

# UsingRunit

[runit](#) provides service management, and can be used to run Cassandra.

Create a `/service/cassandra/run` with the following:

```
#!/bin/bash

cassandra_home=/opt/cassandra/current
cassandra_bin=${cassandra_home}/build/classes
CASSANDRA_CONF=${cassandra_home}/conf
CASSANDRA_USER=daemon
CASSANDRA_MEM=4g
CLASSPATH=${CASSANDRA_CONF}:${cassandra_bin}

for jar in ${cassandra_home}/lib/*.jar; do
    CLASSPATH=${CLASSPATH}:${jar}
done

ulimit -n 65535
cd var

exec chpst -u${CASSANDRA_USER} -e /service/cassandra/env -o 65535 \
/usr/bin/java \
    -ea \
    -Xms${CASSANDRA_MEM} \
    -Xmx${CASSANDRA_MEM} \
    -XX:TargetSurvivorRatio=90 \
    -XX:+AggressiveOpts \
    -XX:+UseParNewGC \
    -XX:+UseConcMarkSweepGC \
    -XX:+CMSParallelRemarkEnabled \
    -XX:+HeapDumpOnOutOfMemoryError \
    -XX:SurvivorRatio=128 \
    -XX:MaxTenuringThreshold=0 \
    -Dcom.sun.management.jmxremote.port=8080 \
    -Dcom.sun.management.jmxremote.ssl=false \
    -Dcom.sun.management.jmxremote.authenticate=false \
    -Dcassandra -Dstorage-config=${CASSANDRA_CONF} \
    -Dcassandra-foreground=yes \
    -cp ${CLASSPATH} \
    org.apache.cassandra.thrift.CassandraDaemon 2>&1
```

Assuming you setup the rest of your runit paths, you can now start cassandra with `'svc start /service/cassandra'`.