CHAPTER 20

Control Plane and System Interaction

Control Plane Interaction

At a conceptual level, packetC capabilities can be categorized as data plane capabilities, which involve examining, changing, and routing packets and control plane capabilities, which involve displaying system-level values and messages. In some systems, the two planes may be on separate boards or on distant components in a far-flung system. In others, such as a PC-based emulation system, the two sets of capabilities might execute on the same hardware. Thus, packetC specifies control plane capabilities in a general way, while allowing considerable latitude in how they are implemented.

The defined control plane capabilities are:

- Alerts
- Information Logging
- Messages

The packetC language specification defines the above capabilities in terms of language features and constructs but does not proscribe additional control plane capabilities. Such capabilities will typically be implemented by using compiler pragmas associated with the control pragma category.

Alerts and Information Logging

packetC provides alert and log commands to facilitate passing messages to the control plane environment and to facilitate logging packet data. The alert command simply sends a message while the log command sends associated packet data. Each of these commands has an associated message group and identification number associated with the action. The values used are the current values within the sys data structure at the time of the command call.

Alert command examples are shown below:

```c
sys.messageGroup = MSG_CRITICAL;
sys.messageId = TCP_OPTION_MSG;
alert;
```
sys.messageGroup = MSG_CRITICAL;
sys.messageId = MSG_BOX_OVERHEATING;
alert;

Log messages are used to send packet data to the control plane. Logging may include a portion of the current packet or the entire packet. Similar to the alert command, a message group and identification is associated with the operation.

Log command examples are shown below:

sys.messageGroup = MSG_CRITICAL;
sys.messageId = 2;
log pkt; // log the entire packet

sys.messageGroup = MSG_MAJOR;
sys.messageId = 21;
log pkt[ 0 : 63 ]; // log first 64 bytes of the packet

alert Statement

The packetC statements consist of alphanumeric keywords, rather than new operator symbols or overloaded operator symbols.

The alert statement sends a text string message to the Control Plane environment, where it may be displayed or otherwise processed (see the section on the Messages to the Control Plane in this chapter). Two system variables located in the global sys variable control the default behavior of alert. sys.messageId is an index of the message to use and sys.messageGroup is an enumeration value that indicates the message group, both of which depend on the definitions from the cloudshield.ph include file shown below in the section on cloudshield.ph file in this chapter.

enum int t_msgGroup = {
    MSG_CRITICAL = 1, // initial default
    MSG_MAJOR,
    MSG_MINOR,
    MSG_WARNING,
    MSG_INFO
};

When an alert statement appears, default values set in the sys variable are used. The user can manipulate those defaults by setting the following values in the sys struct (see Chapter 18).

sys.messageId = 5;
sys.messageGroup = MSG_INFO;

Here are some examples of the alert statement being used.

const int TCP_OPTION_MESSAGE = 12;
...

sys.messageGroup = MSG_CRITICAL;
sys.messageId = TCP_OPTION_MESSAGE;
alert;