TF-A Release Process & TF-A v2.3 release
Agenda

- Understanding the TF-A release process & Testing flow
  - Pre-release activities & code freeze
  - Talk through various testing, test plan updates, change log review and finalization, readme, version fixes, release assessments, post release

- Categories of tests
  - Talk through the categories of tests run (daily, automated, manual and OoB)

- TF-A v2.3 features

- Future improvements planned
  - Automations, open CI
  - Partner feedback request on code freeze and RC candidate testing
TF-A Release Process

Pre-Release
- Guidance on the release schedule can be found on project documentation [1]
- Announcing the code freeze date 2 weeks in advance in tf-a@lists.trustedfirmware.org
  - all patches to be included in the release to be merged ahead of the code freeze date

Code Freeze
- Creating the release candidates & communicating in tf-a@lists.trustedfirmware.org
  - Enables all including partners to test their platforms against the RC

Release Testing
- Test categories run
- If there are failures, debug and apply patches for bug fix, review and merge
- Complete all testing, retag all repositories to next RC and re-run all tests till clean tag is established.

Other release activities
- Internal Test plan updates (future plan to have an external version)
- Change log reviews (separated platform change log from generic code after v2.2)
- Assessments – include test plan review from assigned peer Open source teams internally

Release announcements
- External Release announcement in tf-a@lists.trustedfirmware.org
- Internal Release announcements to peer Open source teams.

Release Testing Flow

Gather patches to be included in release

Code Freeze

Initial Tag (RCO)

Run Manual and Automated Tests

Failure(s)

Debug Fails

ReTag (RCX)

Pass of all tests

Final Tag

Change Log Creation

Take note of testing methods and update test plan

Test Categories: CI automated, Manual->juno_tftf_extensive, >juno_fwu_mbedtls, >get_change_mem_attributes- >xlat_v2_unit_tests, ->Cryptocell, ->stress, ->linux, ->tftf_manual, >out_of_box

Key Challenges:
- Security issues are properly vetted
- Moving as much of the manual testing as possible to automation
- Many issues with Juno board platforms
TF-A & TF-A_Tests v2.3 Highlights

TF-A Highlights

- Add support for Armv8.4-SecEL2 extension through the SPCI defined SPMD/SPMC components.
- Build option to support EL2 context save and restore in the secure world (CTX_INCLUDE_EL2_REGS).
- Add support for SMCCC v1.2 (introducing the new SMCCC_ARCH_SOC_ID SMC). Note that the support is compliant, but the SVE registers save/restore will be done as part of future S-EL2/SPM development.

TF-A Tests Highlights

- More tests are made available in this release to help validate the functionality of TF-A.
- CI upgraded to use GCC 9.2-2019.12 toolchain for tf-a-tests.
- Various improvements to test framework and test suite.


TF-A & TF-A-Tests v2.3 Documentation

Improvements we could look for in future releases

- **Improvements for release cycle**
  - Further automations by using Juno boards over network PDU
    - Potential for automating some manual tests
  - Failure signature categorization and automated re-run of known false failures based on signature
    - Could save run time by maximizing runs on weekends
  - Can change log entry for patches be closer to final by code freeze?
    - using review process to review change log updates too when patches are submitted?

- **Other improvements**
  - Future availability of openCI
    - enables partners to adopt similar test flows for their platforms
  - Partner use of Release Candidates for Platforms
    - Have the RC communicated on tf-a@lists.trustedfirmware.org been used?
    - Communication needs/criteria of additional RCs before final tag