How to Break Secure Boot on FPGA SoCs through Malicious Hardware

Nisha Jacob, Johann Heyszl, Andreas Zankl, Carsten Rolfes, and Georg Sigl

CHES 2017, September 27th, 2017

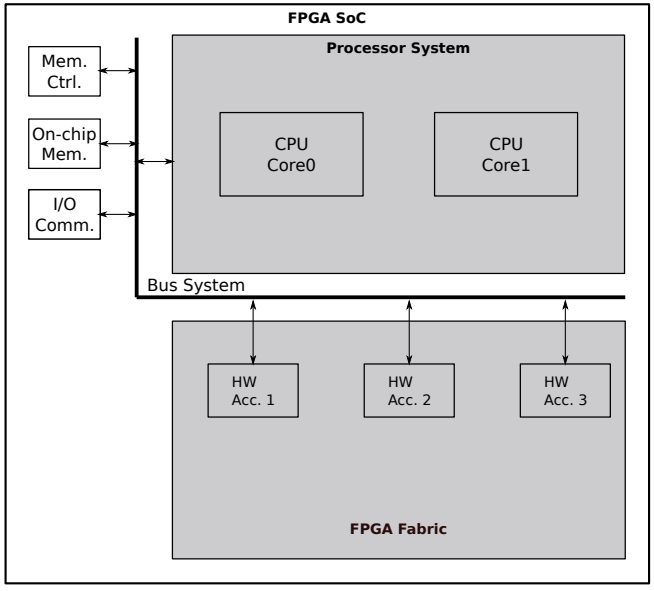
Fraunhofer AISEC

FPGA SoCs in a nutshell

- Include FPGA and hard-core CPU on the same die
- High performance CPU together with customizable hardware accelerators
- In-field updates, for both hardware and software

  e.g., update communication interface to fit new standards

- Shift to contemporary platforms like FPGA SoCs



Fraunhofer AISEC

How to get HW blocks for the FPGA SoCs?

- Lack of time and skill for in-house development

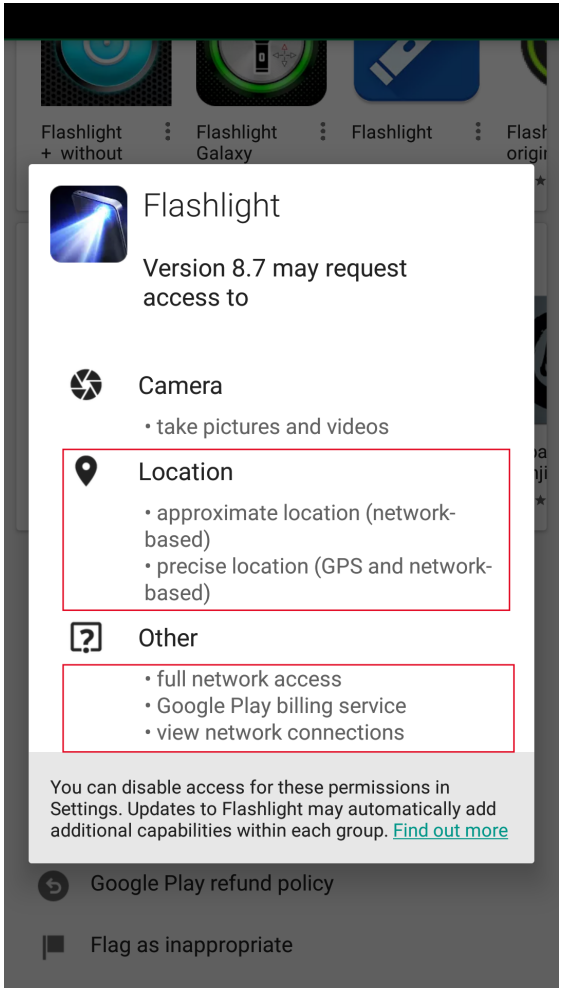
  e.g. as SW design team

- Outsource to third party

- Buy from somewhere else

  e.g. Amazon Web Services market place

Flash light mobile application



Fraunhofer AISEC

Threat of third party IP cores

- Third party IP needs to be tested before integration
- Requires time, resources and skills
- Often IP is delivered as a netlist
- Hard to verify IP does not contain additional functionality
- Malicious functionality in IP cores can ...

  e.g. corrupt sensitive data, memory or cause privilege escalation

Threat of third party IP cores

- Third party IP needs to be tested before integration
- Requires time, resources and skills
- Often IP is delivered as a netlist
- Hard to verify IP does not contain additional functionality
- Malicious functionality in IP cores can ...

  e.g. corrupt sensitive data, memory or cause privilege escalation

  → Do you trust the vendor?

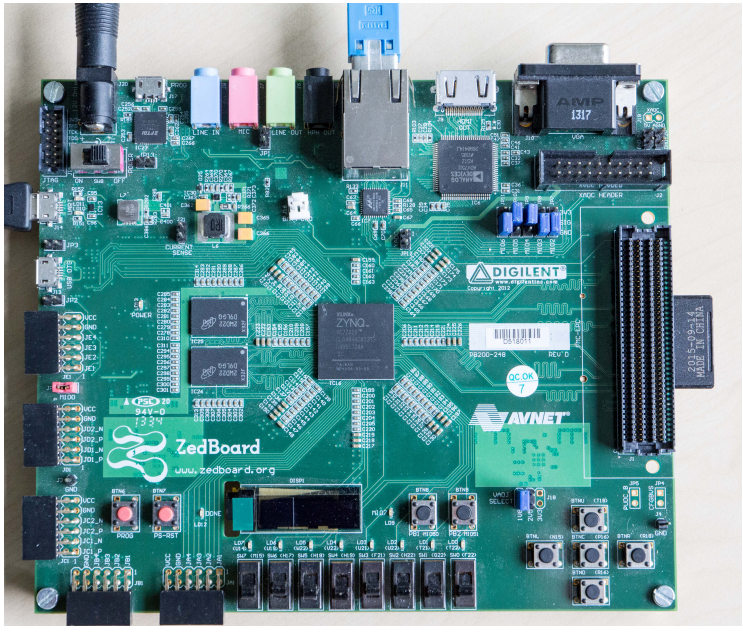
What is secure boot?

- One of the most important security mechanisms
- Foundation for all later security mechanisms
- All executed code is verified before execution
- After system startup, running software can be trusted
- Chain-of-trust is established starting from hardware-root-of-trust,

  e.g. keys in eFuses, ROM

Our contribution

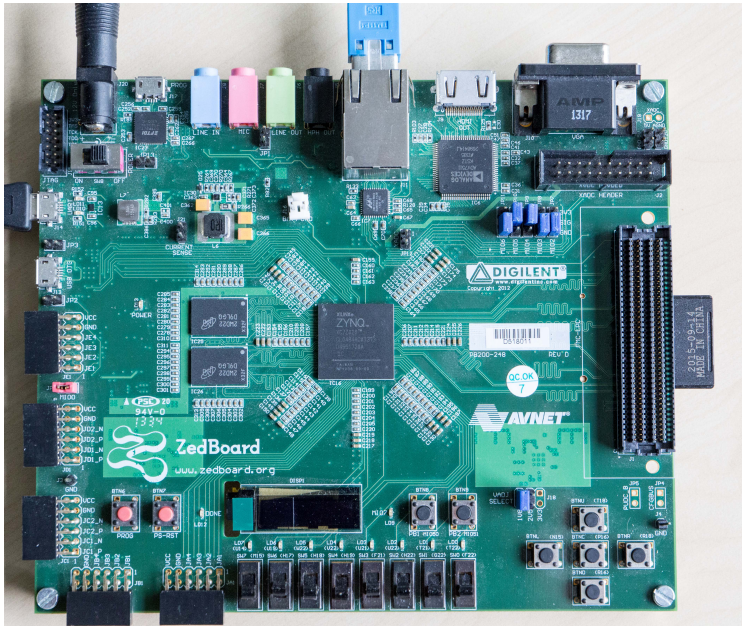
1. **How to break secure boot of FPGA SoCs, …**
   - on Xilinx Zynq-7000



Fraunhofer AISEC

Our contribution

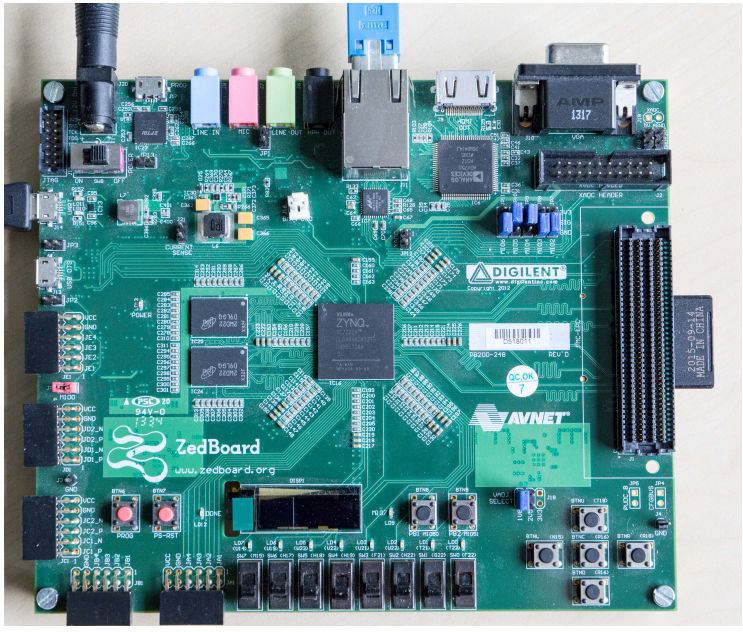
1. **How to break secure boot of FPGA SoCs, …**
   - on Xilinx Zynq-7000
2. **… generalise …**
   - impact on common security mechanisms
   - to other FPGA SoCs



Fraunhofer AISEC

Our contribution

1. **How to break secure boot of FPGA SoCs, ...**
   - on Xilinx Zynq-7000
2. **... generalise ...**
   - impact on common security mechanisms
   - to other FPGA SoCs
3. **... and how to protect against such threats**
   - using a security enhanced AXI-wrapper

