Logging

Keep a record of what ocurred during the execution of a shell script with a logging mechanism. Logs can store any type of infromation you want, but they typically answer who, what, when, where, and why something ocurred.

Linux uses syslog standard for message logging. This allows programms and apps to generate messages that can be captured, processed and stored by the system logger. It elminates the need for each and every app having to implement a logging mechanism. That means we can take advantage of this logging system in our shell scripts.

Syslog standard uses facilities and severities to categorize messaages. Each msg is labeled with a facility code and a severity level. Combination of facilities and severities can be used to determine how a message is handled.

Facilities are used to indicate what type of program or what part of the system the message originated from. Messages that are labeled wit h the **kern** facility originate from the Linux kernel. Messages that are labeled with the **mail** facility come from applications involved in handling mail.

There are several facilities. If your script is involved in handling mail you could use the user facilityu. Also, the facilities ranging from local0 to local7 are to be used to create custom logs. These facilities would also be appropiate for custom written shell scripts.

```
Number Keyword Description
0   kern    kernel messages
1   user    user level messages
2   mail    mail system
3   daemon  styystem daemons
4   auth    security/authorization messages
5   syslog  messages generated by syslogd
6   lpr     line printer subsystem
7   news    networks news subsystem
8   uucp    uucp subsystem
9   clock   daemon
10  authpriv
11  ftp
12  ntp
```

The severities are emergency, alert, critical, error, warning, notice, info, and debug.

```
Code Severity Keyword Description
0   Emergency   emerg(panic)    System is Unusable
1   Alert       alert           Action must be taken inmediately
2   Critical    crit            Critical conditions
3   Error       error(err)      Error conditions
4   Warning     warning(warn)   Warning conditions
5   Notice      notice          Normal but significant condition
```

```
6    Info         info             Informational messages
7    Debug        debug            Debug level messages
```

Each Linux distro uses a slightly sifferent set of defaults, and logging rules are configurable and can be changed.

Many messages are stored in `/var/log/messages` or `/var/log/syslog`.

Logger

Logger command generate syslog messages. In its simplest form you simply supply a message to the logger utility. By derfault, the logger utility creates messages using the user facility and the notice severity

The message generated without options includes date, user and message.

```
logger "Message"
logger -p local0.info "Message" # Uses Local0 Facility
logger -t myscript -p local0.info "Message" # Tag Message
logger -i -t myscript "Message" # Process ID (PID)
logger -s -p local0.info "Message" # -s: Displayed on Screen
```

Create a function in shell script to handle logging

```
logit()
{
    local LOG_LEVEL=$1
    shift
    MSG=$@
    TIMESTAMP=$(date +"%Y-%m-%d %T")
    if [ $LOG_LEVEL = 'ERROR' ] || $VERBOSE
    then
        echo "${TIMESTAMP} ${HOST} ${PROGRAM_NAME} [${PID}]: ${$LOG_LEVEL} ${MSG}"
}
```

- Logit expects that log level followed by a message passed into it.
- `shift` command is run to shift the positional parameters to the left.
- If the log level is error or the VERBOSE global variable is set to true, a message is echoed to the sccreen, which includes info such timestamp, log level and the message.