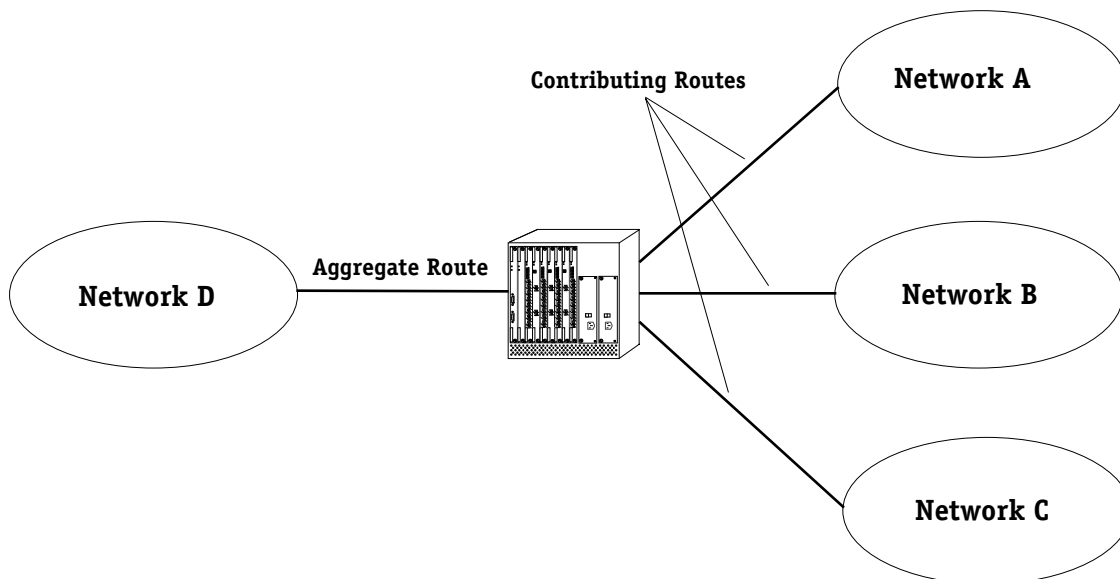


11 Route Aggregation

Route aggregation is a method of generating a more general route (given the presence of one or more specific routes). By combining multiple subnet addresses into a single address, the amount of routing information advertised from the network is reduced. By carefully allocating network addresses to clients, regional networks can announce a single route to a particular destination instead of hundreds.

When an aggregate statement is specified in the configuration file, a destination address is set along with specific criteria for which contributing routes are included in the aggregate statement.



In this diagram, the Network D is only aware of the created aggregate route, rather than the paths directly to Networks A, B, and C. Aggregate routes are not used by the originators of the aggregate route (in this case, Networks A, B, C). The receiver employs the route to send packets back to the aggregate route source. Aggregate routes must be explicitly configured. No aggregation is performed unless specified in the aggregate statement of the configuration file.

A slight variation of aggregation is the generation of a route based on the existence of certain conditions. This is sometimes known as the *route of last resort*. This route inherits the next hop and AS path from the contributing route with the lowest preference. The most common usage for the generate statement is to create a default route based on the presence of a route from a peer on a neighboring backbone.

A route may only contribute to an aggregate route which is more general than itself, and any given route may only contribute to one aggregate route. An aggregate route may contribute to a more general aggregate.

Specifying Preferences

In all route aggregation statements, one of two keywords is used to control how routes compete with other protocols:

preference <i>preference</i>	Specifies the preference value used when comparing this route to other routes from other protocols. The route with the lowest preference available becomes the active route, is installed in the forwarding table, and is eligible to be exported to other protocols. The default preferences are automatically configured by the individual protocols.
restrict	Indicates that these routes are not to be considered as contributors to the routing table. The specified protocol may be any of the protocols supported by GateD.

In some cases, the routes filtered by import statements are not installed in the routing table. In other cases, the routes are installed with a negative preference. This prevents them from becoming active so they will not be installed in the forwarding table or exported to other protocols.

