

# 5 GateD Router Interfaces

A GateD interface is a configured software connection between a router and one of its attached networks. The connection allows a router to communicate with other networks. Usually, this connection is made with a defined Internet Protocol (IP) address. In Alcatel switches, this is done using virtual router ports.

Interfaces must be specified in the **gated.conf** file. Each protocol will use configured interfaces to send and receive data. Interfaces are assigned to specific protocols (RIP, OSPF, or BGP) by using the protocol statements of the configuration file. This allows for specific ports to pass one type of protocol. The interfaces specified in the configuration file must correspond to actual addresses in the switch.

## Interfaces and Switch Router Ports

For example, if a switch has a group with a virtual router port using an IP address of 1.1.1.1, then an interface of 1.1.1.1 can be specified in the **gated.conf** file.

### ◆ Note ◆

IP addresses must be configured on the switch before GateD can be initialized. For information on creating groups, virtual router ports, and assigning IP addresses, see the chapter titled “Managing Groups and Ports” in your switch manual.

Interfaces for a protocol are specified using the **interface** token within the protocol statement. Using the above example, a **gated.conf** file could look similar to the following:

```
rip on {  
    interface 1.1.1.1;  
};
```

The above file would mean that RIP is operating on router interface 1.1.1.1 of the switch, which would correspond to a configured IP address in the switch.

More than one interface can be specified at a time. For example, if RIP needed to be active on interfaces 1.1.1.1 and 1.1.1.2, this could be done by adding more **interface** tokens, as shown:

```
rip yes {  
    interface 1.1.1.1 version 2;  
    interface 1.1.1.2 version 2;  
};
```

The above file would mean that RIP is operating on interfaces 1.1.1.1 and 1.1.1.2 of the switch.

---

Another possible option is to specify a protocol for all interfaces encountered on the switch. This can be done using the **all** modifier for the **interface** token, as shown:

```
rip on {  
    interface all ;  
};
```

In this instance, all interfaces (configured virtual router IP addresses) encountered by GateD on this switch will use RIP as their routing protocol.

## Protocol Specific Interfaces

Anytime the creation of a route is specified, an interface can also be specified. This includes the routing protocols as well as other routing functions such as static routes, importing and exporting routes, and aggregating routes. If no interface is specified, the first interface encountered by the GateD software becomes the default interface.

For specific information on configuring interfaces for protocols and routing functions, see the following sections:

- For RIP and RIP version 2, see Chapter 6.
- For OSPF, see Chapter 7.
- For BGP, see Chapter 8.
- For static routes, local routes, router discovery, and kernel modifications, see Chapter 9.
- For importing and exporting routes and route filtering, see Chapter 10.
- For route aggregation, see Chapter 11.

## Global Interfaces Commands

Besides specifying interfaces for protocols to use, you can also modify the behavior of an interface, or group of interfaces, using the interface statement group in the configuration file. The interface statement group appears second in the configuration file order, immediately following the options statement group.

Details on configuring the interface statement are covered below in *The Interface Statement* on page 5-3.

# The Interface Statement

The following is the full interface statement. These statements are entered in the **gated.conf** file. Each command in the statement is described in the following sections.

```

interfaces {
  options
    [ strictinterfaces ]
    [ scaninterval time ]
  ;
  interface interface_list
    [ preference preference ]
    [ down preference preference ]
    [ passive ]
    [ AS autonomoussystem ]
  ;
  define address
    [ broadcast address ] | [ pointtopoint address ]
    [ netmask mask ]
    [ multicast ]
  ;
};

```

Interface statements must come directly after the option statement, but are not required when using GateD. There are three sections to the interface statement: **options**, **interface**, and **define**. These sections are covered below.

The syntax conventions used in the interface statement are described in Chapter 3 of this manual.

### Options Statements

The options section of the interface statement allows you to set global interface behavior. The statements included as options commands are as follows:

```
options
    [ strictinterfaces ]
    [ scaninterval time ]
;
```

Each command is described below.

#### **strictinterfaces**

This command, when set, indicates that it is a fatal error to reference an interface in the configuration file that is not present when GateD is started and not listed in a define statement. If GateD starts up and reads an interface in the **gated.conf** file that it cannot find in the network, it will shut down. Without this option set, a warning message will be issued for unknown interface, but GateD will not shut down.

#### **scaninterval** *time*

This command specifies how often GateD scans the interface list for changes. The default is every 15 seconds.

