

Chapter 5

Directive Statements

Directive statements provide direction to the GateD configuration language parser about included files and the directories in which these files reside. Directive statements are immediately acted upon by the parser. Other statements terminate with a semicolon (";"), but directive statements terminate with a new line. The two directive statements are:

%directory "directory"

%directory defines the directory in which the included files are stored. When **%directory** is used, GateD looks in the directory identified by the path name for any included files that do not have a fully qualified filename (for example, do not begin with "/"). **%directory** does not actually change the current directory; it just specifies the prefix applied to the included file names. Refer to "%directory" on page 7 in the *GateD Command Reference Guide* for additional information.

%include "filename"

%include identifies an include file. The content of the file is included in the `gated.conf` file at the point in the `gated.conf` file where the **%include** directive is encountered. If the filename is not fully qualified, (for example, it does not begin with "/"), the file is considered to be relative to the directory defined in the last **%directory** directive. The **%include** directive statement causes the specified file to be parsed completely before resuming with this file. Nesting of up to 10 levels is supported. The maximum nesting level may be increased by changing the definition of **FI_MAX** in `parse.h`. Refer to "%include" on page 8 in the *GateD Command Reference Guide* for additional information.

In a complex environment, segmenting a large configuration into smaller, more easily understood segments might be helpful. One of the great advantages of GateD, however, is that it combines the configuration of several different routing protocols into a single file. Segmenting a small file unnecessarily complicates routing configurations.

