FF-A SPMC at EL3

Marc Bonnici, Olivier Deprez
30/06/2022
Introduction

The FF-A (Arm Firmware Framework for Arm A-profile) specification [1] provides a standardised interface between two sandboxes (VMs, SPs etc.)

Key focus areas:

- Discovery
- Communication
- Memory Management

Why Do We Need FF-A?

Pain Points
- Fragmentation and multiple implementations across components
- Duplicated Code

Need for Standardization
- Provide a standard interface to partitions for common tasks
- Single implementation of the programming model across components
- Reduced integration cost
- Support for configurations both with and without S-EL2
  - Provides a migration path for pre 8.4 platforms
  - Changes to secure world configuration can be transparent to the normal world
- Improved portability for a TEE SP
Firmware Framework for Armv8-A

How does it all fit together on a < Armv8.4 system

- VMs implement a generic FF-A driver
  - Implements the S/NS FF-A programming model
  - Provides a transport for service specific protocols

- SPs implement a generic FF-A driver
  - Implements the SP/SPMC FF-A programming model
  - Provides a transport for service specific protocols

- Hypervisor implements a generic FF-A driver
  - Dispatches messages between the two worlds
  - Implements the S/NS FF-A programming model

- SPMD is a generic component in EL3
  - Dispatches messages between the two worlds
  - Agnostic of SP and SPMC implementation

- SPMC is a firmware component in EL3
  - Implements the FF-A programming model
  - Provides a logical isolation for an SP

Single TOS migrates to FF-A
Firmware Framework for Armv8-A

How does it all fit together on a < Armv8.4 system

- SP could also be deployed as S-EL0 Partition
  - Implements the S/NS FF-A programming model
  - Provides a transport for service specific protocols

- Currently available with SPM MM dispatcher
  - Functionality replaced with the FF-A EL3 SPMC

EL3 firmware manages S-EL0 SP
EL3 SPMC: What and Why?

- The EL3 SPMC is the implementation of the FF-A SPMC directly in EL3
  - Experimental support added to TF-A v2.7
- Supports systems without S-EL2 e.g. pre v8.4 platforms
- Serves as a migration path to help transition to running under S-EL2
  - Alignment between Hafnium (S-EL2 SPMC) and EL3 SPMC for SP manifest and major features
- Working closely with open source TOS and other partners during the review process to help ensure target use cases can be met
  - OP-TEE
  - Trusty
EL3 SPMC: Supported Features

- **Single Multi-Core S-EL1 SP Support**
  - SP Entry Point Registration
  - Pinned CPU contexts
  - Ongoing work for supporting S-EL0 partitions

- **Direct Messaging**
  - Register based message passing

- **Logical EL3 Partitions**
  - A simple entity in EL3 that can be communicated with via direct messaging

- **Partition Discovery**
  - Including EL3 and S-EL1 Partitions
  - Partition IDs & partition information

- **Memory Management**
  - Lend memory
    - Lender loses access
    - Borrower(s) share access
  - Share memory
    - Lender and borrower(s) both have access
  - Implemented ABIs:
    - MEM_LEND/MEM_SHARE
    - MEM_RETRIEVE_REQ/RESP
    - MEM_RELINQUISH/MEM_RECLAIM
  - Inc. Fragmented Descriptor Transmission
    - FFA_FRAG_TX
    - FFA_FRAG_RX
  - Support for multiple borrowers
    - From the normal world
  - Inc. Platform hooks for IMPDEF behavior
    - E.g. MPU programming

- **Power Management**
  - Supports Hot-plugging CPUs from Linux
  - SP Subscription to power events:
    - CPU_OFF
    - CPU_SUSPEND
    - CPU_SUSPEND_RESUME

- **Interrupt Support**
  - Signaling and completion mechanisms

- **State Tracking**
  - ABIs invocations / interrupts provoke state transitions

- **FF-A Boot Protocol v1.1**
  - Provides access to an SP’s DT to the SP itself.
EL3 SPMC: S-EL0 Support

