Summary

The secure boot functionality for the Zynq®-7000 All Programmable (AP) SoC provides the capability to authenticate all partitions loaded at boot using RSA-2048 authentication. It also supports advanced encryption standard (AES) encryption of partitions that need confidentiality. The Zynq-7000 AP SoC immutable BootROM includes security functions to provide a hardware root of trust (HROT) to protect against early load attacks.

This application note discusses a method to add measured boot capability to Zynq-7000 AP SoCs used in a connected environment. A server provides remote attestation that the embedded systems boot with trusted software over a secure network. The method uses a trusted platform module (TPM) to enhance the HROT functionality. The TPM provides cryptographic functions in a cost-effective, tamper-resistant device which are an effective complement to Zynq-7000 SoC security functions.

Download the reference design files for this application note from the Xilinx website.

Introduction

In most current applications, Xilinx FPGAs and SoCs are programmed once at the factory and often not reconfigured for the life cycle of the device. A method to add functionality and/or reduce the total cost of ownership (TCO) of an embedded system is to support field updates. In Zynq-7000 AP SoCs, the SoC and programmable logic can be updated, so field updates can be very effective. Field updates are typically done over the Internet, which opens up attacks on an embedded system to anyone with network access. Measured boot and network security are critical in firmware updates.

Figure 1 shows an example system environment that uses measured boot. A server manages the software load, update, and validation of fielded, embedded systems based on the Zynq-7000 AP SoC. The embedded systems connect to the server using Ethernet. In addition to updating software on the embedded systems, the server verifies that the correct, trusted software is loaded. This verification by the server, done at boot and run time, is remote attestation.
Remote attestation capability has been in Linux starting with 2.6.3, and is generally known as integrity measurement architecture (IMA). The Linux extended verification module (EVM) is used in conjunction with IMA. The term measured boot is used because the client returns a value, typically a secure hash algorithm-1 (SHA-1) digest, of each of the partitions loaded. In remote attestation, the server compares the measured logs with known good measurements.

The attestation server knows the characteristics (measurements) of the partitions loaded on the embedded systems, including partition size and digests. At load, the embedded systems send log files to the server containing partition measurements. The server verifies the measurement, and if a client loads software that is different than what is expected, the server executes a policy set up by the server administrator. A policy is a set of actions taken by the server based on measurement results. Policies in the reference design include Allowed, Guaranteed, Blocked, and Isolated.

An example of a policy is to keep the embedded system off the network, update the software, re-run remote attestation, and allow the client to connect to the network if the software can be trusted. Isolating a corrupted embedded system from the network limits its ability to corrupt other embedded systems. This is a typical policy of a server in remote attestation, not the only policy, because the policy is generally defined by the application.

Measured boot is done in addition to, not in place of, secure boot. Measured boot does not prevent malicious software from being loaded. The TPM enhances the HROT and increases the security of the software load/update process. The TPM is placed on the same board as the Zynq-7000 AP SoC. A device ID is associated with the Zynq-7000 SoC-TPM platform. The TPM provides cryptographic functions used in measure boot. The HROT is enhanced with the TPM because an adversary has to defeat both the Zynq-7000 AP SoC and the tamper-resistant TPM for a successful attack.
Figure 2 shows functional components of the Zynq-7000 AP SoC and the Infineon OPTIGA SLB 9670 TPM on the client platform.

At power-up, the Zynq-7000 AP SoC on-chip BootROM code loads the first stage boot loader (FSBL). The FSBL loads U-boot, and U-Boot loads the Linux kernel, root file system, device tree, and Linux application software. In one approach to booting with a chain of trust, the BootROM authenticates/measures the FSBL, the FSBL authenticates/measures U-Boot, and U-Boot authenticates/measures the Linux partitions.

