```
        ROOT_SECRET : myrootfile
    The string "ROOT_SECRET" in file /root/.secret will be replaced with an MD5 hash
    of the concatenation of the lab instance seed and "myrootfile".

CLONE_REPLACE : <filename> : <symbol> : <ignored>
  Replace a symbol with the clone instance number of a container per the CLONE option
  in the start.config file.  This is intended for use in providing instance-
  unique values on cloned containers, e.g., to assign a unique password to
  each container based on the clone number.  If the container has no clone
  instance number then the symbol is replaced with an empty string.
```

The parameter_id fields may be referenced during the automated grading function, described below in section 6.3.

## 5.2 Synchronizing startup and parameterization

System initialization should generally occur as with any Linux based system, e.g., using rc.local or system services. You can enable rc.local by placing `RUN systemctl enable rc-local` in the Dockerfile. Parameterizing occurs subsequent to container "boot", but prior to running the fixlocal.sh script. The Ubuntu based images include a `waitparam.service` that delays reporting its initialization to systemd until parameterization has completed. That service is configured to run prior to rc.local. The service unit file is at:

```
trunk/scripts/designer/system/lib/systemd/system
```

If you have defined system services that should not start until parameterization has occurred, then add this to the `[Unit]` section of their service unit file:

```
After=waitparam.service
```

Note that if your `fixlocal.sh` script starts any such service, you must create a flag directory from within your `fixlocal.sh` script to unblock the waitparam.service. The following lines would achieve that:

```
PERMLOCKDIR=/var/labtainer/did_param
sudo mkdir -p "$PERMLOCKDIR"
```

## 5.3 Parameterizing start.config

Parameterizing of the start.config file occurs prior to Docker container creation. The framework modifies a copy of the file stored in `/tmp/start.config` and uses that when assigning attributes to containers, e.g., IP addresses. Currently only IP addresses within the start.config can be parameteterized (e.g., not user names).

## 5.4 Simple Parameterization for Checking Own-work

The simplest, though by no means robust, strategy for ensuring students have turned in their own work, (vice getting a zip file from a friend and simply changing the name of the file), is to individualize some file on one of the containers, and then check that file and the archive file names during grading. The framework does this automatically and reports on any student archive that does not seem to have originated from a Labtainer initiated with that student's email address.