Universidade Federal da Bahia
Instituto de Computação

Programa de Pós-Graduação em Ciência da Computação

IMPROVING THE SECURITY OF THE BOOT PROCESS FOR IOT DEVICES ON THE RISC-V ARCHITECTURE

Everton Roberto Zanotelli

DISSERTAÇÃO DE MESTRADO

Salvador
16 de dezembro de 2023

EVERTON ROBERTO ZANOTELLI

IMPROVING THE SECURITY OF THE BOOT PROCESS FOR IOT DEVICES ON THE RISC-V ARCHITECTURE

Esta Dissertação de Mestrado foi apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Leobino Sampaio

Salvador
16 de dezembro de 2023

SUMÁRIO

LISTA DE FIGURAS

Capítulo 1

INTRODUÇÃO

The continued expansion of IoT devices is promoting a rise in the attempts of security attacks focused on the boot sequence(CREMER, Frank et al, 2022), aiming to maintain control of this kind of device for a long period. How a computer starts today is reasonable different and more complex than it was a two decades ago, as a procedure that is present in all devices, the boot sequence is a field with plenty of topics to research.

(Arbaugh et al,1997) proposed the first boot protection mechanism, which involves creating a chain of integrity checks for every stage of the boot process. This mechanism ensures that each stage verifies the integrity of the subsequent stage. This mechanism is detailed in the Unified Extensible Firmware Interface (UEFI).

One recent case that calls for attention in this topic is the release of the Black Lotus BootKit(Figure 1.1) where even a fully updated Windows 11 system with UEFI Secure boot active had it's security boot process compromised granting control over the hardware and it's resources(WELIVESECURITY, 2023).

In the IoT domain this kind of security breach gains a massive surface due to the enormous quantity of devices and keeping they all running in a trusted and healthy state is very challenging without establishing a chain of trust from the moment the energy reaches the device to when the operating system and it's applications are loaded. One promising architecture for IoT machines that is open-source is the RISC-V architecture where it's easier to make improvements in the boot process.

RISC-V is a license-free, royalty-free technology that is based on a reduced instruction set computing (RISC) principle where it's purpose is to be simple and extensible enabling customization and innovation of specific necessities on a range from micro controllers to supercomputers(RISCV, 2023).

Also the programming language that is going to be used in this research is Rust for the reason that it address a surface of attacks that older languages are struggling with like buffer overflow or memory corruption, rust deals with it by introducing the concept of ownership(KLABNIK, Steve; NICHOLS, Carol, 2023) where a variable value cannot be altered if not stated otherwise also the Rust compiler is very adaptable to different hardware and architectures and Rust memory management is very efficient to fit in the resource restricted scenario of IoT.