The SHA-1 measurement logs are stored in the TPM platform configuration registers (PCRs). Measurements of the BootROM and the FSBL are done by the FSBL and placed in the PCRs using a serial peripheral interface (SPI) connection. The measurements are transmitted to the server for remote attestation. The TPM cryptographically signs the SHA-1 values in PCRs so that partition measurements are not transmitted from the embedded system in plain text.

For remote attestation of firmware updates, the network connection between the attestation server and clients must be secure. IPsec functionality, including a privacy certificate authority (CA) that generates X.509 certificates, implements the transport layer security (TLS) handshake between the server and client(s). The network security used in the measured boot reference design is discussed in Network Security in Measured Boot, page 12.
Hardware and Software Requirements

The hardware and software requirements for the reference system include the following:

- Avnet Industrial Internet of Things (IIoT) starter kit with MicroZed board
- Infineon OPTIGA TPM 1.2 SLB 9670 peripheral module (Pmod) compatible board
- Two USB type-A to USB mini-B cables (for UART and JTAG communication)
- Micro Secure Digital (microSD) memory card (16 GB)
- Ethernet cable
- Xilinx Software Development Kit 2017.1
- Xilinx Vivado® Design Suite 2017.1 (optional)
- Wind River Pulsar Linux 8.0
- VirtualBox 5.0.26 or higher (or VMware equivalent)
- Ubuntu or Ubuntu command line virtual machine image in Open Virtual Appliance format
- strongSwan TNC software
- Tera Term or equivalent communication terminal software

Depending on the system setup, all of the software listed above might not be necessary.
Reference System Description

Figure 3 shows a desktop setup for the single client system used in the reference design. The client in the Avnet IIoT drives a communication terminal. The strongSwan attestation server runs from VirtualBox. A browser is used to view the measurements and the implementation of the policy.
Figure 4 shows the Avnet IIoT starter kit with the MicroZed board mounted on the Avnet Arduino carrier card. It also shows the Infineon OPTIGA TPM 1.2 SLB 9670 Pmod plugged into the J2 PS PMOD connector.

The Infineon OPTIGA SLB 9760 attached to PMOD connector J2 communicates to the Zynq-7000 AP SoC using the processing system (PS) serial peripheral interface (SPI) driver. Wind River Pulsar Linux (WRPL) 8.0 runs on MicroZed and includes the strongSwan client software. Prior to booting WRPL, the Zynq-7000 AP SoC runs the FSBL. The FSBL runs pre-boot authentication on the BootROM and FSBL.

The FSBL then executes PCR extend commands to log BootROM and FSBL SHA-1 digests into the PCR[0] and PCR[4]. The PCR extend command is similar to a write command. It differs from the write value in that the value loaded into the PCR is an SHA of all cumulative SHA values loaded into the PCR. Because a PCR extend rather than a write is done, an adversary is unable to read PCRs and get “expected” values.
Figure 5 shows three possible system setups. The strongSwan software running on an Ubuntu server does both the remote attestation and the network security. The Ethernet connection can be either direct from the PC to the Avnet board or dynamic host configuration protocol (DHCP) using Ethernet wall sockets. In (a), the server is run on an Ubuntu installation on either VirtualBox or VMware on the PC. In (b), an Ubuntu-based PC runs the strongSwan server. In (c), the server runs on an Amazon web server (AWS). Xilinx does not provide the AWS account.

The reference system provides methods for a measured boot of a single client, providing remote attestation of early load software (BootROM, FSBL) which is new functionality for embedded systems. An actual connected system has multiple embedded devices, and the strongSwan server measures all Linux partitions loaded, not just the early load software.