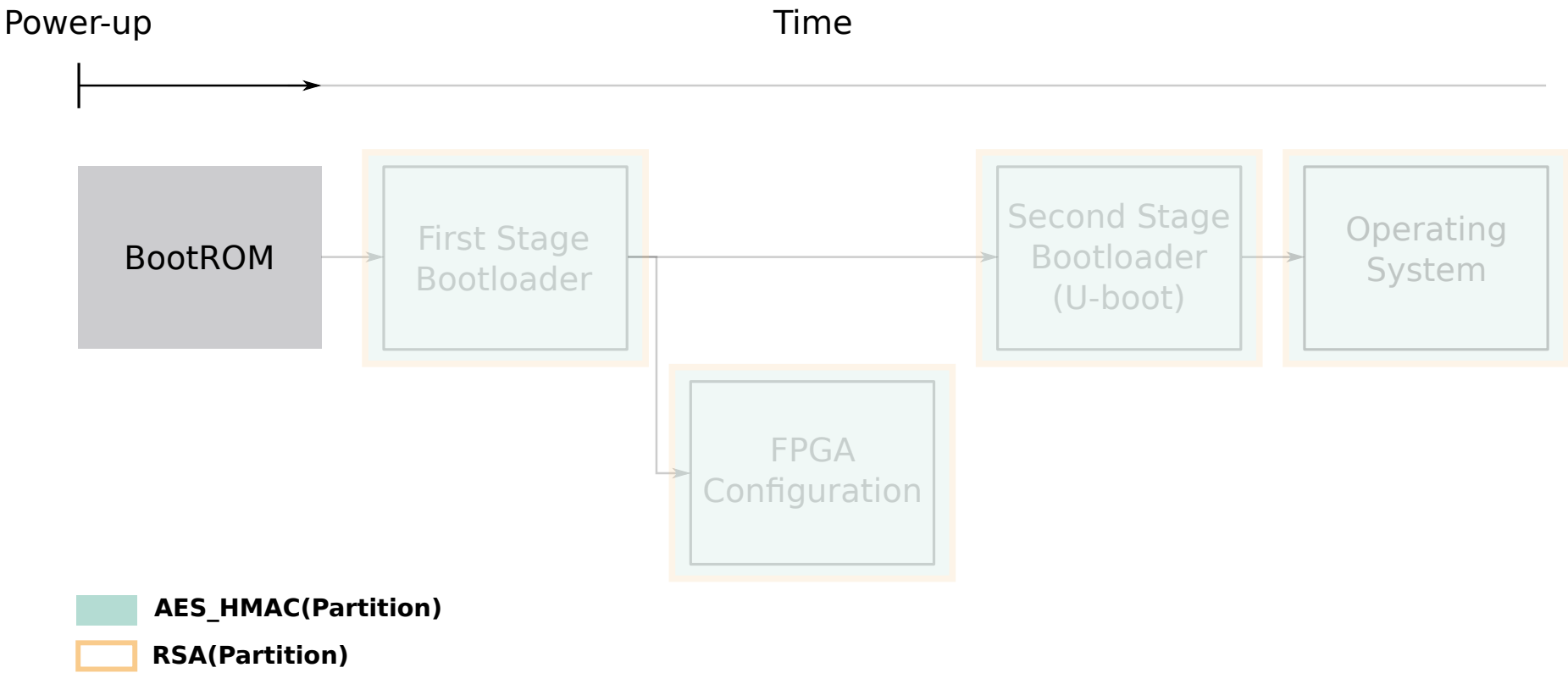


Fraunhofer AISEC

How to break secure boot of FPGA SoCs...

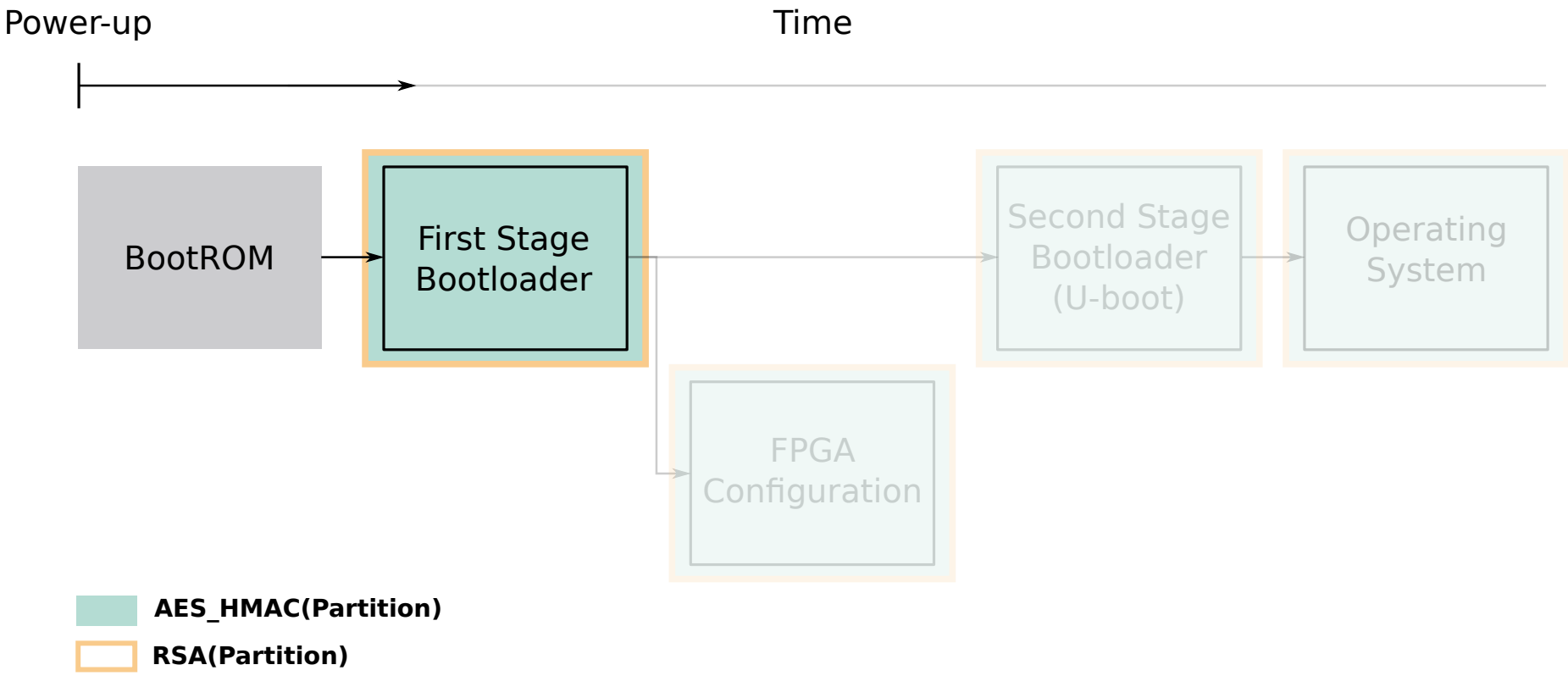
on Xilinx Zynq-7000

Secure boot on Xilinx Zynq-7000



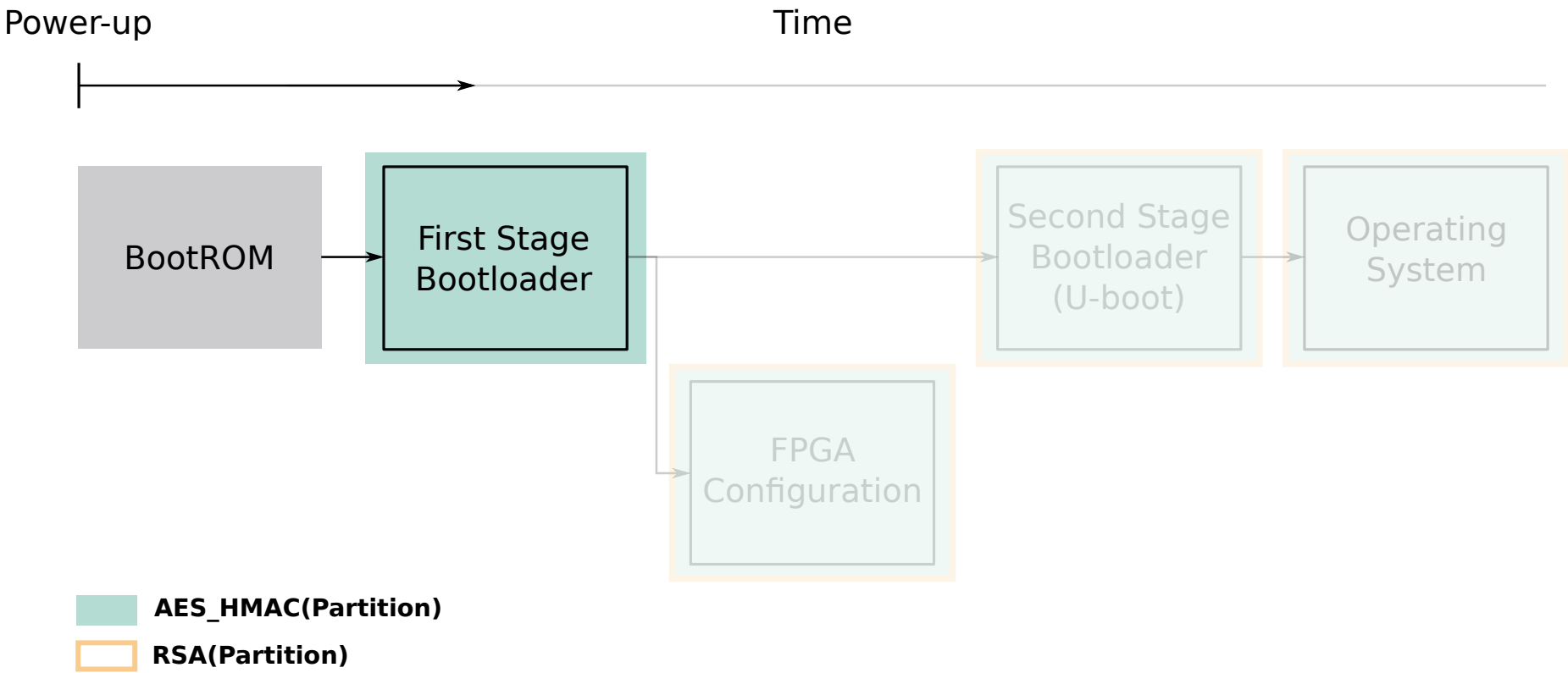
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



