are satisfied, and the policy is of the type *accept*. Following are the types of actions:

- **set-aggregator:** An action which sets the BGP aggregator. Specify the aggregator AS and the aggregator IP address.
- **modify-as-path**: An action which changes the BGP AS-path. Specify the AS(s) to exclude from the AS-path in the *exclude* option and the AS(s) to prepend to the AS-path in the *prepend* option.
- **remove-community:** An action which removes the BGP community attribute. Specify the community filter to use to remove matching communities.
- **set-community:** An action which sets the BGP community attribute. Specify the set-community-type as:
    - community-attribute-case: Mention The new community attribute values in *community-attribute.*
    - none-case
- **set-extended-community:** An action which sets the BGP extended community attribute. Specify the following:
    - route-target: add new extended-community route target value to the list.
    - site-of-origin: add the new extended-community site of origin value to the list.
- **set-next-hop:** An action which sets the next hop. The next-hop could be
    - ip-address: The new next hop IP address to set
    - peer-address: Set the next hop to the IP address of the peer

- **set-local-preference:** An action which sets the BGP local preference. Specify the local preference value
- **modify-metric:** An action which sets the metric. You can perform the following metric-based actions
    - set: The metric value
    - add: The metric value to add
    - subtract: The metric value to subtract
- **set-originator-id:** An action which sets the originator ID. Specify the new originator ID to set.
- **set-origin:** An action which sets the origin. Specify the BGP origin value to be set.
- **set-bgp-weight:** An action which sets the BGP weight. Specify the BGP weight value to be set.
- **set-tag:** An action which sets the tag. Specify the tag value.
- **continue:** A flow action that advances to the next (or specified) entry in the policy
- **call:** A flow action calls the given policy If this policy returns reject then the current policy will terminate and return reject.



