TF-A Tech Forum
Secure EL2 firmware

Olivier Deprez (Arm)
Andrew Walbran (Google)
July 2020
Agenda

• Why an S-EL2 firmware?
• What is Hafnium?
• Hafnium as the S-EL2 firmware
• Project goals and status
• TF-A release contents
• PSA FF-A adoption
• CI and testing
• Upstream activity
• Q&A
Why an S-EL2 firmware?

- “Isolation using virtualization in the Secure world” white paper
- Current architecture
Why an S-EL2 firmware?

- Challenges
  - Trusted Applications ecosystem
  - Integration of code from multiple vendors in the secure world
  - Principle of least privilege
  - Normal world protection from secure world
Why an S-EL2 firmware?

- **Challenges**
  - Trusted Applications ecosystem
  - Integration of code from multiple vendors in the secure world
  - Principle of least privilege
  - Normal world protection from secure world

- **Solution**
  - Isolation between multiple mutually mistrusting Trusted OSes
  - Leverage “Secure EL2” Armv8.4-SecEL2 extensions
  - PSA FF-A (formerly SPCI)
    - Secure Partition Manager at S-EL2
    - Standard APIs across boundaries (Hypervisor/VMs, SPM/SP, Hypervisor/SPM)
What is Hafnium?

- Originally a Google project
- Type-1 “bare-metal” Hypervisor running in the Normal World
- Supports AArch64 NS-EL2
What is Hafnium?

- Originally a Google project
- Type-1 “bare-metal” Hypervisor running in the Normal World
- Supports AArch64 NS-EL2

- Instantiates untrusted VMs at NS-EL1
- Isolates VM memory through Stage-2 MMU
- Provides VM-to-VM communication
- Low latency primary VM schedules secondary VMs
What is Hafnium?

- Originally a Google project
- Type-1 “bare-metal” Hypervisor running in the Normal World
- Supports AArch64 NS-EL2

- Instantiates untrusted VMs at NS-EL1
- Isolates VM memory through Stage-2 MMU
- Provides VM-to-VM communication
- Low latency primary VM schedules secondary VMs

- TEE communication infrastructure (e.g. Trusty)
What is Hafnium?

- Originally a Google project
- Type-1 “bare-metal” Hypervisor running in the Normal World
- Supports AArch64 NS-EL2

- Instantiates untrusted VMs at NS-EL1
- Isolates VM memory through Stage-2 MMU
- Provides VM-to-VM communication
- Low latency primary VM schedules secondary VMs

- TEE communication infrastructure (e.g. Trusty)

- Fast build system
- Build targets FVP, QEMU, Rpi
- Hafnium test suite
Hafnium as the S-EL2 firmware

- Proposed TF-A end-to-end stack
Hafnium as the S-EL2 firmware

- Hafnium is a good fit to fulfil isolation of multiple Secure Partitions
- Aim to extend the VM isolation in NWd, to Secure Partitions in the SWd
Hafnium as the S-EL2 firmware

• Hafnium is a good fit to fulfil isolation of multiple Secure Partitions
  • Aim to extend the VM isolation in NWd, to Secure Partitions in the SWd

• Transition to trustedfirmware.org completed in June 2020
  • Project relicensed to BSD-3. Linux driver GPLv2
  • Co-maintained by Arm and Google
  • Handled by OSS TF-A team within Arm
Hafnium as the S-EL2 firmware

• Hafnium is a good fit to fulfil isolation of multiple Secure Partitions
  • Aim to extend the VM isolation in NWd, to Secure Partitions in the SWd

• Transition to trustedfirmware.org completed in June 2020
  • Project relicensed to BSD-3. Linux driver GPLv2
  • Co-maintained by Arm and Google
  • Handled by OSS TF-A team within Arm

• NWd Hypervisor codebase mainly maintained and used as:
  • a test vehicle to schedule SPs
  • a sanity checker for PSA FF-A in the NWd (Hypervisor)
Project goals

• Open-source S-EL2 “Secure Partition Manager” reference firmware
• Part of broader trustedfirmware.org reference implementation
Project goals

• Open-source S-EL2 “Secure Partition Manager” reference firmware
• Part of broader trustedfirmware.org reference implementation

• Adopt PSA FF-A in Hafnium code base
• Well progressed on the NWd side and benefits SWd
• But still requires further additions/adaptations:
  • Booting in the SWd
  • Power management
  • Missing FF-A ABIs
  • World switch through SPMD
  • Memory sharing (VM-SP / SP-SP)
  • Interrupt handling
Project status

- Phased development started in the open
  - SPM boot story, prototyping
  - Secure boot enablement, FF-A setup and discovery, patch upstream kick-off
  - Multicore boot, Multiple secure partitions, Interrupt management, memory sharing
  - IO/SMMU support, S-EL0 only partitions, AArch32 SPs
TF-A releases contents

• v2.3
  • Armv8.4-SecEL2 extension support
  • SPMD supports SPMC at S-EL1 or S-EL2
  • EL2 context save/restore in SWd
  • Platform changes to boot Hafnium/SPMC at S-EL2
  • Secure Partitions packaging