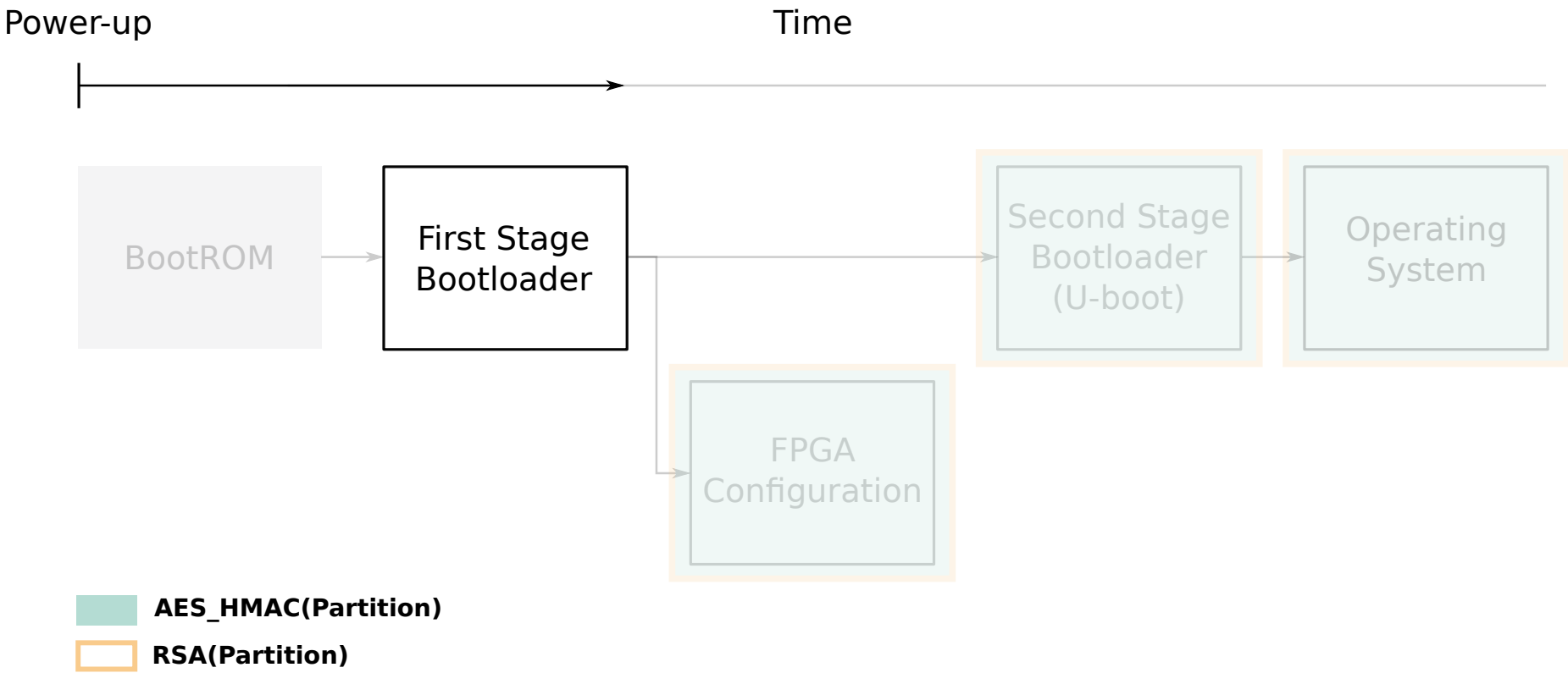
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



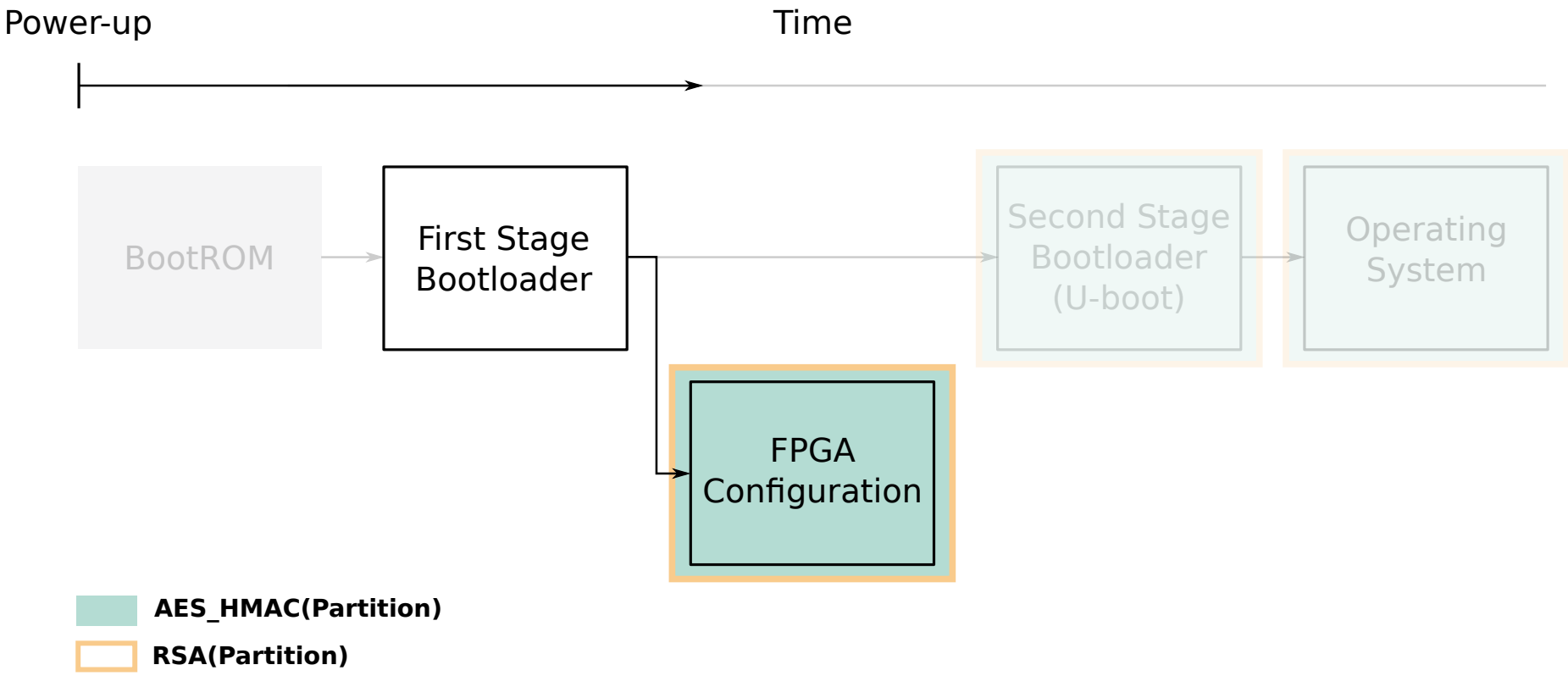
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



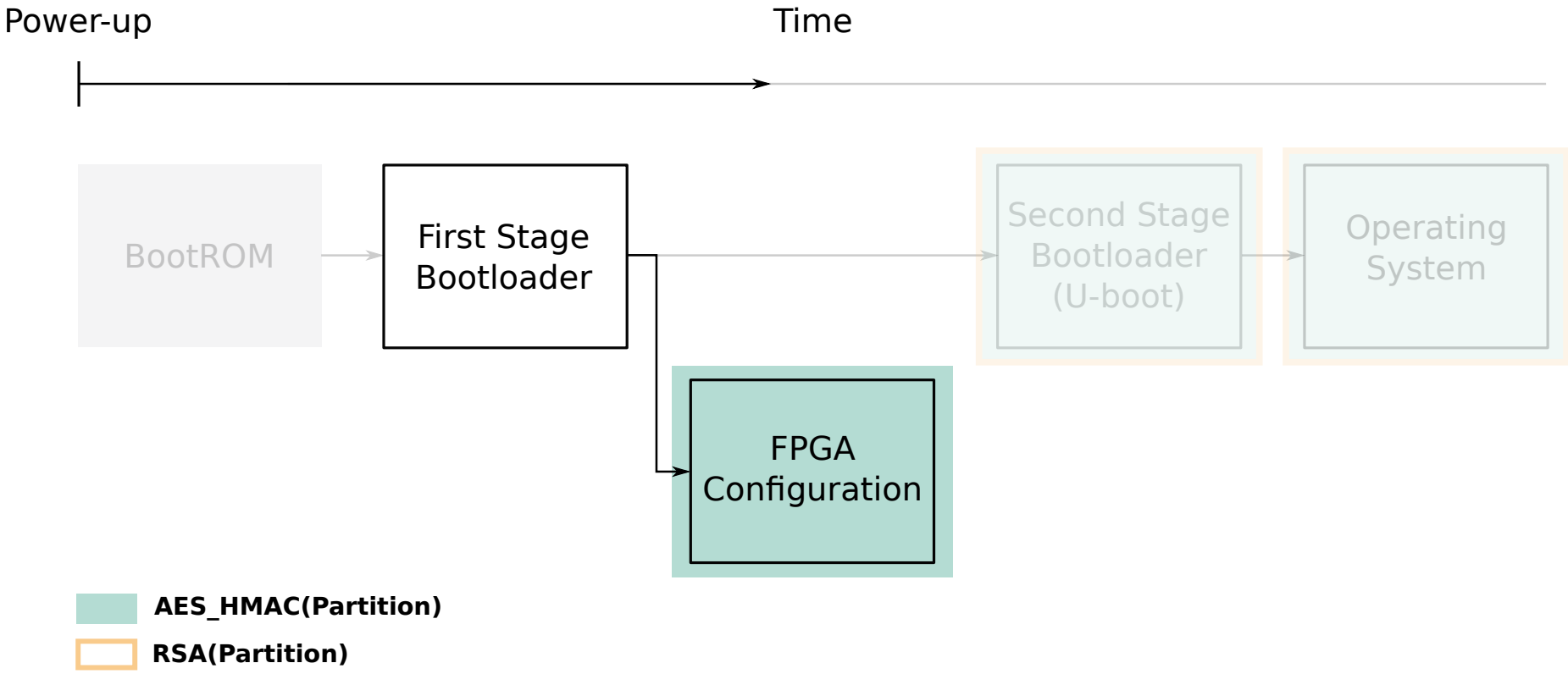
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