- Upstream EL3 SPMC patches currently only support an S-EL1 partition.
- Support to enable a single S-EL0 partition is under development and almost ready to post for review.
- Currently validated use cases:
  - Secure boot
  - RAS
EL3 SPMC: Build Options

```make
make \ 
CROSS_COMPILE=aarch64-none-elf- \ 
SPD=spmd \ # Using FF-A SPMD Component 
SPMD_SPM_AT_SEL2=0 \ # Running on a system without S-EL2 
SPMC_AT_EL3=1 \ # Enable the FF-A EL3 SPMC 
BL32=<path-to-tee-binary> \ # S-EL1 SP image 
BL33=<path-to-bl33-binary> \ 
PLAT=fvp \ 
all fip
```

### Example Boot Logs

<table>
<thead>
<tr>
<th>INFO:</th>
<th>BL31: Initializing runtime services</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFO:</td>
<td>EL3 Logical Secure Partition init start.</td>
</tr>
<tr>
<td>INFO:</td>
<td>LSP: Init function called.</td>
</tr>
<tr>
<td>INFO:</td>
<td>EL3 Logical Secure Partition init completed.</td>
</tr>
<tr>
<td>INFO:</td>
<td>Secure Partition context setup start.</td>
</tr>
<tr>
<td>INFO:</td>
<td>Manifest size = 465 bytes.</td>
</tr>
<tr>
<td>INFO:</td>
<td>SP boot info @ 0x4021000, size: 529 bytes.</td>
</tr>
<tr>
<td>INFO:</td>
<td>SP manifest @ 0x4021040, size: 465 bytes.</td>
</tr>
<tr>
<td>INFO:</td>
<td>Entry point address = 0xff200000</td>
</tr>
<tr>
<td>INFO:</td>
<td>SPSR = 0x3c5</td>
</tr>
<tr>
<td>INFO:</td>
<td>Secure Partition setup done.</td>
</tr>
<tr>
<td>INFO:</td>
<td>BL31: Initializing BL32</td>
</tr>
<tr>
<td>INFO:</td>
<td>Secure Partition (0x8001) init start.</td>
</tr>
<tr>
<td>INFO:</td>
<td>Secure Partition initialized.</td>
</tr>
<tr>
<td>INFO:</td>
<td>BL31: Preparing for EL3 exit to normal world</td>
</tr>
<tr>
<td>INFO:</td>
<td>Entry point address = 0x88000000</td>
</tr>
<tr>
<td>INFO:</td>
<td>SPSR = 0x3c9</td>
</tr>
</tbody>
</table>

**EL3 Logical Partition Setup**

**SPMC Partition Setup**

**S-EL1 Partition Initialisation**

**Normal World Handoff**
TF-A Details and Next Steps
TF-A EL3 SPMC (Jun'22)

- EL3 SPMC core changes TF-A v2.7 May'22
  - Released as experimental feature.
  - 45 patches developed (Arm arch team), reviewed (TF-A + partners) and merged.
  - Single S-EL1 partition (TEE) configuration. Complies with FF-A v1.1 EAC0 specification.
  - Partner contributions welcome for new feature development onwards.

- Test and CI changes (10) under development/review.
  - TSP adopting FF-A. NS side linux based test driver.
  - Review & merge TSP+CI changes (Jul'22).

- FF-A Architecture Compliance Suite
  - Runs against the EL3 SPMC
  - Few fixes planned in coming weeks
  - Plan to document test results and waived findings.

- Hikey960 platform changes (7) under review.

- Documentation updates (Aug'22)
  - EL3 SPMC threat model and design doc.
EL3 SPMC to SEL2 SPMC features catch up

- FF-A v1.1 features picked early in the EL3 SPMC
- Goal to maintain a smooth migration from EL3 SPMC to SEL2 SPMC
- Catch up the SEL2 SPMC:
  - Memory sharing to multiple borrowers.
  - Memory sharing structures forward compatible.
  - NS bit passed in memory retrieve response.
  - Power management run-time.
  - FF-A ACS results "match".
Thank You
Danke
Gracias
Grazie
謝謝
ありがとう
ありがとう
Asante
Merci
감사합니다
धन्यवाद
धन्यवाद
شكرًا
תודה