

Long Term Support Proposal

Okash Khawaja - Google
Varun Wadekar - NVIDIA

Soul

- Propose a long term support story for all projects on tf.org
- Try to close all the open questions
- Gather feedback from the community

Motivation

- Once products are released, they need to be maintained in-field with minimum risk
- Minimise risk when deploying updates to live server systems e.g. those in data centers
- Full update of TF-A in-field is considered too risky, but critical bugs need patching
- Best option is incremental updates with the bare minimum changes to minimise risk and churn during incident response
- Reduce or eliminate efforts for partners that manage their own LTS branches downstream

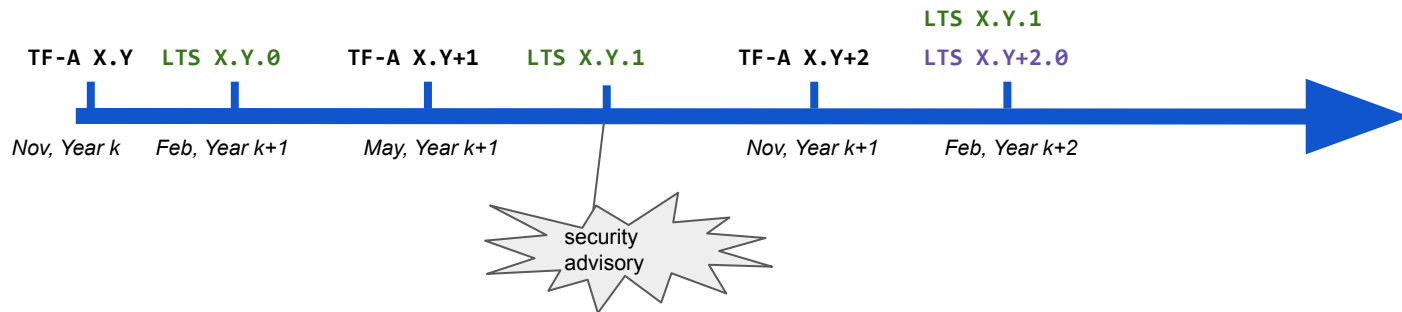
Scope of the branch

- Security advisories
- CPU errata fixes
- Arm IP errata fixes
- Partner IP errata fixes
- Platform bug fixes
- No new features
- Always aligned with mainline branch
- Security advisories for external components e.g. libfdt

Release model

- One release per year in Nov*, supported for five years
- Soak time to allow additional testing and hardening
- Separate branch; leverage testing provided by the current CI/CD flow
- Patches flow from main branch to LTS
- Platforms already in the upstream testing farm will get coverage by default
- Downstream platforms will have to verify LTS branches separately to ensure quality for their platforms

Example branch timeline



- Two releases per year - Nov and May
- Nov release is used as the baseline for the LTS branch
- Soak time introduced to allow additional testing and hardening
- LTS branch version is bumped when a fix is merged

Guidelines

- Impartial panel of maintainers for reviews
- Objective and well-defined merge criteria to avoid confusion and discussions at random points in time when there is a "candidate" patch
- Automate, automate, automate
- Reviewers should not focus too much on "what" and instead focus on "how"
- Constantly refine the merge criteria to include more partner use cases

Testing criteria

- Pre-defined set of TFTF tests
- CI/CD static analysis scans
- Coverity scans
- Platform tests (downstream and upstream)

Open questions

- Is Nov LTS release convenient for all partners?
- How do we handle LTS-only bug fixes?
- Maintainership model: TF.org or a panel of maintainers made from partners?
- Can we increase the LTS branch support for 7 years?
- Should we include bug fixes for external components? E.g. libfdt

Call to action

- Review this proposal and provide feedback
- Help us refine the execution model for the LTS branch