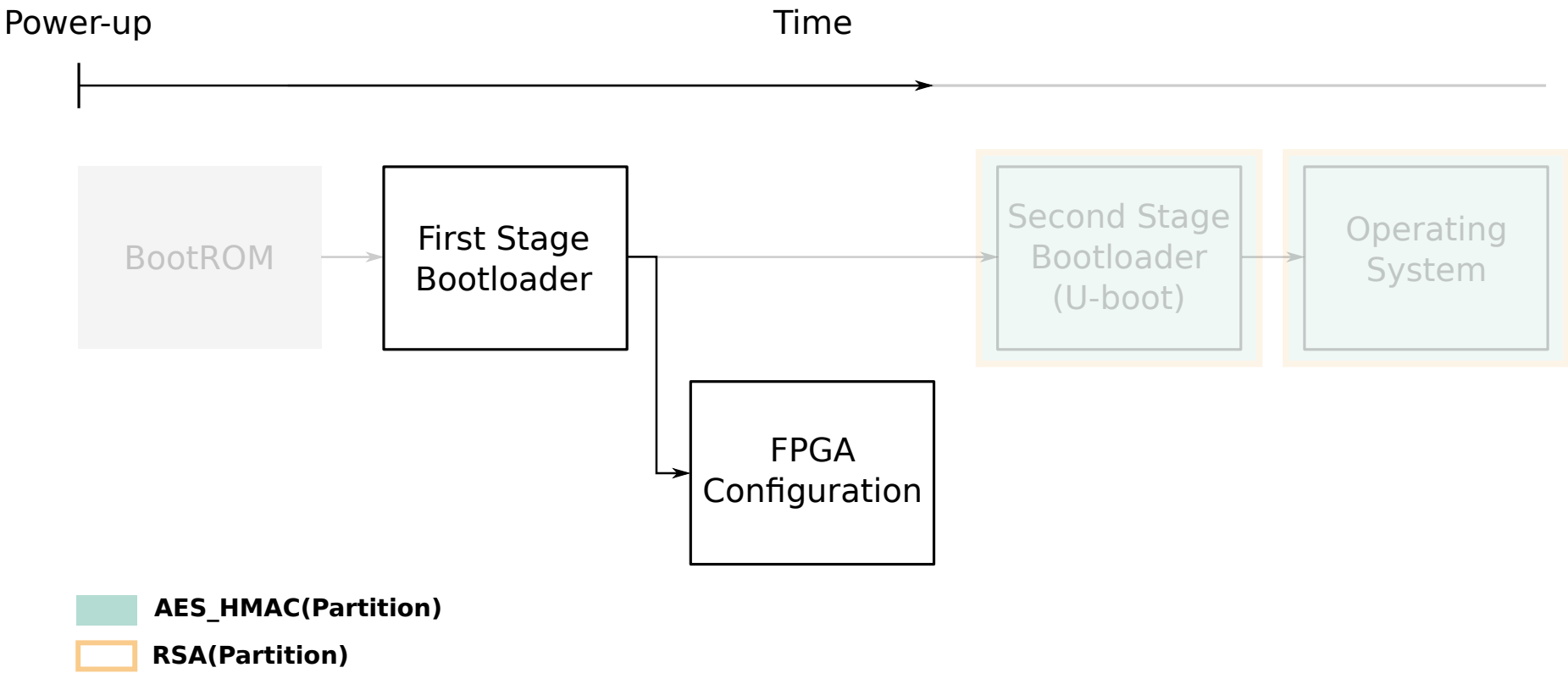
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



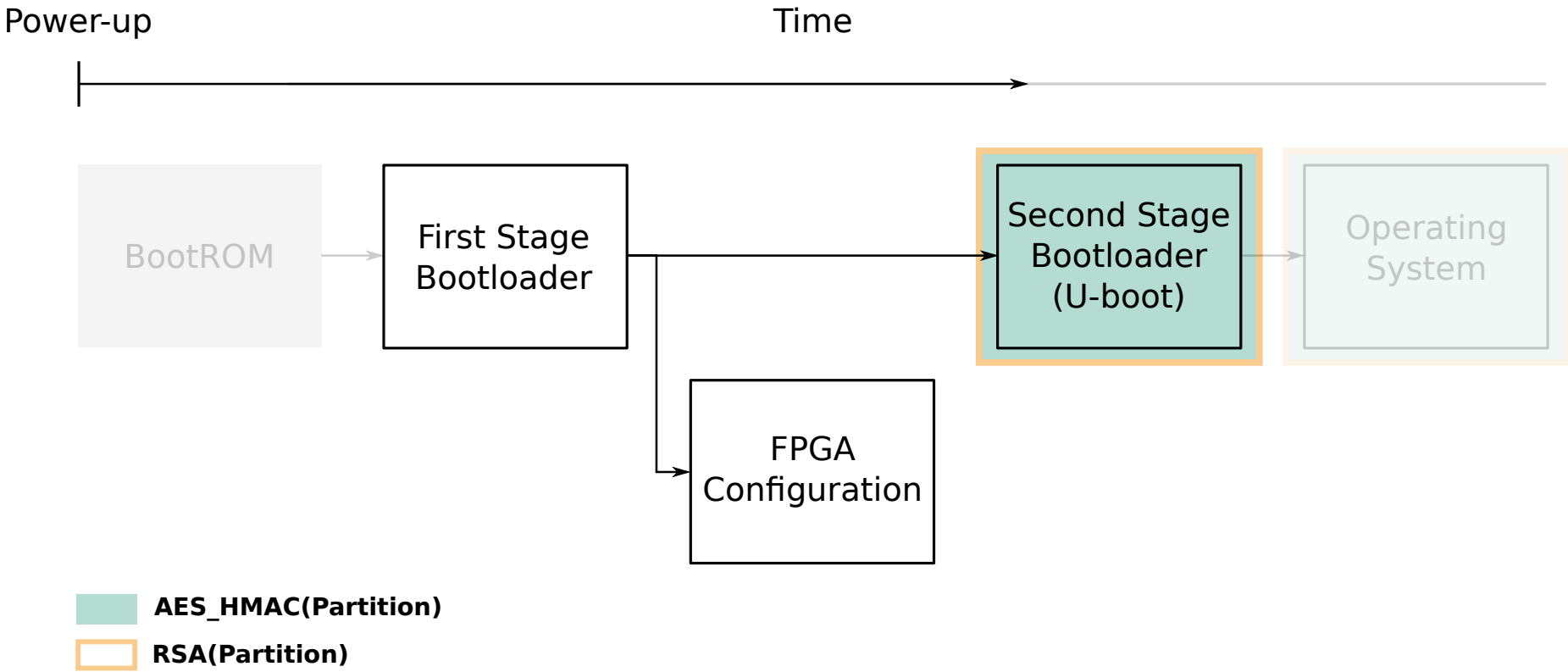
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



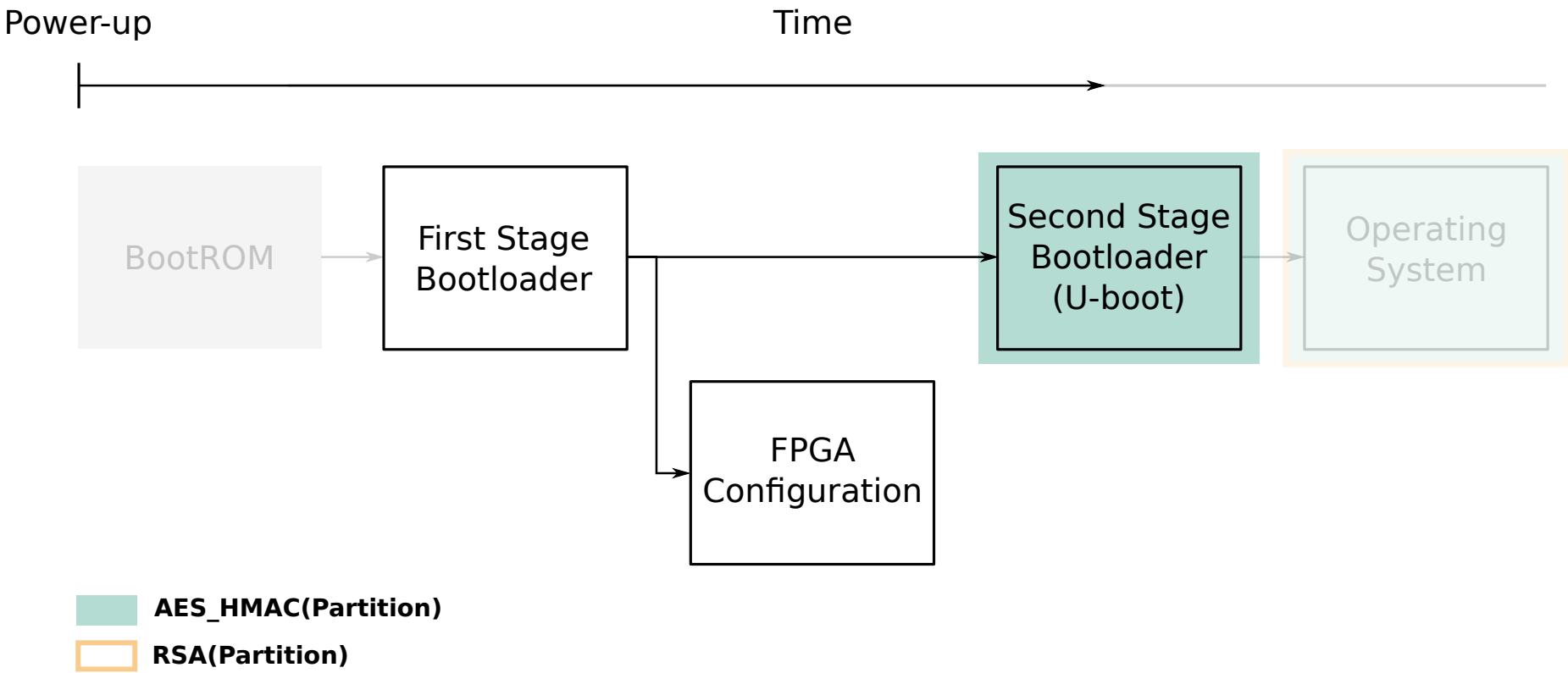
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