**Hardware Root of Trust**

In Zynq-7000 AP SoCs, the HROT is based on the first code executed by the ARM® CPU0 at power-on. The code is stored in on-chip, metal-masked ROM, and is referred to as BootROM code. BootROM code is immutable, and its principle function is to perform device initialization and load the FSBL into read/writable on-chip memory (OCM). Neither the BootROM nor the OCM are accessible at device pins. The BootROM Configuration Flowchart figure in *Zynq-7000 All Programmable SoC Technical Reference Manual* (UG585) [Ref 1] provides the flow of the BootROM code functionality. If secure boot is specified, the BootROM authenticates the FSBL using the RSA-2048 standard prior to execution of the FSBL. The Zynq-7000 AP SoC HROT is enhanced by adding a TPM to the embedded platform. The TPM provides partition measurements, cryptographic functions, and secure key storage for keys used by the Zynq-7000 AP SoC.
In Zynq-7000 AP SoCs, the term **secure boot** is used to define the secure loading of the bitstream and software at power-on. The bitstream is loaded into on-chip configuration memory. Software partitions encrypted in non-volatile memory (NVM) are generally authenticated, decrypted, and copied to DDR memory. As shown in Figure 6, the RSA-2048 authentication uses a chain of trust sequentially on each partition loaded. Each partition can use its own private/public key pair. Partitions can optionally be authenticated using keyed-hash message authentication code (HMAC). In its simplest form, all partitions loaded by the FSBL are authenticated using RSA-2048 from the XILRSA library. In an alternative boot flow, U-Boot has access to the XILRSA library. In secure boot, if either RSA or HMAC authentication fails, the Zynq-7000 AP SoC transitions to a lockdown state.

A malicious actor needs to steal the RSA private key, which is not stored in the device, and the HMAC key. To protect against an insider attack, the SDK Bootgen key management can be split so that independent parties handle RSA, AES, and HMAC keys.

---

Figure 6: Zynq-7000 AP SoC Chain of Trust Boot

---

**Measured Boot**

Measured boot is recommended when embedded systems are connected to a network. Secure boot should still be used in systems which use measured boot. Secure boot and measured boot functionality are complementary. Connecting embedded systems to a network provides a method for firmware updates. Embedded systems connected to a network have a wider attack surface than closed systems. Hackers with network access are a common security threat. Remote attestation addresses this vulnerability during boot and run time. In secure and measured boot, all files/partitions are authenticated and measured.

The basics of measured boot are discussed in Integrity Measurement Architecture. The measured boot in the reference design uses a TPM for added security. Trusted Platform Module, Zynq 7000 SoC-TPM Interface, and Network Security in Measured Boot discuss measured boot using a TPM. Secure boot and measured boot do not use programmable logic resources, so the Zynq-7000 AP SoC unit cost is not affected when measured boot is used.

Using measurements, an attestation server can periodically execute run-time integrity checks on clients, and execute a policy based on the results. This is important in connected systems because the probability of an attack during run time is high. The ability to implement a policy is an improvement over the runtime integrity checker (RTIC) described in Run Time Integrity and Authentication Check of Zynq-7000 AP SoC System Memory (XAPP1225) [Ref 2].
IMA provides the basis for measured boot. An overview of IMA is provided in strongSwan documentation [Ref 3].

In remote attestation with IMA, the server compares measurements received from clients with reference integrity measurements (RIMs) and acts according to a predefined policy. In the reference design, this is referred to as the policy decision point (PDP). After running measured boot, a server website provides a summary of measurements and policy for the reference design.

In an IMA implementation, the client runs integrity measurement collection (IMC). The server runs integritymeasurement verification (IMV). Figure 7 shows an example log when all Linux partitions are measured.

```bash
ls /sys/kernel/security/ima
ascii_runtime_measurements_binary_runtime_measurements runtime_measurements_count violations
sado cat /sys/kernel/security/ima/runtime_measurements_count

sado less /sys/kernel/security/ima/ascii_runtime_measurements
```

_X-Ref Target - Figure 7_

**Figure 7:** Integrity Measurement Architecture Evidence Log
Documentation on TPM functionality is provided by the Trusted Computing Group (TCG), beginning with the TPM Main Specification [Ref 4]. TPM 1.2 was the most commonly used TPM in 2016. The Infineon OPTIGA SLB9670 TPM supports TPM 1.2 and 2.0. TPMs are very small, cost-efficient devices that provide root of trust for reporting (RTR) and root of trust for storage (RTS) security. This application note focuses on the RTR in which the measurement log file held in the TPM’s PCRs is reported to the server.

