

# Test jobs and developing tests

## Files

All files related to testing should be placed inside the related project source in the testing directory.

Test scripts file names should all use the template test-<project>.sh and be lower case.

### Files when using run\_tests()

When using run\_tests() all those tests should be placed inside a directory named test.d inside the testing directory.

Test scripts file names when using run\_tests() must be called test.sh and placed in a directory inside the test.d and that directory should have an appropriated name regarding the test:

```
testing/test-project.sh
testing/test.d/00-simple-test/test.sh
testing/test.d/10-simple-test2/test.sh
```

## Test example / templates

Here are some test examples and/or templates.

### Test without run\_tests()

Here is a simple test that runs a command and checks the output if it was successfully.

```
#!/usr/bin/env bash
source `dirname "$0"`/lib.sh && init || exit 1

check_if_tested softhsm && exit 0
start_test softhsm

test_ok=0
(
    log_this softhsm-init-token softhsm --init-token --slot 0 --label OpenDNSSEC --pin 1234 --so-pin 1234 &&
    log_grep softhsm-init-token stdout "The token has been initialized."
) &&
test_ok=1

if [ "$test_ok" -eq 1 ]; then
    log_cleanup
    set_test_ok softhsm || exit 1
    exit 0
fi

exit 1
```

### Tests with run\_tests()

Using run\_tests() is good if you have many tests for a project. This example gives the tests additionally functions to use that are directly related to the project.

### Main test-<project.sh>

```

#!/usr/bin/env bash
source `dirname "$0"`/lib.sh && init || exit 1

ods_reset_env ()
{
    echo "ods_reset_env: resetting opendsnsec environment"
    echo "y" | ods-ksmutil setup &&
    log_this softhsm-init-token softhsm --init-token --slot 0 --label OpenDNSSEC --pin 1234 --so-pin 1234 ||
    return 1

    if ! log_grep softhsm-init-token stdout "The token has been initialized."; then
        return 1
    fi
}

require opendsnsec

check_if_tested opendsnsec && exit 0
start_test opendsnsec

test_ok=0
(
    run_tests test.d
) &&
test_ok=1

if [ "$test_ok" -eq 1 ]; then
    set_test_ok opendsnsec || exit 1
    exit 0
fi

exit 1

```

## Simple test.d/00-start-stop/test.sh

This simple test starts and stop the program and checks the syslog for the appropriated messages that the start and stop was successful.

```

#!/usr/bin/env bash

ods_reset_env &&
log_this ods-control-start ods-control start &&
syslog_waitfor 60 'ods-signerd: \[engine\] signer started' &&
syslog_grep 'ods-enforcerd: Sleeping for' &&
log_this ods-control-stop ods-control stop &&
syslog_waitfor 60 'ods-signerd: \[engine\] signer shutdown' &&
syslog_grep 'ods-enforcerd: all done'

```