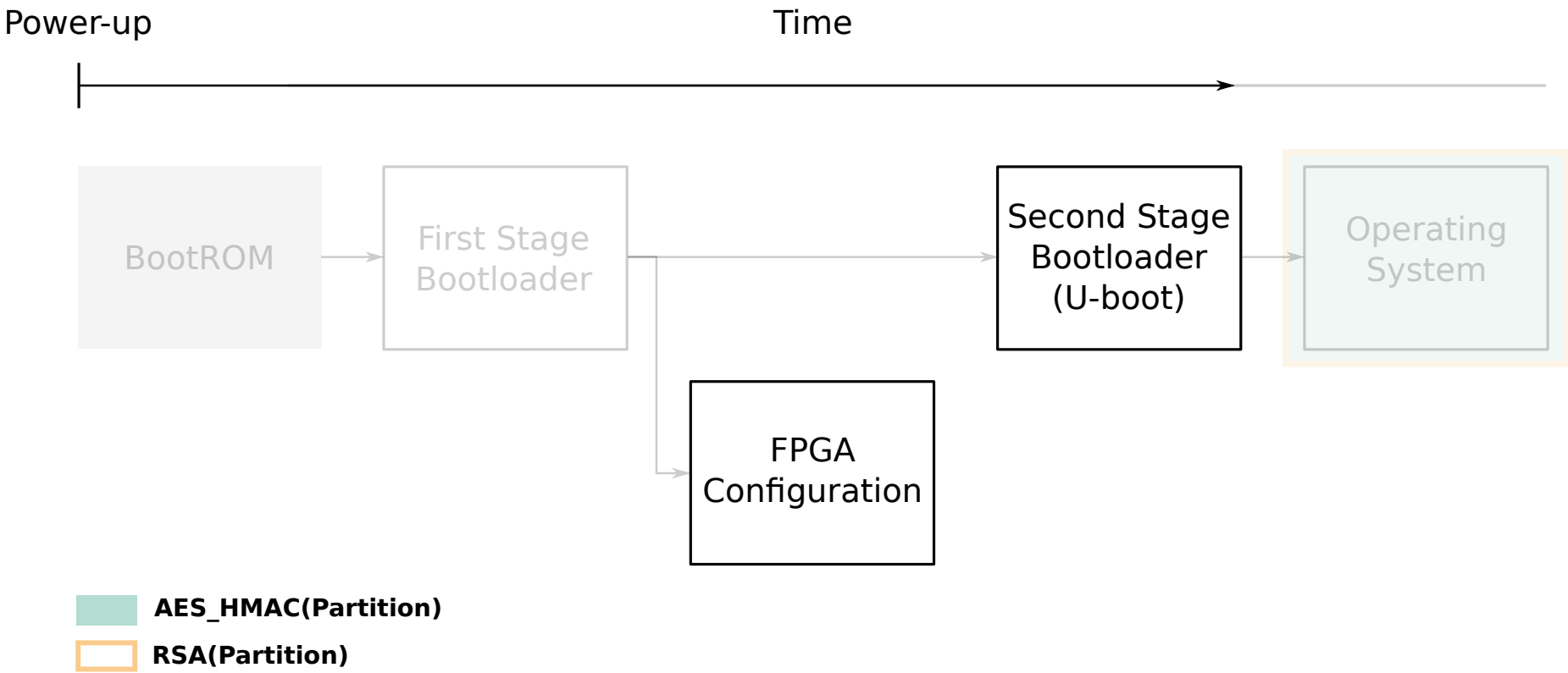
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



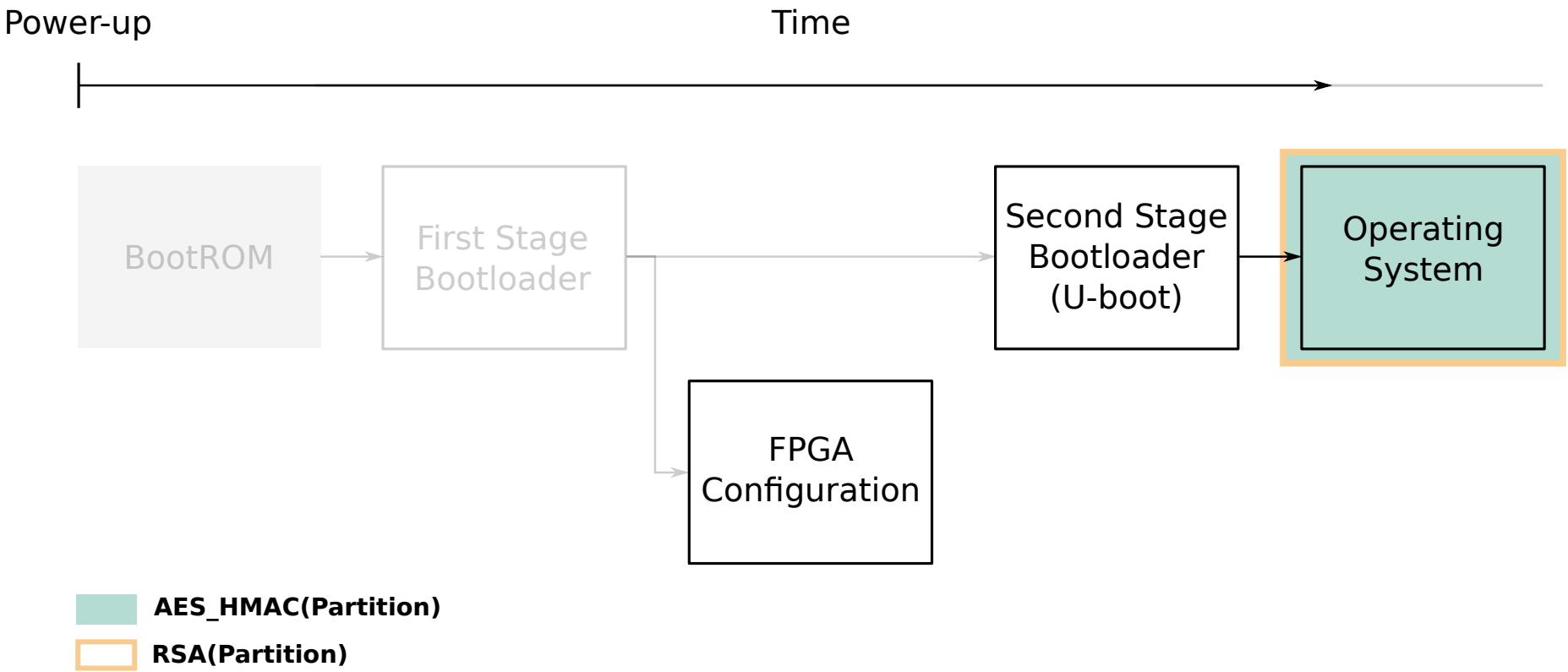
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



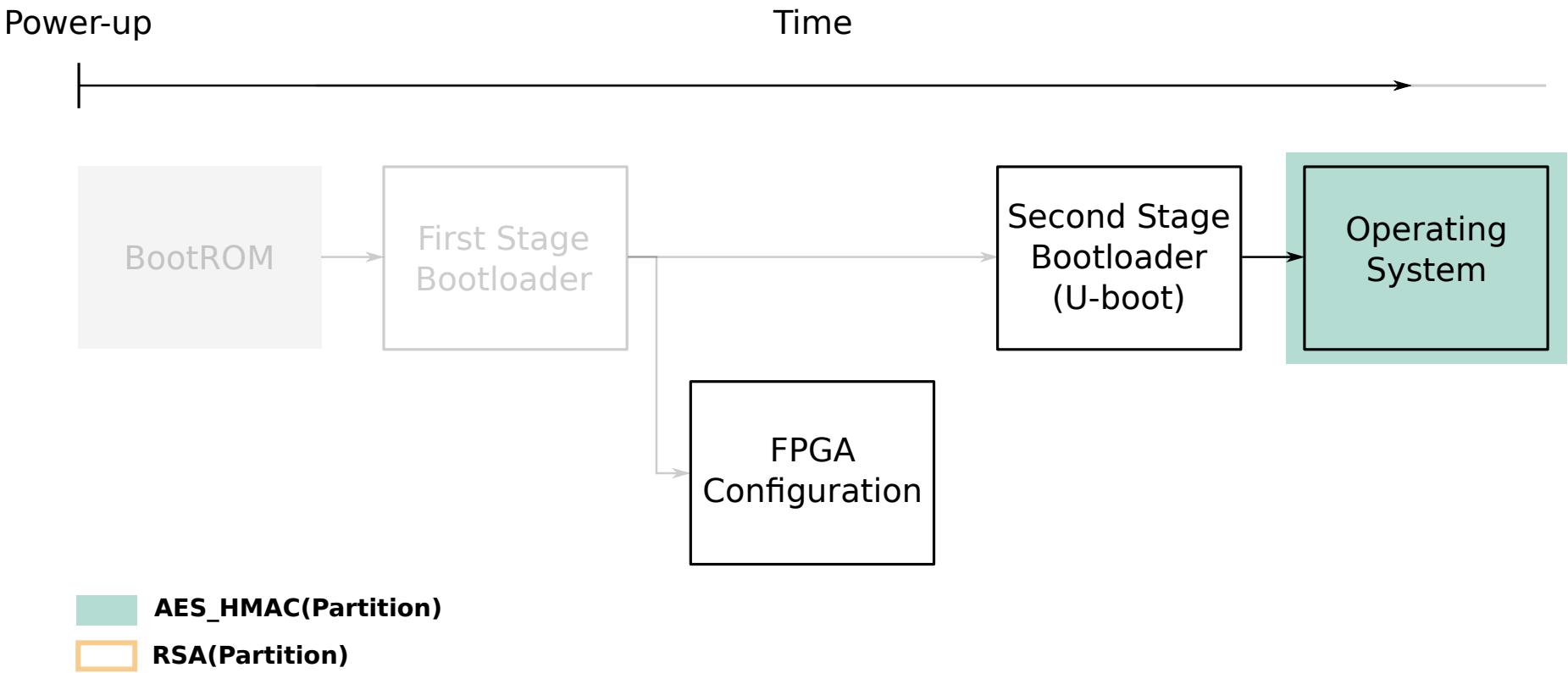
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



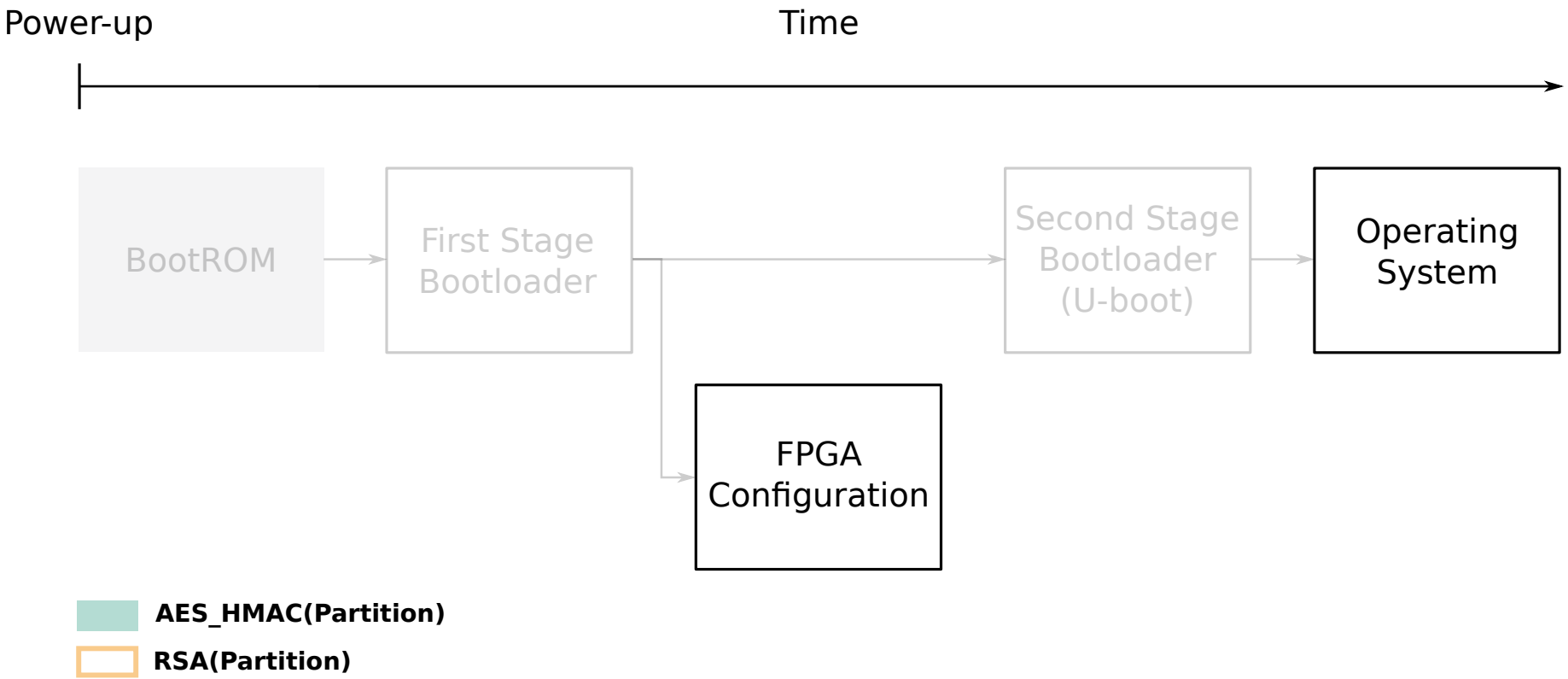
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