In addition to support for RTR, TPMs provide capability that might be useful in Zynq-7000 SoC applications. TPMs provide re-programmable non-volatile memory. The TPM hardened cryptographic functions allow a key to be securely transmitted to the Zynq-7000 device on demand. TPMs provide a random number generator (RNG). RNGs can be used to generate keys.

The TPM RTR support operates within the IMA framework, providing significant security enhancements. When a TPM is added, the server’s remote attestation of a client is based on a quote. A quote is measurement or evidence on the partitions booted. In TPM 1.2, an SHA-1 digest is used as the measurement for partitions loaded. In TPM 2.0, an SHA-2 digest is used as the measurement log for partitions loaded. The SHA digests are stored in the PCRs. Figure 8 shows the server-client communication for remote attestation.

The flow in Figure 8 is outlined below.

1. The strongSwan attestation server requests a quote from the client. When requesting the quote, the server sends a nonce, which is a random number used to protect against playback attacks.

2. The client or Zynq-7000 SoC/TPM generates the evidence for the partitions loaded. The SHA-1 hashes are stored in the TPM PCRs. The SHA-1 of the BootROM code is stored in PCR[0], and the SHA-1 digest of the FSBL is stored in PCR[4].

3. The Zynq-7000 SoC/TPM client sends the quote to the server. This includes signed evidence and includes the original nonce.

4. The strongSwan server appraises the quote and, based on the results, follows a policy setup by the system administrator.
Zynq 7000 SoC-TPM Interface

The Zynq-7000 SoC-TPM interface provides the communication between the Zynq-7000 device and the Infineon OPTIGA SLB9670 TPM. The interface uses commands from a tpm_toolbox. The tpm_toolbox supports the following categories of commands:

- PCR reset
- Physical presence
- Get capability
- TPM startup/activate/physical enable
- PCR read/PCR extend

There are multiple commands in each category. A subset of the commands is used in the reference design. The Zynq-7000 AP SoC connects to the SLB9670 TPM using the SPI bus. The Zynq-7000 AP SoC contains a hardened SPI IP in the PS and a soft AXI SPI IP in the programmable logic (PL). The PS SPI is used in the reference design because it saves PL resources.

Figure 9 shows SPI-TPM functions implemented in the FSBL for the reference design.

Figure 9: FSBL TPM SPI Driver Functional Diagram

In the measured boot reference design, the FSBL is modified to calculate the SHA-1 of the BootROM and the FSBL, and then extend the SHA-1 digests into the TPM’s PCRs. The SHA-1 values are calculated in shal.c. Code to take ownership and activate the TPM is in slb9670_tpm_spi.c. The PCRs are extended in slb9670_spi_tpm.c. Other files added to fsbl/src include tpm_tools.h, tpm_tools.c, tpm_spi.c, tpm_spi_tis.c, and tpm.h. Because BootROM code is not accessible by the FSBL, the SHA-1 calculated for the BootROM is calculated on the cyclic redundancy check (CRC) written by the BootROM code.

The FSBL TPM driver can be encrypted when stored in NVM and then decrypted and run from OCM. The reason for the FSBL extending the TPM PCRs with early load measurements is to limit the malicious attacker’s time to change the code.

In the Avnet Starter IIoT board, the PS SPI interfaces to the SLB9670 Pmod using an MIO connection. To drive the pin reset of the TPM, the Zynq-7000 AP SoC hardware design includes a PS GPIO which is used to drive the TPM reset pin. The ResetTPM function is in main.c.
Network Security in Measured Boot

Software updates and remote attestation require a secure connection between a server and the embedded system clients. The network has a large attack surface because it can be attacked by any adversary with access to the Internet. For firmware updates, a server to client(s) connection is used. In some factory automation environments, client-to-client communication is also needed to coordinate operational procedures.