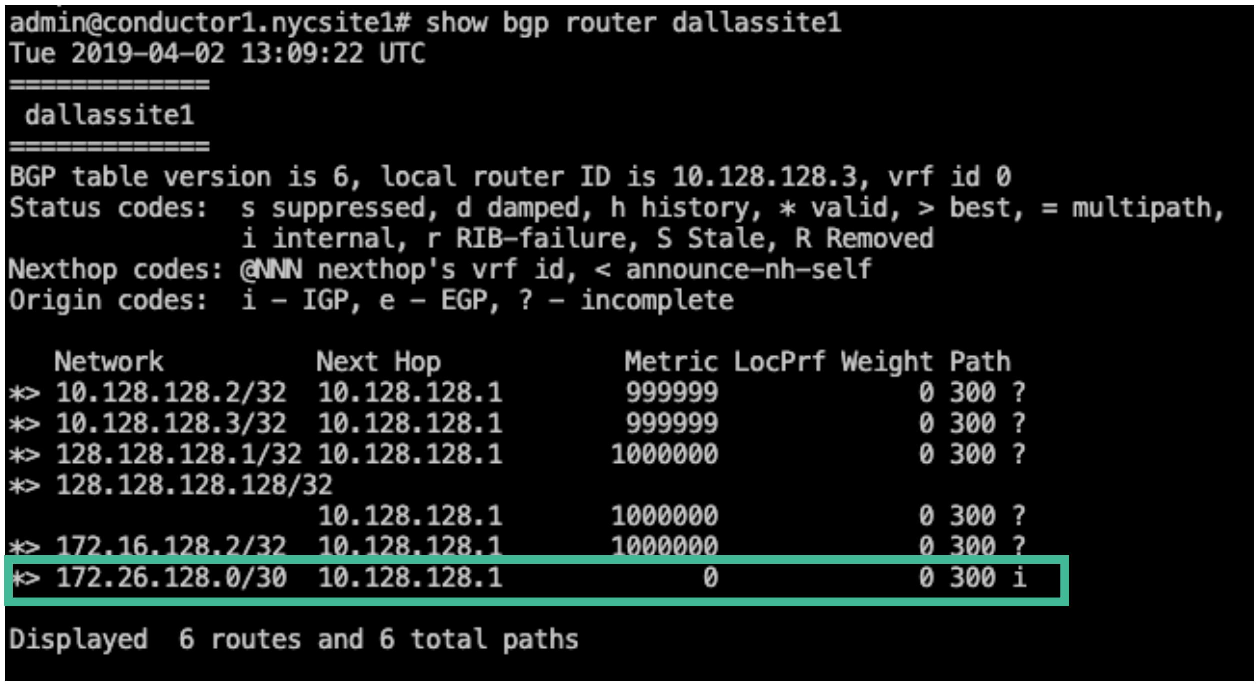
SAMPLE CONFIGURATION:

```
config

    authority
        name            CompanyX

        remote-login

        exit

         routing

            filter  match-private-addresses
                type  prefix-filter
                name  match-private-addresses

                rule  match-private-addresses
                    name    match-private-addresses
                    filter  accept
                    prefix  2.2.2.0/24
                exit
            exit

            filter  exclude
                type  prefix-filter
                name  exclude

                rule  exclude
                    name    exclude
                    prefix  192.168.7.0/24
                exit
            exit

            filter  match_boston
                type  prefix-filter
                name  match_boston

                rule  match_boston
                    name    match_boston
                    prefix  172.26.128.0/30
                exit
            exit

            filter  match_exclude
                type  community-filter
                name  match_exclude

                rule  match_exclude
                    name       match_exclude
                    community  128:128
                exit
            exit
```

```
 policy  add-as-path
    name       add-as-path

    statement  add-as-path
        name       add-as-path
        policy     accept

        condition  address-prefix-filter-condition
            type           address-prefix-filter-condition
            prefix-filter  match-private-addresses
        exit

        action     modify-as-path
            type     modify-as-path
            prepend  "100 100 100 100"
        exit
    exit

    statement  accept_all
        name    accept_all
        policy  accept
    exit
exit

policy  exclude
    name       exclude

    statement  exclude
        name       exclude
        policy     reject

        condition  address-prefix-filter-condition
            type           address-prefix-filter-condition
            prefix-filter  exclude
        exit
    exit

    statement  accept_all
        name  accept_all
    exit
exit

policy  match_boston
    name       match_boston

    statement  match_boston
        name       match_boston

        condition  address-prefix-filter-condition
            type           address-prefix-filter-condition
            prefix-filter  match_boston
        exit

        action     set-community
            type                 set-community
            community-attribute  128:128
        exit
    exit

    statement  accept_all
        name  accept_all
    exit
exit
```

```
                policy  match_exclude
                    name       match_exclude

                    statement  match_exclude
                        name       match_exclude
                        policy     reject

                        condition  community-filter-condition
                            type              community-filter-condition
                            community-filter  match_exclude
                        exit
                    exit

                    statement  accept_all
                        name  accept_all
                    exit
                exit
            exit
        exit
exit
```

These routing policies can be applied to routing protocols such as BGP and OSPF. In the example below, we are advertising route 172.26.128.0/30 from bostonsite1 to dallassite1. Router ID for bostonsite1 is 10.128.128.1.



```
admin@conductor1.nycsite1# show bgp router dallassite1
Tue 2019-04-02 13:09:22 UTC
=============
 dallassite1
=============
BGP table version is 6, local router ID is 10.128.128.3, vrf id 0
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.128.128.2/32  10.128.128.1        999999             0 300 ?
*> 10.128.128.3/32  10.128.128.1        999999             0 300 ?
*> 128.128.128.1/32 10.128.128.1       1000000             0 300 ?
*> 128.128.128.128/32
                    10.128.128.1       1000000             0 300 ?
*> 172.16.128.2/32  10.128.128.1       1000000             0 300 ?
*> 172.26.128.0/30  10.128.128.1             0             0 300 i

Displayed  6 routes and 6 total paths
```

