Content

• DRM Applications and Secure Video Path
  • Regular Secure Video Path Design with Trustzone
  • TZMP1 Design Concepts
  • Reference Implementation Details
General Process of DRM

- **DRM Protocols**
  - Authorizing
  - Key Exchange

- **Encoded Content**
  - Casting
  - Encode/Encrypt

- **Encrypted Content**
  - Networking

- **Video Path**
  - Decrypt/Decode
Ideal Model of Secure Video Path

- Encrypted Content
- Secured Environment
  - Decrypt
  - Decode
  - Composite
  - Decoded Frame
- Encoded Content
Content

• DRM Applications and Secure Video Path

• Regular Secure Video Path Design with Trustzone
  • TZMP1 Design Concepts
  • Reference Implementation Details
Arm Trustzone

Non-Secure World

Non-Trusted Apps

Trusted OS

Secure World

Trusted OS

Trusted Application

Hardware isolated Execution Environment
Regular Design with TrustZone

Non-Secure World

Secure World

Decrypt

Receive

Compose

Decode

© 2018 Arm Limited
Issues of Regular Design

Mature multimedia frameworks in Non-Secure OS

Porting?

Lack of API Runtime more attack interfaces
Could we protect content in non-secure world to avoid much changes?

Restrict accessing into single direction
Content

- DRM Applications and Secure Video Path
- Regular Secure Video Path Design with Trustzone

**TZMP1 Design Concepts**

- Reference Implementation Details
Protected Memory and Secured Playback

- Non-Secure Bitstream could only be accessed by Non-Secure hardware components.
- Secure Bitstream could only be accessed by ‘Protected’ hardware components.
- Display controller could only be accessed by ‘Protected’ hardware components.
- Display controller could read both types of bitstreams.
- Mention the word ‘Protected’ – leads to Protected Memory and Protected Mode of HW.
Firewall of Accessing - Arm Trustzone Controller 400

<table>
<thead>
<tr>
<th>Region</th>
<th>Ranges</th>
<th>NSAID 1</th>
<th>NSAID 2</th>
<th>…</th>
<th>NSAID 16</th>
<th>Secure Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>All Memory</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>1</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>2</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>3</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>4</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>5</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>6</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
<tr>
<td>7</td>
<td>Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
<td>RW Configurable</td>
</tr>
</tbody>
</table>

- Each non-secure memory accessing hardware is assigned with a Non-Secure Access ID (NSAID)
- TZC-400 checks NSAID and region permissions to decide access availability
- Total 8 regions and 16 NSAIDs are supported in TZC-400
- Secure accessing is also checked by TZC-400
Content

• DRM Applications and Secure Video Path
• Regular Secure Video Path Design with Trustzone
• TZMP1 Design Concepts

• Reference Implementation Details
# Platform and Software Components

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Board</td>
<td>Juno-r2 + 2 x Logictile</td>
</tr>
<tr>
<td>Non-Secure</td>
<td>Android</td>
</tr>
<tr>
<td>OS</td>
<td>lsk-4.4-armlt Kernel</td>
</tr>
<tr>
<td>Secure OS</td>
<td>OPTEE OS</td>
</tr>
<tr>
<td>Boot</td>
<td>Arm-tf</td>
</tr>
<tr>
<td>Media IP</td>
<td>Arm Mali V550 G71 DP650</td>
</tr>
<tr>
<td>DRM</td>
<td>Clearkey / Widevine</td>
</tr>
</tbody>
</table>
## Overall Reference Implementation

<table>
<thead>
<tr>
<th>ExoPlayer</th>
<th>Media Framework</th>
<th>SurfaceFlinger</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRM Plugin</td>
<td>OMX.decoder.secure</td>
<td>Protected Surface</td>
</tr>
<tr>
<td>Crypto CA</td>
<td>OMX.decoder</td>
<td>Surface</td>
</tr>
</tbody>
</table>

