TrustedFirmware.org security incident handling process

Change proposal

Current process overview

- Unified Process for handling SW security incidents available since 2020
  - Active for TF-A, TF-M, OP-TEE, Hafnium, MCUboot and Mbed TLS projects
  - Errata workarounds for Arm IP vulnerabilities uses a different Arm-owned process
- Was originally broadly based on the kernel process (which has evolved significantly since)
  - With influence from earlier TF-A and OP-TEE project-specific processes
  - Contrast is that this process combines vulnerability fixing and disclosure (kernel process only handles former)
- Reporters can use top level email alias or project specific aliases
- 2 stage disclosure process for vulnerability fixes
  - 1st stage for Especially Sensitive Stakeholders (ESS, used by TF-A only), 2nd stage for other Trusted Stakeholders (TS)
  - At each stage stakeholders can request up 7 days embargo, 14 in exceptional cases
  - Fix is then made public by default but vulnerability information remains under embargo
  - Export control prevents some projects (e.g. Mbed TLS) disclosing fixes to a restricted audience
- Vulnerability information made public as an advisory by default 90 days after initial report

Problem statement

- Volume of security incidents has dramatically increased
  - e.g. TF-A went from ~10 in 10 years to ~10/month
  - AI tools have made it much easier to identify potential issues
    - Initially generated a lot of false positive but hit rate is now much better
  - Arm bug bounty program has incentivized reporting
    - Has now been paused but there's a significant pipeline to flush out
- Cost of going through security incident process is high
  - Triaging complex AI-generated reports takes a long time
  - Increasing amount of reporter comms when triaging bug bounty submissions
  - Manually creating advisories and disclosing to stakeholders is slow
  - CVE create/update process is slow (Mitre are unresponsive)
  - Lack of tooling/automation
- => Known bugs remain unfixed for longer
- These problems are shared by many open-source projects

Kernel changes (some of this is radical)

- Torvalds declared in May that security@kernel.com has become unmanageable
- Bugs found with AI assistants should be treated as public
- Prioritize handling of bugs with a provided fix (AI tools are good at this)
- Get reporters to send directly to affected subsystem maintainers, cc'ing security team
- CVE assignment is done in a separate process and usually after the fix has been merged – these are batched on known dates
  - The Kernel is now its own CNA
- Limited 7-14 day embargo period retained (for non-public bugs)

Observations

- It’s hard to be sure that reports are AI-assisted
  - Leaving security teams to make subjective judgement could lead to disputes
- Initial ESS/TS vulnerability notification is not that painful
- The painful parts (for some projects) are:
  - CVE/advisory creation/management, can be mitigated by:
    - Being a CNA, if GitHub Security Advisory feature not applicable
    - Decoupling CVE creation from vulnerability fixing
    - Reducing level of CVE detail
  - Managing per-vulnerability public disclosure dates and long embargo periods
    - Can be mitigated by batching CVE public disclosures on fixed schedule

TF Proposal

- Adopt some of the kernel changes…
- Ask reporters to contact code owners directly, cc'ing project security team
  - Retire/hide top-level security alias – it's not adding any value
- Ask reporters to provide a fix in patch form (accept that quality may be low)
- Publish CVEs and start LTS backports after merge but keep as part of main process
  - Need more reliable CVE creation – Linaro is becoming a CNA soon
  - Projects can then link directly to CVEs in their security pages instead of creating separate advisories
  - Encourage projects to publish in batches, on known day (e.g. once per month/quarter) or at release time
  - Make CVE descriptions short and to the point (like commit messages) – don't obsess about impact and exploitability

TF Proposal [2]

- Continue notifying ESS/TS about vulnerabilities, and preferring fixes to be upstreamed quickly
  - Continue to encourage CVSS severity assessment, though not mandatory (no change)
- Optionally include platform code in the process
  - Requires platform owners to opt-in and meet conditions (e.g. be responsive)
  - One member has explicitly requested this
- Need to find time for better automation
  - e.g. ticket creation, templates for mails to stakeholders, CVE/advisory creation, LTS backports