Syslog

Log a message to the system.

```
openlog("ProgName",options,facility);
syslog(LOG_NOTICE,"Out of Disk");
```

`openlog`: called once per program
Establishes syslog defaults.
Name of program.
options–such as include the pid with the message.
facility–type of log

`syslog`: called for each message to be logged
Sends a message to `syslogd`.
`LOG_NOTICE`–the log level
message–to be recorded.

```
openlog("mail",LOG_PID,LOG_MAIL);
syslog(LOG_EMERG,"Failed");
```

`"mail"` our name for logging purposes
`LOG_PID` –include process PID in log
`mail` –which log to record it in
`LOG_EMERG` –log level
`Failed` –message to be logged

`logger -p mail.emer "Failed"`   (Script call)

syslog.conf

`syslogd`–gets the message, handles it as defined by the configuration file `/etc/syslog.conf`.

`syslog.conf` format: selector — action

Selectors:
`*.emerg` —all at `LOG_EMERG` or higher.
`mail.*` — all levels of info from mail
`news,lpr.err` —all `news` or `lpr` at `LOG_ERR` or higher.
`*.=debug` – only `LOG_DEBUG` (not "and higher")
`*.!debug` – lower than `LOG_DEBUG`
`*.=debug,news.none` – all debug, except news

Actions:
`sam` —if sam is logged in, display it on his terminal
`/var/log/cron` —put it into this file.
`@aardvark.cecs.csulb.edu` —send it to this machine.

Examples:

```
mail.*              /var/log/maillog
*.notice            root
kern.emerg          /dev/console
cron.err            @aardvark.cecs.csulb.edu
```

Syslog will create log files, it will not create directories, do that by hand.

syslog startup

`syslogd -r` — enables remote machines to report log entries

`-h` — if you received a remote log entry you are allowed to forward it.

Synchronization

The unix file system allows buffering.
If a write has been requested, the write will be performed when convenient.
This is more efficient interms of disk access.

`syslog` traditionally does not use buffering. You may tell it to do so by adding a minus sign in front of an entry in `syslog.conf`. For example:

```
mail.*            -/var/log/maillog
```

Down side: if it's an error message about what is causing the machine to crash, it probably won't get written before the machine crashes.
At a minimum, do not use the minus for levels `alert` or `emerg`, since these are often the last message before some sort of a crash.