### Drivers

- OP-TEE OS (TEE)
- Crypto TA

### Hardware

- VPU
  - NSAI D PROT
  - NSAI D_PUB
- GPU
  - NSAI D_PR
  - NSAI D_PU
- DPU

### Protected Memory

- Decrypted
- Decode

### Non-Secure Memory

- Clear Bitstream
- Decode
- Decoded
Required Software Modifications

- ExoPlayer
  - Media Framework
    - OMX.decoder
    - OMX.decoder.secure
    - SGL CA
  - SurfaceFlinger
    - Protected Surface
    - Surface
    - Gralloc

- Drivers
  - ION

- Hardware
  - VPU
    - NSAID PROT
    - NSAID_PUB
  - GPU
    - NSAID_PR
    - NSAID_PUB
  - DPU
    - NSAI

- OP-TEE OS (TEE)
  - Crypto TA
  - SGL TA

- TZC400 Setup
## Memory Regions

<table>
<thead>
<tr>
<th>Android (Linux)</th>
<th>Kernel Managed</th>
<th>ION::UNMAPPED HEAP</th>
<th>ION::CARVEOUT</th>
<th>TEE PARAM</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Regions</td>
<td>Non-Secure Memory</td>
<td>SECURE Firmware</td>
<td>SECURE Bitstream</td>
<td>SECURE Frame</td>
<td>TEE PARAM</td>
</tr>
<tr>
<td>OPTEE OS</td>
<td>X</td>
<td>Secure Data Path</td>
<td></td>
<td>TEE PARAM</td>
<td>Runtime</td>
</tr>
</tbody>
</table>

© 2018 Arm Limited
Secure VPU Firmware Loading

- Mandatory for firmware based decoder
- Unnecessary for non-firmware based decoder
- SDP Usage
Adopt DRM Crypto

- Decrypt in OPTEE OS
- Put result into protected memory
- Take advantage SDP of OPTEE OS SDP
Adopt Secure Decoder

Android

ExoPlayer

Media Framework

OMX.decoder

OMX.decoder.secure

Drivers

Hardware

VPU

NSAID_PROT

NSAID_PUB

Decrypted

Decode

‘Secure Video Path’ in Android

Is DRM required ‘Secure Codec’?

Is ‘Secure Codec’ available?

Setup Secure Video Path

Choose ‘Secure Codec’ component

Apply Protected Surface for output

SECURE FRAME

SECURE BITSTREAM

SECURE FRAMEWORK
Graphics and Display

- GPU and Display calls gralloc for surface buffers
- Gralloc allocates memory from specified buffer due to flags
- Call ION APIs for protected buffer
## References

<table>
<thead>
<tr>
<th>Components</th>
<th>Repository</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Workspace</td>
<td>To be upstreamed in April</td>
</tr>
<tr>
<td>2 Arm-tf</td>
<td><a href="https://github.com/ARM-software/arm-trusted-firmware">https://github.com/ARM-software/arm-trusted-firmware</a> (Upstreaming)</td>
</tr>
<tr>
<td>3 OPTEE OS</td>
<td><a href="https://github.com/OP-TEE/optee_os">https://github.com/OP-TEE/optee_os</a> (Done)</td>
</tr>
<tr>
<td>4 Android manifest</td>
<td>To be upstreamed in April</td>
</tr>
<tr>
<td>5 Secure Gadget Library</td>
<td>Upstreaming in linaro private repository</td>
</tr>
<tr>
<td>7 Multimedia IP</td>
<td>Contact Arm support</td>
</tr>
<tr>
<td>8 Linux and DTS</td>
<td><a href="https://git.linaro.org/landing-teams/working/arm/kernel-release.git">https://git.linaro.org/landing-teams/working/arm/kernel-release.git</a> (Upstreaming)</td>
</tr>
<tr>
<td>9 Arm Connected Community Page</td>
<td>Planed to be done by ~April</td>
</tr>
</tbody>
</table>
Thank You
Danke
Merci
谢谢
ありがとうございます
Gracias
Kiitos
감사합니다
धन्यवाद
תודה