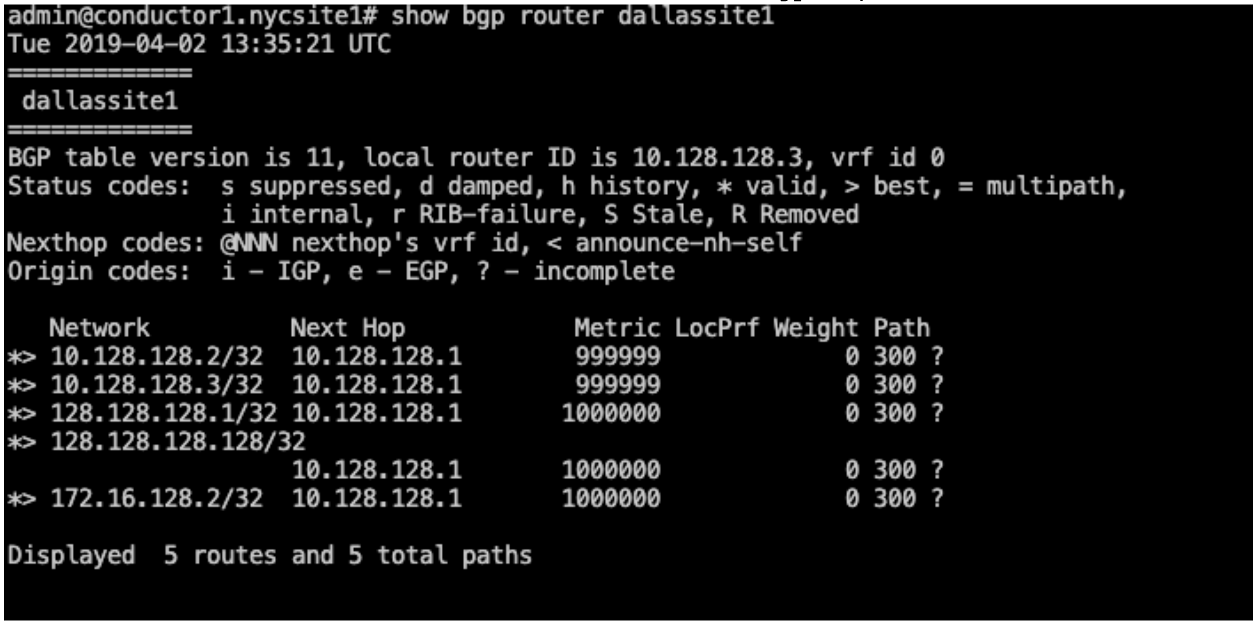
We will now add a community string (128:128)to this route being advertised from bostonsite1.

```
admin@datacenter1.bostonsite1# config authority router bostonsite1 routing default-
instance
admin@datacenter1.bostonsite1 (routing[type=default-instance])# routing-protocol bgp
admin@datacenter1.bostonsite1 (routing-protocol[type=bgp])# address-family ipv4-
unicast
admin@datacenter1.bostonsite1 (address-family[afi-safi=ipv4-unicast])# show
afi-safi  ipv4-unicast

network   172.26.128.0/30
    network-address  172.26.128.0/30
    policy           match_boston
exit
```

Now, adding policy on dallassite1 to reject routes with the community string 128:128.

```
admin@branchoffice2.dallassite1# config auth router dallassite1 routing default-
instance
admin@branchoffice2.dallassite1 (routing[type=default-instance])# routing-protocol
bgp
admin@branchoffice2.dallassite1 (routing-protocol[type=bgp])# neighbor 10.128.128.1
admin@branchoffice2.dallassite1 (neighbor[neighbor-address=10.128.128.1])# show
neighbor-address  10.128.128.1
neighbor-as       300

transport

    local-address
        routing-interface  bgp-int-dallas
    exit
exit

multihop
    ttl  5
exit

neighbor-policy
    inbound-policy  match_exclude
exit
```

Notice that we do not see route 172.26.128.0/30 in the `show bgp` output of dallassite1:



```
admin@conductor1.nycsite1# show bgp router dallassite1
Tue 2019-04-02 13:35:21 UTC
=============
 dallassite1
=============
BGP table version is 11, local router ID is 10.128.128.3, vrf id 0
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.128.128.2/32  10.128.128.1        999999             0 300 ?
*> 10.128.128.3/32  10.128.128.1        999999             0 300 ?
*> 128.128.128.1/32 10.128.128.1       1000000             0 300 ?
*> 128.128.128.128/32
                    10.128.128.1       1000000             0 300 ?
*> 172.16.128.2/32  10.128.128.1       1000000             0 300 ?

Displayed  5 routes and 5 total paths
```