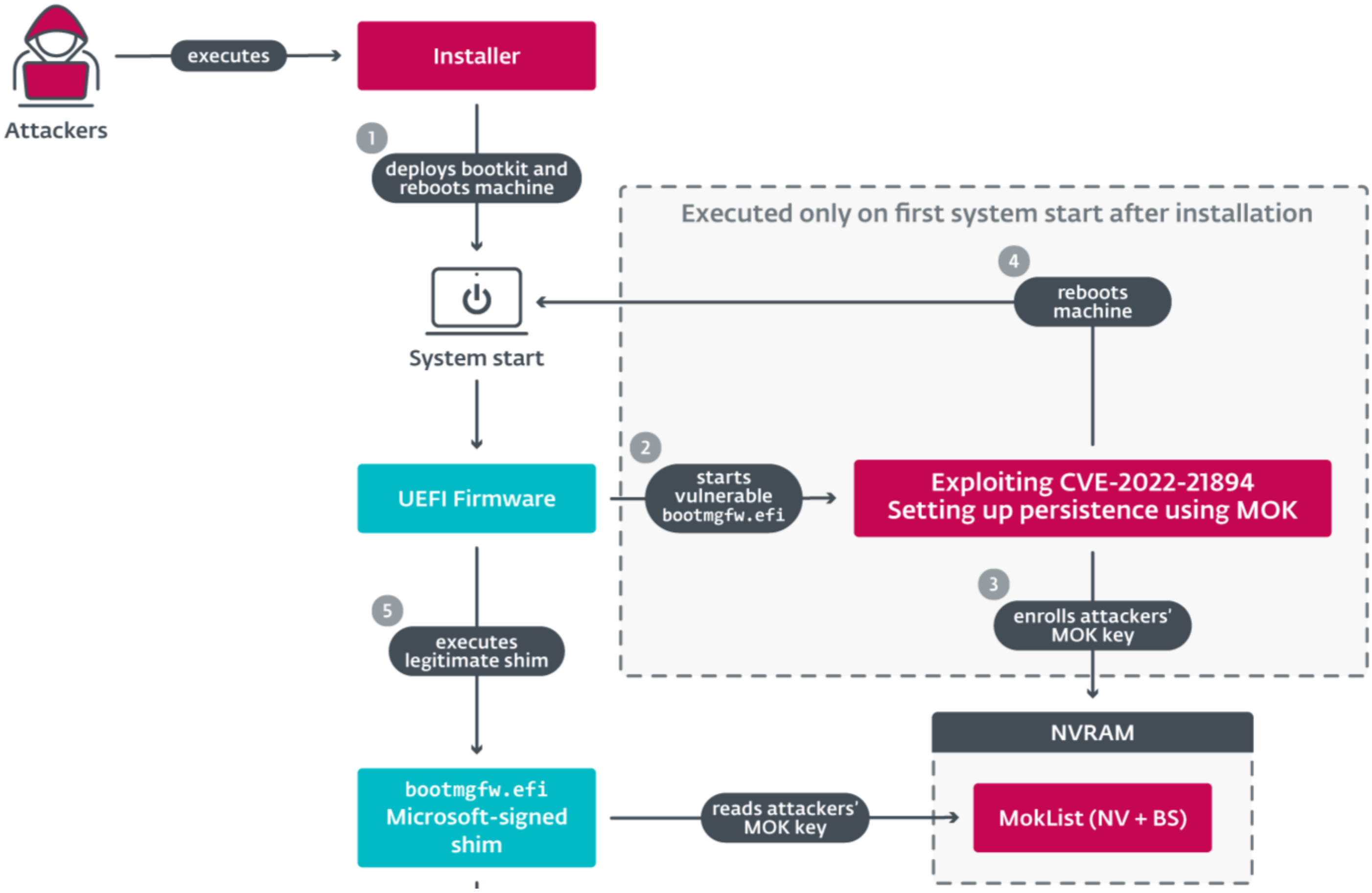
Attackers
executes
Installer
1
deploys bootkit and reboots machine
Executed only on first system start after installation
4
reboots machine
System start
2
starts vulnerable bootmgfw.efi
UEFI Firmware
Exploiting CVE-2022-21894
Setting up persistence using MOK
3
enrolls attackers' MOK key
5
executes legitimate shim
NVRAM
bootmgfw.efi
Microsoft-signed shim
reads attackers' MOK key
MokList (NV + BS)

Figura 1.1 Black Lotus flow attack(Red: Malicious Code, Blue: Legit code)

1.1 PROBLEMS AND MOTIVATIONS

The boot sequence is a crucial procedure of a system security. It ensures that the system only starts from reliable sources and that the boot process itself is secure. This can include methods such as secure boot, where the system validates the integrity of the bootloader and the operating system before they are allowed to run, and anti-rollback protections, which prevent the system from starting an older version of the operating system that could be vulnerable to attacks. One of the principles utilized in secure systems is the creation of a chain of trust for all software components that run from the initial boot loader up to trusted applications(Y. Chao and Y. Meng-ting, 2010). This chain of trust is established from a root of trust that is difficult to tamper with. This process is known as a secure boot sequence(Y. Chao and Y. Meng-ting, 2010).

The Basic Input/Output System (BIOS) is a firmware element that resides in nonvolatile memory, typically a flash chip. The BIOS is responsible for initiating the boot loader, which is the initial software component that gets loaded during the boot process. The boot loader is stored on the hard drive, alongside the operating system and applications.

For cyber attackers, it is advantageous to compromise a component that is loaded earlier than one that is loaded later. This is because gaining control at an early stage allows them to control all subsequent components. While successful attacks against user-mode software programs may not be considered significant achievements in the security community at present, the BIOS and boot loader are becoming more attractive targets. A number of such attacks have been reported in recent years(CREMER, Frank et al, 2022).

The security gap that exists between the firmware and the operational system phase is a significant area of concern in the boot process. However, mechanisms such as the Root of Trust for Measurement and the Firmware Attack Surface Reduction approach are being used to mitigate this gap and ensure the security of the system during the boot process(Banik, S., Zimmer, V., 2022).

