## QTemp Controller Container

### Description

This is a custom-built container based on Ubuntu 20.04 incorporating only Python modules for external connectivity rather than Ubuntu client applications such as MySQL or MQTT Mosquito.

The primary program is configured to run as a non-privileged user within the container to minimize the impact on the underlying OS if the service is compromised.

**Note:** Minimal networking tools were also included to support connectivity troubleshooting since a default Ubuntu container does not have even basic network testing, such as the ability to ping another server on the network. These packages can be removed if a smaller container is required without impacting the controller function.

### Software build list

|  |  |
| --- | --- |
| File Name | Function |
| ssat3\_controllerClientV2\_2.Dockerfile | Build instructions for Docker |
| .env\_controller | Run time parameter file to be used during container startup |
| controller\_requirements.txt | Text file listing modules that must be installed during container build |
| controllerClientv2\_2.py | Primary program interacting with Monitoring and actuator nodes using MQTT broker as the intermediary communication mechanism and managing user state through database connections. |
| appDb.py | A customized python module to decouple database interactions from primary control program logic |
| appEnc.py | A customized python module to decouple encryption key management and encryption functions from the primary program control logic. |
| manode\_keypairs.csv | Symmetric encryption keys loaded on application startup |

### Container Build Instructions:

* Copy all files from the build list to a clean directory
* Use the docker build command to create a new image in the server's local Docker repository. Include a version tag to support better version control within the environment. ( note "." at the end of the command, it is needed to include all files within the directory in the build process.
  + sudo docker build -t ssat3\_controller-main:2.0 -f ssat3\_controllerClientV2\_0.Dockerfile.

**Example of build process staging and initial build commands:**

Text

Description automatically generated

The build process can take a few minutes, look for the successfully built message and confirm the image name is correct.

Text

Description automatically generated

An environment file is needed to generate environment variables within the container's operating system that will be used by the program to determine IP addresses of other network services, access credentials and so forth. This makes the container both reusable and allows for simple recovery if a core password is compromised; I.E., change the password on the remote service and restart the container with the updated environment file containing the new credentials.

sudo docker run -h controller-main --name controller-main --env-file .env\_controller -v /opt/storage/conf/priv:/opt/storage/conf/priv --add-host=mqbroker.qtemp.local:10.100.200.3 --add-host=appdb.qtemp.local:10.100.200.3 -d -t ssat3\_controller-main:2.0

<https://docs.docker.com/engine/reference/commandline/run/>

(Note MQTT user ID and password must be in place on MQTT broker for the controller application prior to running the container)

sudo docker run -h controller-main --name controller-main-22 --env-file .env\_controller -v /opt/storage/conf/priv:/opt/storage/conf/priv -v /opt/storage/logs:/opt/storage/logs --add-host=mqbroker.qtemp.local:10.100.200.3 --add-host=appdb.qtemp.local:10.100.200.3 -it ssat3\_controller-main:2.2.1

Once the docker run command completes the first time, it can be stopped and started as needed. The controller program prints a good deal of data to standard out; this can be viewed using the docker attach command. (see screenshot below)

Text

Description automatically generated with medium confidence