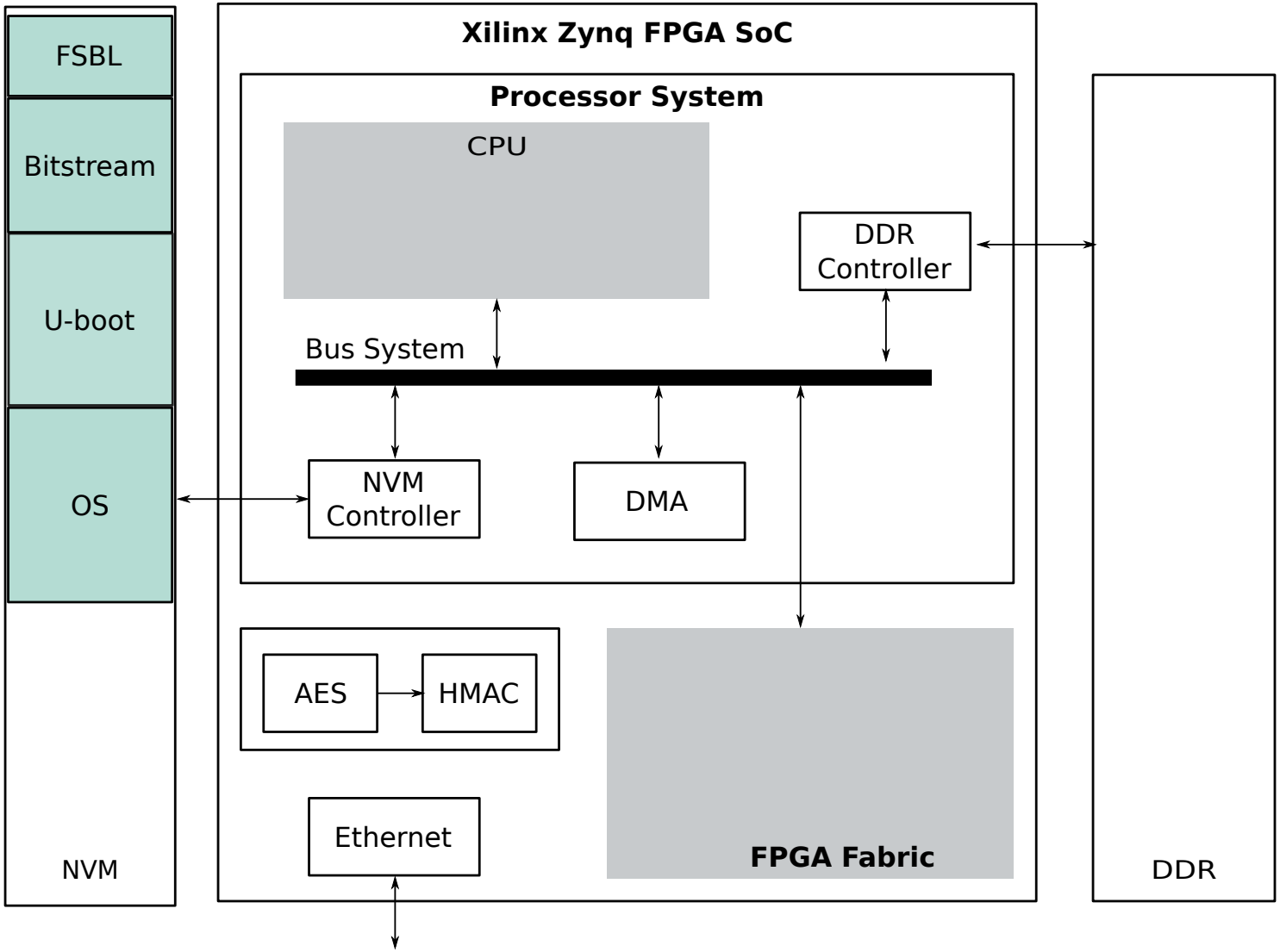
Fraunhofer AISEC

Secure boot on Xilinx Zynq-7000



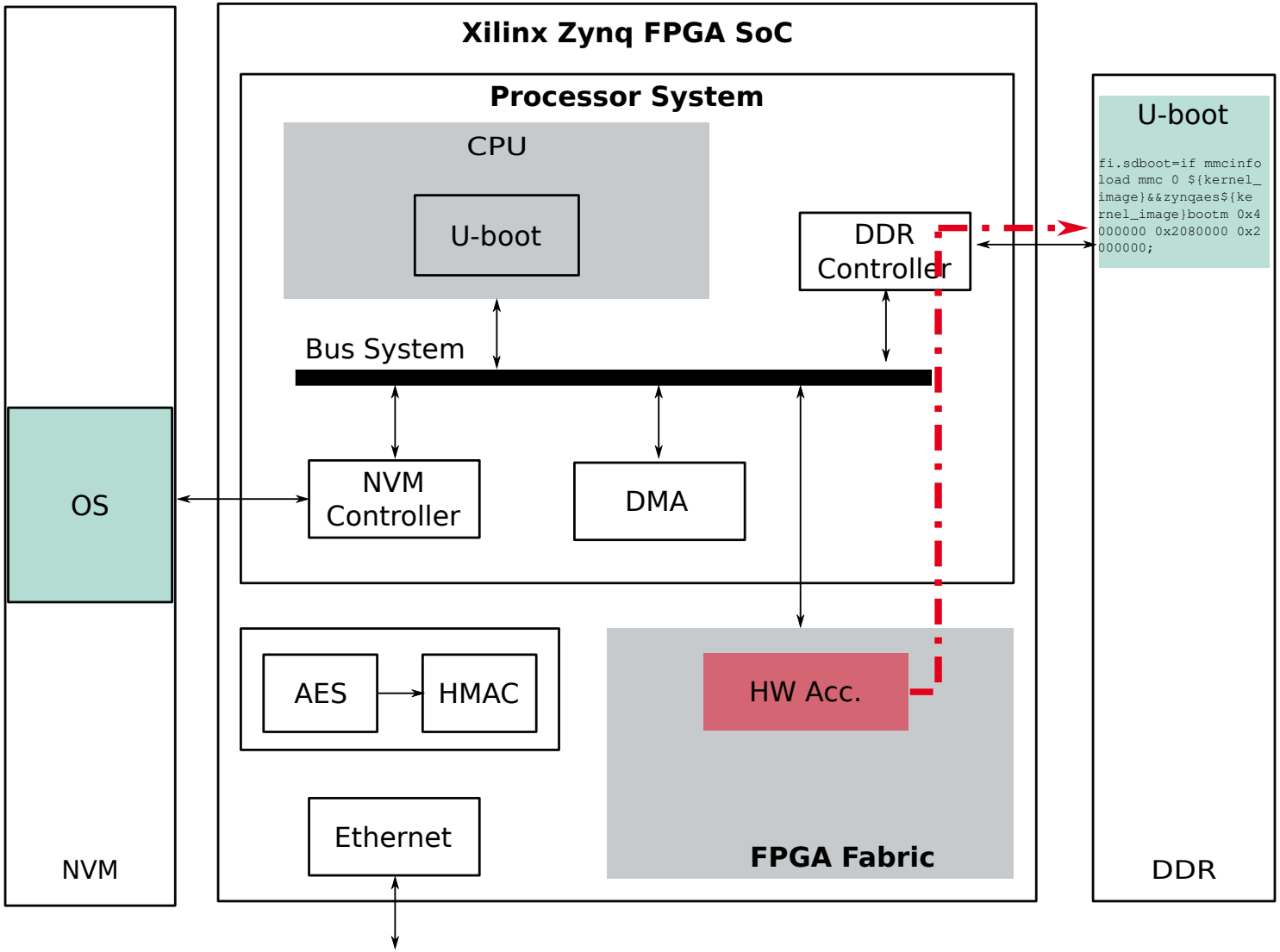
Fraunhofer AISEC

Breaking secure boot on Zynq-7000



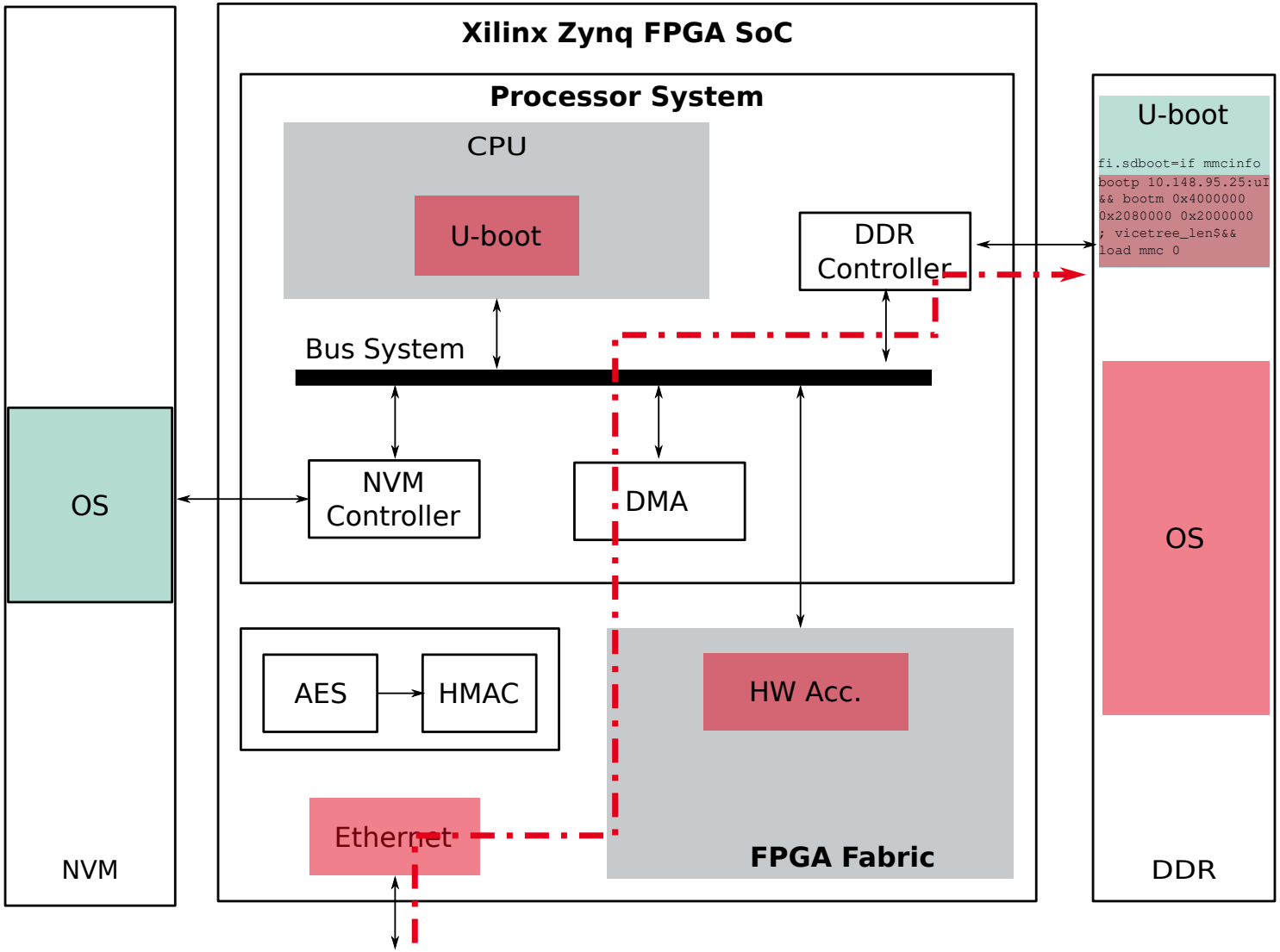
Fraunhofer AISEC

Breaking secure boot on Zynq-7000