Figure 10 shows a high-level view of strongSwan’s implementation of TCG’s trusted network connect (TNC). In the TNC architecture, the server connects to clients in the integrity evaluation layer. The client does integrity measurement collection (IMC), and the server does integrity measurement verification (IMV). The software on the client is the platform trusted service (PTS), trust software stack (Trousers), and the TPM tools.

PTS uses the Trousers library to access the TPM and IMA measurements. The reports use standard PTS formats for interoperability between applications and vendors.

The policy decision point (PDP) defines the action taken by the server after measurement verification. A typical policy/action is to limit network access until remediation is done. A different policy is used when availability is a critical driver, and some out-of-range measurement verifications are not treated as critical. The TPM Main Specification [Ref 4] provides an overview of the TNC architecture.

The underlying security for TNC in the reference design uses IPsec. This includes conventional technology such as Internet key exchange (IKEv2), public key infrastructure (PKI), and the transport layer security (TLS) handshake in which the encryption and authentication algorithms are negotiated and pre-shared keys are exchanged. A virtual private network is set up in the strongSwan architecture. A privacy CA generates the x509 certificates. The strongSwan Readme.txt provides information on the IPsec flow.
Reference Design Functional Overview

The following steps are done in the reference design to set up IMA, TPM, and network security.

• Activate IMA in the Linux kernel
• Configure the IMA policy
• Activate the TPM
• Set up the privacy certificate authority (CA)
• Set up the attestation client (Zynq-7000 AP SoC)
• Generate an attestation identity key (AIK)
• Configure the integrity measurement collectors
• Configure the TNC client
• Configure the VPN connection
• Set up/configure the attestation server (strongSwan VPN/TNC server)
• Collect measurement values
• Register the device with the policy manager

The process is defined on the strongSwan website.

Conclusion

Zynq-7000 AP SoCs provide significant advantages in their ability to program both hardware and software on the same device. Cost-effective firmware updates are a key to increasing embedded system capability and providing maintenance to reduce the TCO. Remote firmware updates rely on using the Internet, opening the embedded system to cryptographic attacks. This application provides mechanisms that provide proven-in-use security for connected devices.

References

2. Run Time Integrity and Authentication Check of Zynq-7000 AP SoC System Memory (XAPP1225)
3. Linux Integrity Measurement Architecture wiki.strongswan.org/projects/strongswan/wiki/IMA
4. TPM Main Specification www.trustedcomputinggroup.org/tpm-main-specification
5. Secure Boot of the Zynq-7000 All Programmable SoC (XAPP1175)

Revision History

The following table shows the revision history for this document.

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>03/07/2017</td>
<td>1.0</td>
<td>Initial Xilinx release.</td>
</tr>
</tbody>
</table>

Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE, and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx’s Terms of Sale which can be viewed at http://www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such applications, please refer to Xilinx’s Terms of Sale which can be viewed at http://www.xilinx.com/legal.htm#tos.

Automotive Applications Disclaimer

XILINX PRODUCTS ARE NOT DESIGNED OR INTENDED TO BE FAIL-SAFE, OR FOR USE IN ANY APPLICATION REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS APPLICATIONS RELATED TO: (I) THE DEPLOYMENT OF AIRBAGS, (II) CONTROL OF A VEHICLE, UNLESS THERE IS A FAIL-SAFE OR REDUNDANCY FEATURE (WHICH DOES NOT INCLUDE USE OF SOFTWARE IN THE XILINX DEVICE TO IMPLEMENT THE REDUNDANCY) AND A WARNING SIGNAL UPON FAILURE TO THE OPERATOR, OR (III) USES THAT COULD LEAD TO DEATH OR PERSONAL INJURY. CUSTOMER ASSUMES THE SOLE RISK AND LIABILITY OF ANY USE OF XILINX PRODUCTS IN SUCH APPLICATIONS. © Copyright 2017 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Vivado, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.