

## Chapter 10

# Configuration Initialization and Re-initialization

### 10.1 Overview

When GateD is started or re-initialized (with the HUP signal), it goes through the following series of events.

### 10.2 Phase 1 - Initialization of Process

At startup or reinit time, GateD attempts to find the state of the kernel routing table and the configuration of the machine's interfaces.

**Note:** In order to do the former, GateD must be running as root.

The process goes through the following sequence:

#### 10.2.1 Reading the Kernel Routing Table

The reading of the kernel routing table is only done once to find the initial state of the table. After it has been read, GateD listens for changes via the routing socket, kmem, or ioctl interfaces. Which one GateD uses depends on the operating system. (See "Chapter 18 Kernel Interface" on page 95 for more information.)

#### 10.2.2 Reading the Kernel Interface List

At startup and reinit time, as well as periodically during operation, GateD will scan the list of interfaces. It does this using one of several methods, depending on the operating system. Any time the interface list is scanned, the entire list is read and changes are reported to the protocols individually.

### 10.3 Phase 2 - Initialization of Tasks

Each task has a callback hook associated with initialization, pre-parse initialization, and policy initialization, and interface changes among other things. At startup, each protocol's callback is called in this order:

1. **var init** - invokes the protocol's `var_init()` procedure
2. **parse** - parses configuration file
3. **init** - initializes the protocol with parser information
4. **reinit** - invokes the task's reinit procedure
5. **if\_notify** - notifies tasks of changes of all physical and logical interfaces
6. **reinit finalize** - calls the task's `reinit_finalize` procedures
7. **new policy** - configured policy is processed. The task's `newpolicy` procedures are invoked.

Some protocols defer initialization of protocol interface structures until the first Interface Change notification, and at that time, they are added or deleted according to the configuration of the machine and the information in the config file. See "Chapter 9, Interacting with GateD" of *Operating GateD* for more information about how GateD interacts with system administration actions.

## 10.4 Phase 3 - Re-initialization of Tasks

GateD can be re-initialized while running without disturbing the operation of existing protocol sessions. Reconfiguration can enable instances of protocols that were previously disabled and can disable protocols that are operating at the time of re-initialization. GateD rereads its configuration file upon receiving a SIGHUP signal and processes the changes between its current configuration and that represented in the new instance of the configuration file.

Upon receipt of a SIGHUP signal, GateD calls each running task's cleanup function to give each task an indication that a re-initialization is taking place. A task's cleanup routine basically saves parts of its current configuration state and frees up other parts, such as policy, that can safely be rebuilt without knowing its previous state. Then the seven initialization steps listed above are performed again. As part of re-initialization, all routes are run through both import and export policy again because the policy may have changed. Routes that were imported before the re-initialization may no longer be importable under the new policy, and routes that were unimportable under the old policy may now be importable. The same holds true for the exportation of routes. A change to the importation of routes requires flashing the changes to GateD's route table to all protocols.