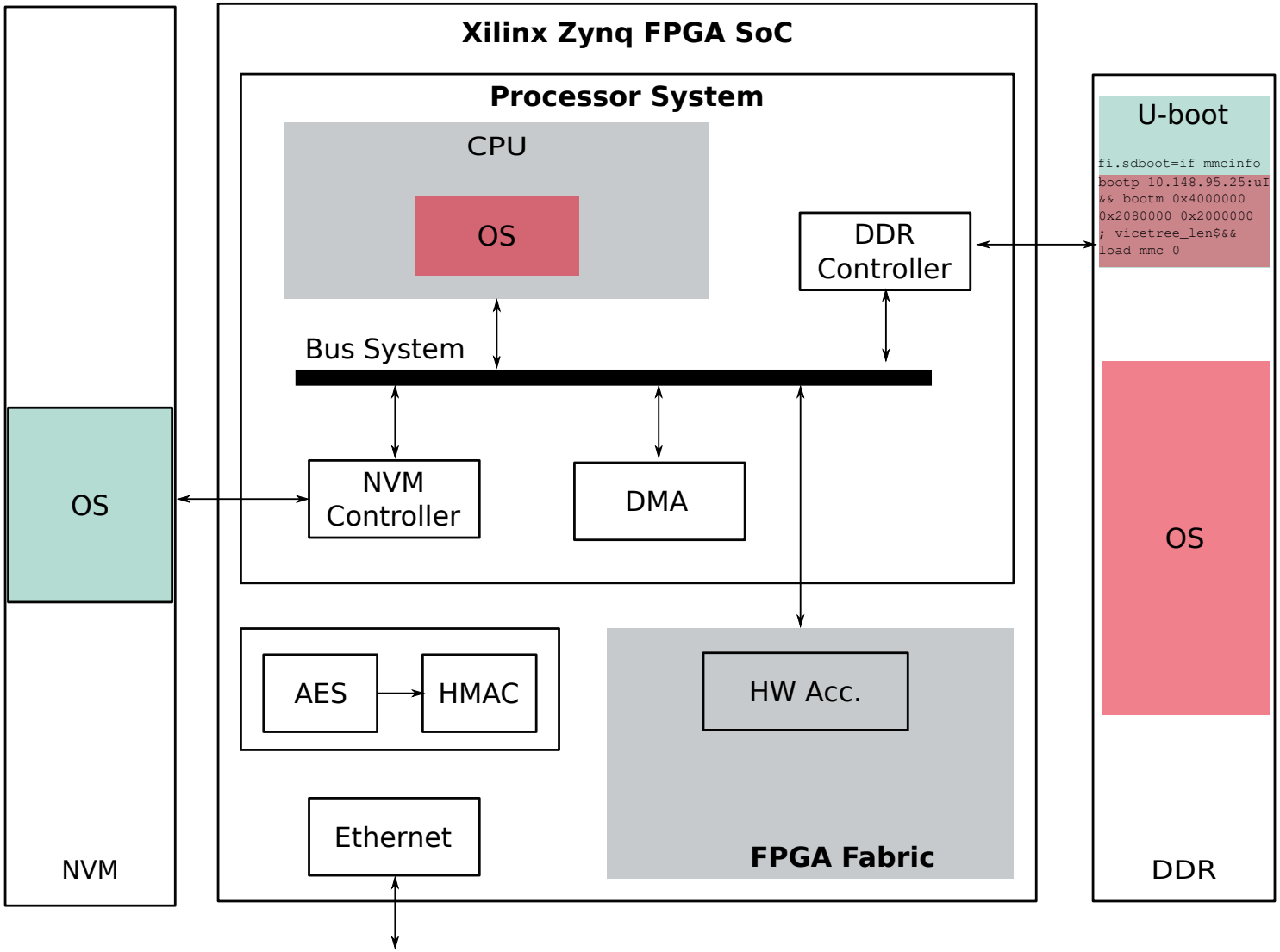
Fraunhofer AISEC

Breaking secure boot on Zynq-7000



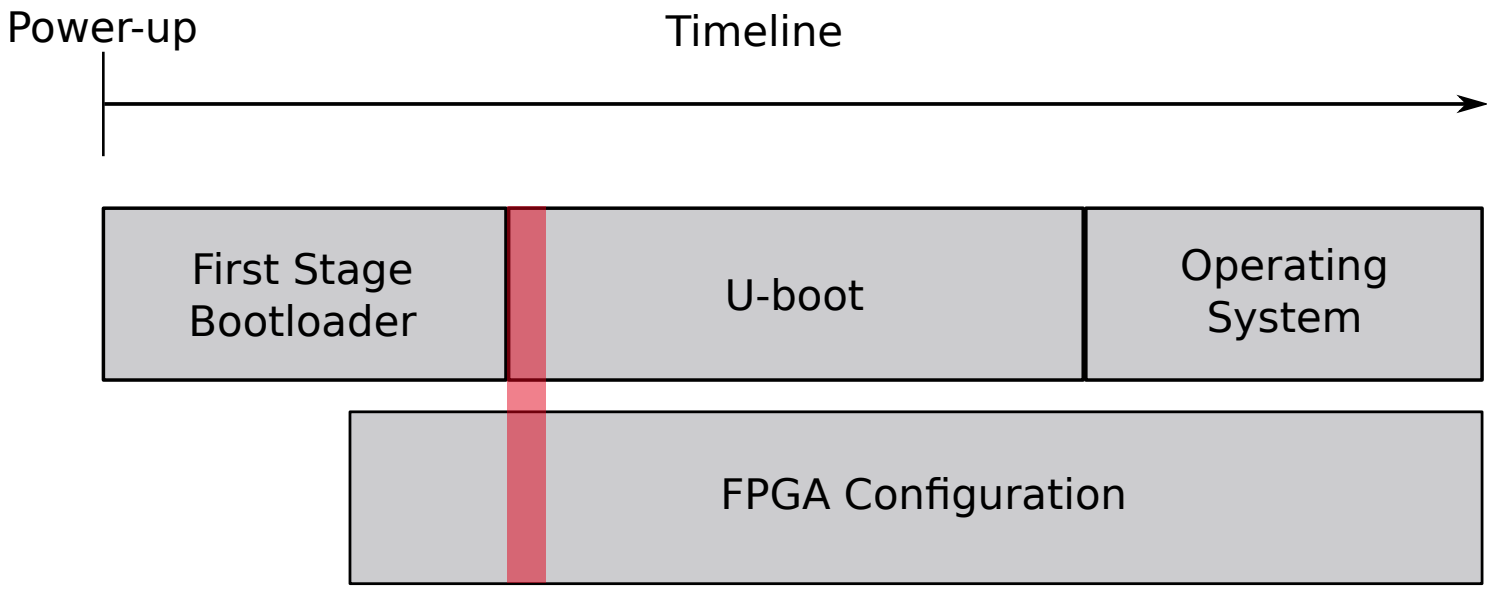
Fraunhofer AISEC

Breaking secure boot on Zynq-7000



Fraunhofer AISEC

Attack timeline



Fraunhofer AISEC

How to break secure boot of FPGA SoCs, ...