• v2.4 (planned by Q4)
  • Secure boot of Secure Partitions
  • SPMC first tag in a TF-A release
    • FF-A setup and discovery interfaces
    • Minimum support to boot OP-TEE as a SP + NWd driver probing
    • Reproducible builds and CI
PSA FF-A adoption

- Migration path for TOS vendors
PSA FF-A adoption

- Migration path for TOS vendors
- SPMD support for pre-Armv8.4 systems
  - SPMC at S-EL1
PSA FF-A adoption

- Migration path for TOS vendors
- SPMD support for pre-Armv8.4 systems
  - SPMC at S-EL1
- Generic SPMD
  - Migrate existing SPDs to Arm standards as much as possible
PSA FF-A adoption

- Migration path for TOS vendors
- SPMD support for pre-Armv8.4 systems
  - SPMC at S-EL1
- Generic SPMD
  - Migrate existing SPDs to Arm standards as much as possible
- SMC service forwarding to SPMC
  - Former EL3 SPM using MM interface
  - Hooks for new services
CI

- Hafnium CI
  - Host unit tests, arch tests, VM API tests, Linux tests
  - Run manually on the developer machine, possibly using docker
  - Jenkins job run on any patch submission, requires Verified+1 to merge
CI

- Hafnium CI
  - Host unit tests, arch tests, VM API tests, Linux tests
  - Run manually on the developer machine, possibly using docker
  - Jenkins job run on any patch submission, requires Verified+1 to merge
  - Daily TF-A CI test runs
  - Expect to improve both “CIs” integration as part of OpenCI project

<table>
<thead>
<tr>
<th>Description</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boot OPTEE as SPMC at S-EL1 (pre-Armv8.4)</td>
<td>TFTF at NS-EL2 + SPMD at EL3 + OP-TEE/SPMC at S-EL1</td>
</tr>
<tr>
<td>Check Hafnium/SPMC boot using Armv8.4-SecEL2 extensions</td>
<td>TFTF at NS-EL2 + SPMD at EL3 + Hafnium/SPMC at S-EL2 + Cactus at S-EL1</td>
</tr>
<tr>
<td>Hafnium Hypervisor and SPMC using Armv8.4-SecEL2 extensions</td>
<td>Linux PVM at NS-EL1 + Hafnium/Hypervisor at NS-EL2 + SPMD at EL3 + Hafnium/SPMC at S-EL2 + Cactus at S-EL1</td>
</tr>
<tr>
<td>Bare-metal secure partitions, check Linux boot in PVM</td>
<td></td>
</tr>
<tr>
<td>Boot OP-TEE as a Secure Partition on top of SPMC</td>
<td>TFTF at NS-EL2 + SPMD at EL3 + Hafnium/SPMC at S-EL2 + OP-TEE Secure Partition at S-EL1</td>
</tr>
<tr>
<td>OP-TEE as a Secure Partition on top of SPMC, Linux boot and</td>
<td>(under development)</td>
</tr>
<tr>
<td>OP-TEE kernel module init.</td>
<td>Linux at NS-EL1 + SPMD at EL3 + Hafnium/SPMC at S-EL2 + OP-TEE Secure Partition at S-EL1</td>
</tr>
<tr>
<td>Secure boot of secure partitions using dual root key CoT</td>
<td></td>
</tr>
<tr>
<td>Secure boot of secure partitions using TBBR single root key CoT</td>
<td></td>
</tr>
</tbody>
</table>
Upstream activity

- Now happening through https://review.trustedfirmware.org
- 3 patches merged
- 15 WIP/under review
  - Early bring up patches
  - Direct messaging implementation
  - FF-A manifest parsing
  - Partition info get FF-A ABI
  - https://review.trustedfirmware.org/q/topic:%22spm-wip%22+(status:open%20OR%20status:merged)
Resources

- Meet at Linaro Virtual Connect September 2020
- Hafnium documentation
  - https://review.trustedfirmware.org/plugins/gitiles/hafnium/hafnium/+/HEAD/README.md
- PSA FF-A
- TF-A SPM documentation
  - https://review.trustedfirmware.org/c/TF-A/trusted-firmware-a/+/4637
- Gerrit code reviews
  - https://review.trustedfirmware.org/q/status:open+project:hafnium/hafnium
- ML
  - https://lists.trustedfirmware.org/mailman/listinfo/hafnium
- Phabricator
  - https://developer.trustedfirmware.org/tag/hafnium/
TFTF and Cactus TF-A-tests test harness

- Testing at NS physical FF-A interface
- Cactus bare-metal partitions
SPMD support for pre-Armv8.4 systems

NS-EL0
- Client app.
- Kernel
- OP-TEE driver
- pFF-A

NS-EL1
- OP-TEE driver
- pFF-A

NS-EL2
- SPMC
- OP-TEE
- TA
- TA

S-EL0
- S-EL1
- S-EL2

EL3
- SPMD

© 2020 Arm Limited (or its affiliates)