Attendees: Dan Handley(Arm), Anton Komlev(Arm), Dominik(Nordic), Eric Finco(ST), Julius Werner(Google), Kevin Townsend(Linaro), Shebu(Arm), Don, Kangkang(Futurewei), Kevin Oerton(NXMLabs), Antonio(Arm), David Brown(Linaro), Andrej Butok(NXP)

Minutes:

- TF-M Roadmap (Shebu)
    - Shebu shared/walked through slides
    - TF-M v1.7 was a big release. v1.8 more of an incremental release
    - Enabled more PSA compliance tests with PSA ADAC library on MuscaB1 platform
    - AntonK: Regarding KConfig, not just about improving flexibility, but also scope. Can now configure whole project with KConfig
    - AntonK: Regarding T_COSE, one small function is not upstream. Hoping to be able to migrate to the upstream version next time.
        - KevinT: Suspect there are other changes, which might make it hard to migrate to upstream in the short term.
    - AntonK: In future, we hope to enable building S and NS parts as separate projects instead of 1 monolithic project, to help with integration.
    - Theme for last year and this year is non-functional improvements (e.g. memory, performance)
    - Cryptocell driver refactoring: Want to show better examples of applications interacting with crypto hw
    - One thing not shown is the Runtime Security Subsystem (RSS) roadmap. Presentation on this at next week's Linaro Connect.
    - This will be realised in the TF-M project
    - Shebu advertised TF related Linaro Connect sessions:
        - KevinT: DavidB also has a talk on Friday, regarding T_COSE work we're doing.
        - Videos will be made publicly available afterwards.
        - Don has arranged an informal TF session on Thursday afternoon (4pm) if there's anything else to discuss outside the sessions.
    - Eric: Thought there was an item task around a tool to generate TF-M as a CMSIS pack. Has this been dropped?
        - Unfortunately, tool is not scalable for complex projects like TF-M. Works for smaller projects but you have to generate manually for TF-M. No effort planned on this at the moment.
- PSA crypto API issues in TF-M (Antonio)
    - Antonio walked through slides
    - In case of TF-M, all implementation specific behaviour is in crypto_struct - crypto_platform is empty
    - KevinT: I think this will solve most of our problems. Goal is to get these changes into Zephyr.
    - KevinT: Got TF-M working with Zephyr and NS MbedTLS. Now plan to upstream changes to Zephyr for the next release with TF-M 1.8 and MbedTLS 3.4.
    - KevinT: Next Zephyr release conveniently has a freeze date the same week as TF-M. Please let me know of any issues

    - Andrej: We are compiling Mbed TLS and TF-M as a single project, which causes problems when building.
    - Antonio: We also added a change to introduce #defines at link time.
    - Andrej: But only for NS apps? Need it for secure apps too.
    - Antonio: Didn't have a use-case for that until now. May need to think about this some more.
    - DavidB: Is there a plan to do this "right". E.g. Single API supporting multiple different implementations in the same build? Any future plans to get to the point where a key can be created on the secure and/or non-secure side?
    - Antonio: Personal view is when TF-M is used, all will be done by 1 implementation. With TF-M in TZ, the aim is for all crypto to happen on the secure side. Can see why one may want to do it on the NS side, but the effort may be too large. Don't see this happening soon.
    - DavidB: Imagine when TLS is on the NS side. With this solution, keys would always be on the secure side. Don't know if there is a use-case for having keys on the NS side, but there could be. In Zephyr, this results in a tight coupling between TF-M and Mbed TLS.
    - DavidB No working LTS versions for this yet. So won't get fixes backported.
    - KevinT: We're getting pushback from Zephyr TSC
    - KevinT: TF-M effectively decides what version of Mbed TLS everyone in the Zephyr ecosystem uses.
    - Antonio: Fair points. ABI compatibility is out of scope for now but when this is addressed, this problem should go away
    - DavidB: Maybe we just need to define this as a goal.
    - DavidB: Might need some commitment from MBed TLS. It's not just about API source compatibility.
    - Shebu: Only going to happen when PSA Crypto API development settles.
    - DavidB: TF-M chose to use a non LTS version, which is passed on to the rest of the Zephyr ecosystem.
    - Shebu: TF-M had to pick up a newer Mbed TLS version to use newer features.
    - Shebu: Can only do this when development settles down.
    - DavidB: Effectively we have 2 different use-cases in Zephyr.
    - Shebu: Focus currently is on creating a new PSA Crypto repo.
    - Shebu: Wasn't there an earlier discussion about Zephyr supporting both LTS and tip versions?
    - DavidB: Difficult with current Zephyr config system.

- AOB
    - Future TSC Topics:
        - Secure world use of vector features to push out to next TSC
        - TF.org position on Rust is a future topic
            - David Brown : Clarify secure side vs non-secure side usage of Rust
        - MBed TLS roadmap for next meeting
        - Open to other topics
        - Always interested in member use cases

- KevinT: Can present work on Confidential AI? Can present if interested.