... generalise ...

impact on common security mechanisms

Why existing mechanisms do not help

- IOMMU
  - Prevent unauthorised accesses to memory by peripherals
  - Typically **initialized by OS**
  - **Older generations have no IOMMU**

Why existing mechanisms do not help

- IOMMU
  - Prevent unauthorised accesses to memory by peripherals
  - Typically **initialized by OS**
  - **Older generations have no IOMMU**

- TrustZone
  - Prevents unauthorised access to secure world
  - Crypto cores are typically within Trustzone
    - **Whole system can be corrupted**
  - Since U-boot runs in normal world
    - **Normal world IP cores can still carry out the attack**

generalisation to other FPGA SoCs

Impact on different FPGA SoCs

- Xilinx Zynq UltraScale+

  → XMPU from Xilinx allows dynamic memory restriction from early boot stages

Impact on different FPGA SoCs

- Xilinx Zynq UltraScale+

  → XMPU from Xilinx allows dynamic memory restriction from early boot stages

- Altera allows different boot options

  → Depends whether FPGA boots before the OS or during the CPU
  Often FPGA is needed early (e.g. for acceleration)

Impact on different FPGA SoCs

- Xilinx Zynq UltraScale+

  → XMPU from Xilinx allows dynamic memory restriction from early boot stages

- Altera allows different boot options

  → Depends whether FPGA boots before the OS or during the CPU
  Often FPGA is needed early (e.g. for acceleration)

- Microsemi uses non-volatile FPGA

  → Threat is imminent from the very start

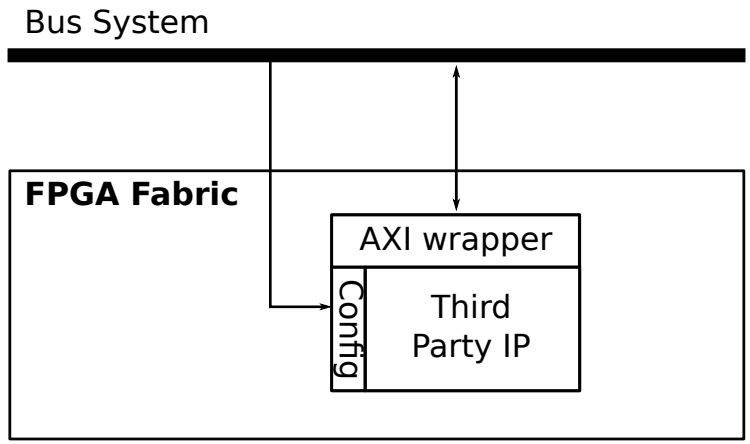
How to break secure boot of FPGA SoCs, ...

... generalise ...

... and protect against such threats

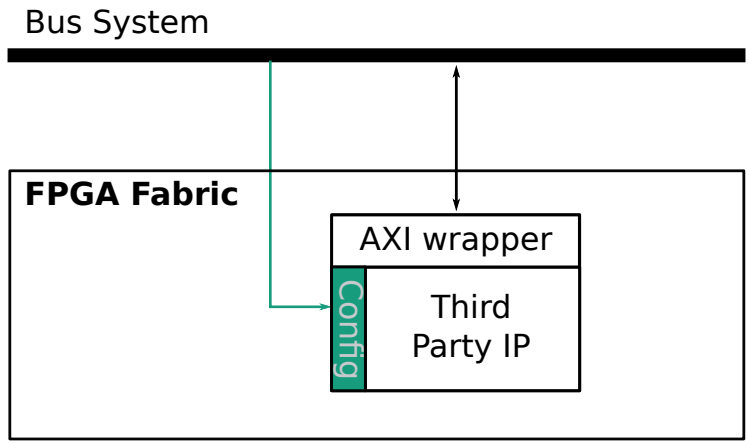
using a security enhanced AXI-wrapper

Wrapper functionality



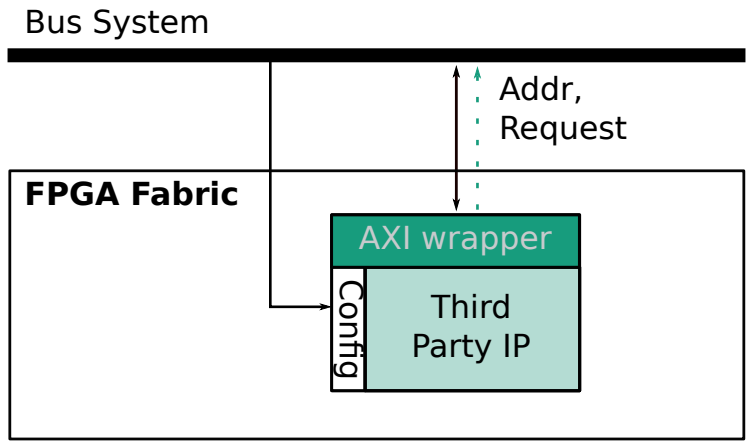
- Does the following checks
  - Address location being accessed during read/write
  - Accesses when system is in idle state
  - Number of transactions being executed
  - TrustZone setting

Wrapper functionality



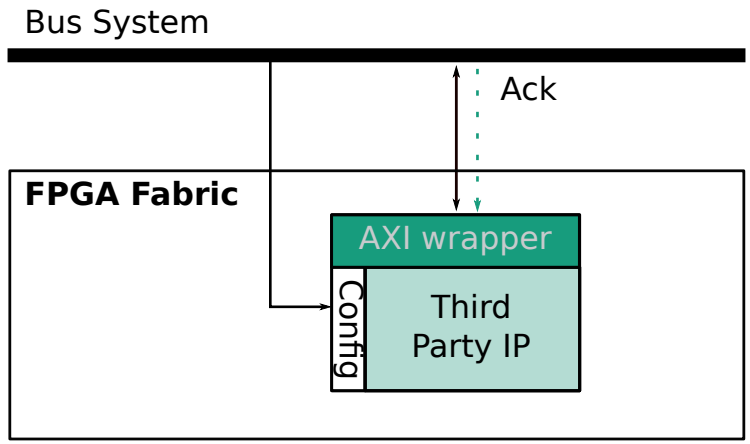
- Uses configuration information sent by software
  - Source address, destination address, length, enable, TrustZone setting

Wrapper functionality

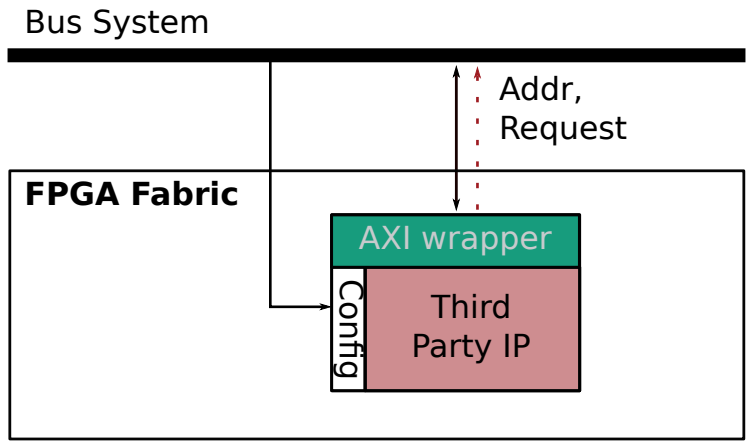


- Checks are interleaved between AXI handshaking

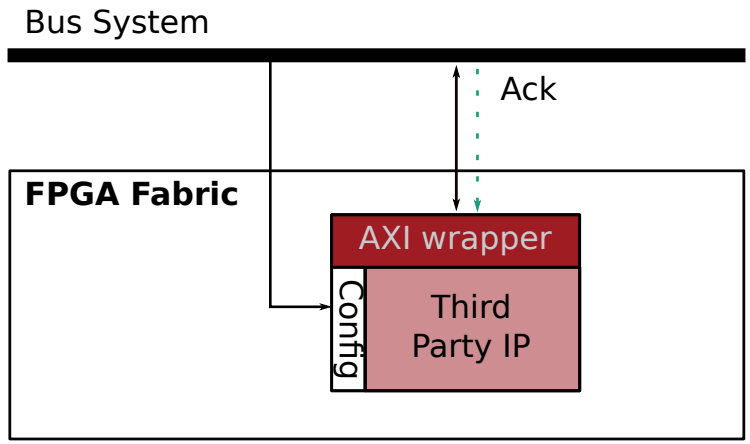
Wrapper functionality



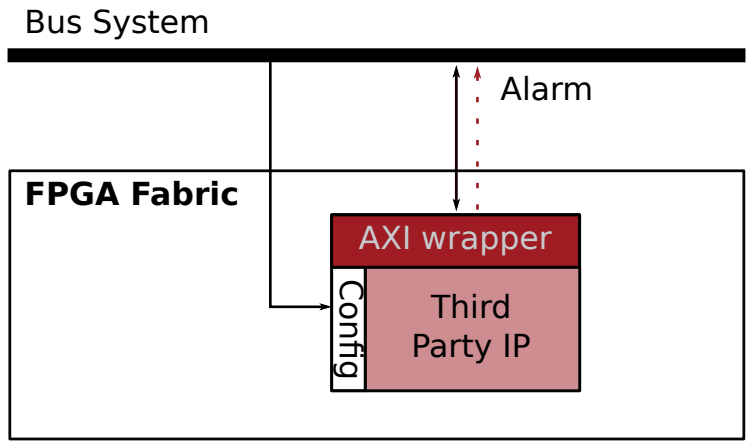
Wrapper functionality



Wrapper functionality



Wrapper functionality



Benefits of our solution

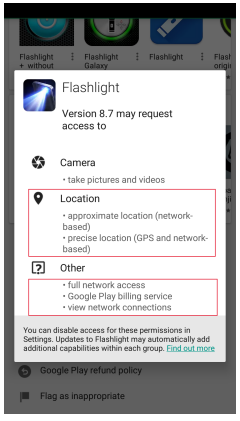
- Interleaves checks between the AXI-handshake
  - Does not affect the performance
- Minimalist functionality and small code base
- Support easy review and re-use
- Functional from power-up and does not rely on OS

- Sources can be downloaded from:

  `https://github.com/Fraunhofer-AISEC/axi-firewall`

Take-home-message

- **System security is threatened** through malicious hardware
- Especially on FPGA SoCs, many HW cores will be sourced from third parties

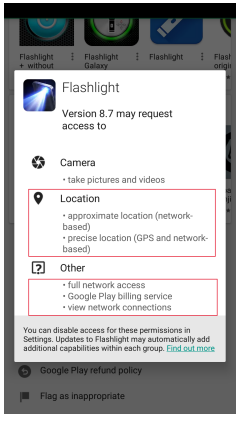


Remember! Hardware cores can also include additional functionality

- This paper shows **compromise of secure boot**
- At DATE 2017, we showed threat to crypto keys in running systems

Take-home-message

- **System security is threatened** through malicious hardware
- Especially on FPGA SoCs, many HW cores will be sourced from third parties



Remember! Hardware cores can also include additional functionality

- This paper shows **compromise of secure boot**
- At DATE 2017, we showed threat to crypto keys in running systems
- Take care of isolating such hardware cores

  → Use simple wrapper or XMPU etc

Nisha Jacob

nisha.jacob@aisec.fraunhofer.de

Hardware Security Department

Fraunhofer-Institute for
Applied and Integrated Security (AISEC)

Garching (near Munich)
Germany

http://www.aisec.fraunhofer.de

Fraunhofer AISEC