13.2 syslogd Daemon

Unix systems normally start a daemon named `syslogd` from one of the system initializations scripts, and it runs as long as the system is up. Berkeley-derived implementations of `syslogd` perform the following actions on startup:

1. The configuration file, normally `/etc/syslog.conf`, is read, specifying what to do with each type of log message that the daemon can receive. These messages can be appended to a file (a special case of which is the file `/dev/console`, which writes the message to the console), written to a specific user (if that user is logged in), or forwarded to the `syslogd` daemon on another host.
2. A Unix domain socket is created and bound to the pathname `/var/run/log` (`/dev/log` on some systems).
3. A UDP socket is created and bound to port 514 (the `syslog` service).
4. The pathname `/dev/klog` is opened. Any error messages from within the kernel appear as input on this device.

The `syslogd` daemon runs in an infinite loop that calls `select`, waiting for any one of its three descriptors (from Steps 2, 3, and 4) to be readable; it reads the log message and does what the configuration file says to do with that message. If the daemon receives the `SIGHUP` signal, it rereads its configuration file.

We could send log messages to the `syslogd` daemon from our daemons by creating a Unix domain datagram socket and sending our messages to the pathname that the daemon has bound, but an easier interface is the `syslog` function that we will describe in the next section. Alternately, we could create a UDP socket and send our log messages to the loopback address and port 514.

> Newer implementations disable the creation of the UDP socket, unless specified by the administrator, as allowing anyone to send UDP datagrams to this port opens the system up to denial-of-service attacks, where someone could fill up the filesystem (e.g., by filling up log files) or cause log messages to be dropped (e.g., by overflowing `syslog's` socket receive buffer).
>
> Differences exist between the various implementations of `syslogd`. For example, Unix domain sockets are used by Berkeley-derived implementations, but System V implementations use a `STREAMS` log driver. Different Berkeley-derived implementations use different pathnames for the Unix domain socket. We can ignore all these details if we use the `syslog` function.