The Aggregate and Generate Statements

The following are the complete aggregation and generation statements. Two variations of each statement are shown, one for each of the preference options (*route_filter* and **restrict**):

```

aggregate
default | ( network [ ( mask mask ) | ( masklen number ) ] ) [ preference preference ] [
brief ] {
    proto [ all | direct | static | kernel | aggregate | proto ]
    [ ( as autonomous system ) | ( tag tag ) | ( aspath aspath_regexp ) ] restrict ;

    proto [ all | direct | static | kernel | aggregate | proto ]
    [ ( as autonomous system ) | ( tag tag ) | ( aspath aspath_regexp )
    [ preference preference ] {
        route_filter [ restrict | ( preference preference ) ] ;
    } ;
};

generate
default | ( network [ ( mask mask ) | ( masklen number ) ] ) [ preference preference ] {
    proto [ all | direct | static | kernel | aggregate | proto ]
    [ ( as autonomous system ) | ( tag tag ) | ( aspath aspath_regexp ) ] restrict ;

    proto [ all | direct | static | kernel | aggregate | proto ]
    [ ( as autonomous system ) | ( tag tag ) | ( aspath aspath_regexp )
    [ preference preference ] {
        route_filter [ restrict | ( preference preference ) ] ;
    } ;
};

```

Routes that match the criteria specified in the statement (and that become part of the aggregate route) are called *contributing routes*. They are ordered according to the aggregation preference that applies to them. (Route filters are described below.)

If there is more than one contributing route with the same aggregating preference, the contributing route's own preferences are used to order the routes. The preference of the aggregate route will be that of contributing route with the lowest aggregate preference.

The syntax used in the aggregate and generate statement is the same as the syntax described in Chapter 1 of this manual.

default

Specifies that a default route should be generated (network 0.0.0.0 and mask 0.0.0.0).

(*network* [(**mask** *mask*) | (**masklen** *number*)])

This statement allows you to specify the destination address of the aggregate route with a network address, or a network address with a mask or a mask length number.

preference *preference*

This command specifies the preference to assign to the resulting aggregate route. The default preference is 130.

brief

This command is used to specify that the AS path should be truncated to the longest common AS path. The default is to build an AS path consisting of all contributing AS paths.

proto [**all** | **direct** | **static** | **kernel** | **aggregate** | *proto*]

This command allows you to choose what type of route should be included in the aggregate route. In addition to the special protocols listed, the contributing protocol may be chosen from among any of the ones supported by GateD.

as *autonomous_system*

This command restricts the selection of routes to those learned from the specified autonomous system.

tag *tag*

This command restricts the selection of routes to those with the specified tag. Tags are specified in the GateD protocol statements.

aspath *aspath_regexp*

This command restricts the selection of routes to those that match the specified AS path.

restrict

This command, when appended to the aggregate criteria, indicates that the matching routes are *not* to be considered as contributors of the specified aggregate. The specified protocol may be any of the protocols supported by GateD.

route_filter

When no route filtering is specified, all routes from the specified source will match that statement. If any filters are specified, only routes that match the specified filters will be considered as contributors.

See Chapter 10 of this manual for specifics on creating a route filter.

preference *preference*

Specifies the preference to assign to contributing routes that match the aggregate or generate criteria. The default preference is 130.

Aggregate and Generate Statement Example

The following is an example of a route aggregation statement:

```
aggregate 192.0.0.0 mask 255.0.0.0 preference 20 brief {  
    proto all { all; } :  
    proto all as 5 restrict ;  
};  
  
generate 0.0.0.0 masklen 0 {  
    proto direct {  
        172.28.5.0 mask 255.255.255.0 preference 20;  
    };  
    proto direct {  
        80.0.0.0 masklen 8 preference 10;  
    };  
};
```

In the above example, the aggregate statement specifies:

- An aggregate route to network 192.0.0.0 with a network mask of 255.0.0.0 is created.
- This route has a preference of 20, and the AS path will be truncated to the longest common AS path in the contributing routes.
- All protocols (**direct**, **static**, **kernel**, **aggregate**, **OSPF**, **OSPFASE**, **RIP**, and **BGP**) are included in the aggregate route.
- All protocols from AS 5 are restricted and not considered part of the aggregate path.

The generate statement specifies:

- A default route (**0.0.0.0 mask 0.0.0.0**) is generated.
- Only directly connected routes 172.28.5.0 and 80.0.0.0 can contribute to the generated route.
- The generated route will have a preference of 20 or 10, depending on which router (either 172.28.5.0 or 80.0.0.0) contributes to the generated route.