## Interface Statements

The interface statement sets interface options on the specific interfaces. The commands included as interface statements are as follows:

```
interface interface_list
  [ preference preference ]
  [ down preference preference ]
  [ passive ]
  [ AS autonomous system ]
```

When using this set of commands, you must first specify the interface address by entering the interface token, followed by an interface address, as shown:

```
interface 1.1.1.1
```

Any further use of the interface commands would only affect interface 1.1.1.1. For example, if you added a preference value to the statement, as shown:

```
interface 1.1.1.1
  preference 30
```

a preference of 30 would be attached to interface 1.1.1.1, and *only* interface 1.1.1.1.

It is also possible to specify more than one interface address, or all of them. To configure more than one interface, enter the **interface** token multiple times, as shown:

```
rip yes {
  interface 1.1.1.1 preference 30
  interface 1.1.1.2 preference 20
};
```

To configure all interfaces, enter **all** after the **interface** token, as shown:

```
interface all preference 10
```

It is not always advisable to use the **all** modifier, as it increases the memory usage for the routing table.

Using **all** rather than specifying interfaces has a second consequence. If an interface is configured to run a routing protocol and GateD does not see routing information on that interface, then routing on that interface will be disabled. When this occurs, the preference for the interface changes to 120 (low priority and used for all interfaces that are down), the switch will “poison” routes to the network attached to the interface, GateD will ultimately stop advertising routing updates to the interface completely, and exclude the interface from advertisements to other interfaces.

The **all** specification may seem like an easy way to enable RIP, but it may also lead to the disabling of interfaces that do not participate in routing. It is better to explicitly define all interfaces that will use a protocol.

Commands available for this statement are described below:

**preference** *preference*

This command sets the preference for routes to this interface when it is operational. The default preference is **0**. For information on preference, see Chapter 4 of this manual.

**down preference** *preference*

If routing information is not received from an interface for an extended period of time, GateD assumes the interface is down. This command sets the preference for routes to this interface when GateD does not believe it to be functioning properly. The default value is **120**. For information on preference, see Chapter 4 of this manual.

**passive**

If routing information is not received from an interface for an extended period of time, GateD assumes the interface is down. This command prevents GateD from changing the preference of the route to this interface if it is not believed to be functioning properly due to lack of received routing information. GateD will only perform this check if the interface is actively participating in a routing protocol.

**AS** *autonomoussystem*

When a route traverses several autonomous systems, it records the AS identifiers in a list that is called an AS path. This command allows you to specify a number as the identifier to be used for the AS path list.

## Define Statements

The define statement specifies interfaces that might not be present when GateD is started. When using the **strictinterfaces** token (described in *Options Statements* on page 5-4), any interface referenced in the configuration file that is not detected in the actual network causes a fatal error that shuts down GateD. Using the define statement to mark interface addresses prevents this from occurring. The commands included as define statements are as follows:

```
define address
    [broadcast address] | [pointtopoint address]
    [netmask mask]
    [multicast]
```

When using this set of commands, you must first specify the interface address by entering the define token, followed by an interface address, as shown:

```
define 1.1.1.1
```

In this example, any further use of the define commands would only affect interface 1.1.1.1.

Commands available for this statement are described below:

**broadcast** *address*

This command sets the specified define interface as broadcast capable (e.g., Ethernet or Token Ring) and specifies the broadcast address.

**pointtopoint** *address*

This command sets the specified define interface as a point-to-point interface (e.g., SLIP or PPP) and specifies the address on the local side. The first address on the define statement references the address of the host on the remote end of the interface; the address specified after the **pointtopoint** keyword defines the address on the local side of the interface.

An interface not defined as broadcast or point-to-point is assumed to be non-broadcast multi-access (NBMA), such as an X.25 network.

**netmask** *mask*

This command sets the subnetmask to be used on the specified define interface. This is ignored on point-to-point interfaces.

**multicast**

This command specifies that the define interface is multicast capable.

# Interface Statement Example

The following is an example of an interface statement, with an explanation of the characteristics of the individual statements:

```
interfaces {  
    options strictinterfaces;  
    interface 1.1.1.1 preference 90 down preference 200 passive;  
    define 1.1.1.5 multicast;  
};
```

In the above interface statement:

- The **interface** token is listed to note the beginning of the interface statement.
- The **strictinterfaces** token specifies that only detectable network interfaces are allowed to be referenced in the **gated.conf** file. If an interface is listed in the file that is not detected, GateD will shut down.
- Interface 1.1.1.1 is specified as having a route preference of 90, and a down preference of 200. The **passive** token states that even if no routing updates are heard from this interface address, it should not be labeled as down.
- The **define** token for interface 1.1.1.5 specifies that when GateD starts up, if interface 1.1.1.5 is *not* detected, GateD will keep running. This is an exception to the **strictinterfaces** option referenced above. The **multicast** token denotes interface 1.1.1.5 as multicast capable.