The motivation behind this research is to address this security gap. The RISC-V community is actively exploring security solutions aimed at achieving a root of trust (RoT) and ensuring that sensitive information on RISC-V devices is not tampered with or leaked. However, many RISC-V security research projects are underway, and a comprehensive survey of RISC-V security solutions has not yet been conducted.

The research will focus on suggesting some improvements around the RISC-V boot sequence. This will involve a set of evaluations in the aspects of secure boot, memory protection, PUF-based key management. The boot flow on the RISC-V architecture has five stages in which the main focus of attackers in this phase is the gap that exists between the handover of control from the bootloader to the operating system where the majority of vulnerabilities are located(Banik, S., Zimmer, V., 2022).

The tests are going to be made first on a QEMU virtualization of a RISC-V machine with the virt board where we can test different tweaks to the boot process. Due to the diverse nature of RISC-V hardware, QEMU offers extensive support for a wide range of RISC-V guests. This is in contrast to x86 hardware, which is more consistent. RISC-V CPUs are often integrated into "system-on-chip"(SoC) designs from various companies,

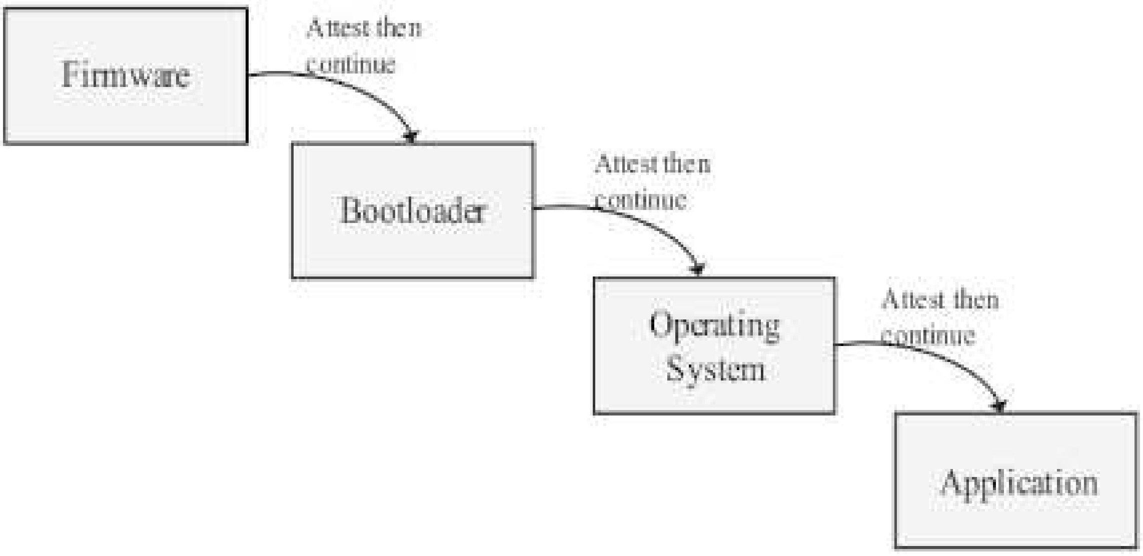

Figura 1.2 The boot flow

each with different devices. and these SoCs are then used in machines that can also vary significantly even if they use the same SoC for that reason our virtualization tests will be executed on the virt board, a platform which doesn't correspond to any real hardware and is designed for use in virtual machines(QEMU, 2023).

The second test will be run on bare-metal to get as close as possible of a real case scenario of a IoT device with the boot process tampered using the MangoPi MQ-Pro board with a XuanTie C906 RISC-V CPU, Integrated 2 Kbits OTP storage space, can boot from SD card, eMMC, SPI NOR Flash or SPI NAND Flash.

1.2 RESEARCH HYPOTHESIS

Considering the security scenario described in section 1.1 involving the improvement of the boot flow the following research hypotheses were defined:

- The open source nature of the RISC-V architecture can enable specific optimization and strategies in terms of security in different stages of RISC-V boot sequence applied to the IoT scenario allowing the build of trust in one of the critical points during the lifetime of a system.

- The strategy of enforcing dominance over the boot stage of IoT devices can improve control, ownership and trust over all the devices therefor reducing management and operation spending's due to the capability of recovering the device to a trusted state wherever its behaviour becomes suspect.

1.3 OBJECTIVES

The present work has the general objective to investigate, develop and propose a suggestion based on the programmable capability of the RISC-V architecture to reduce the

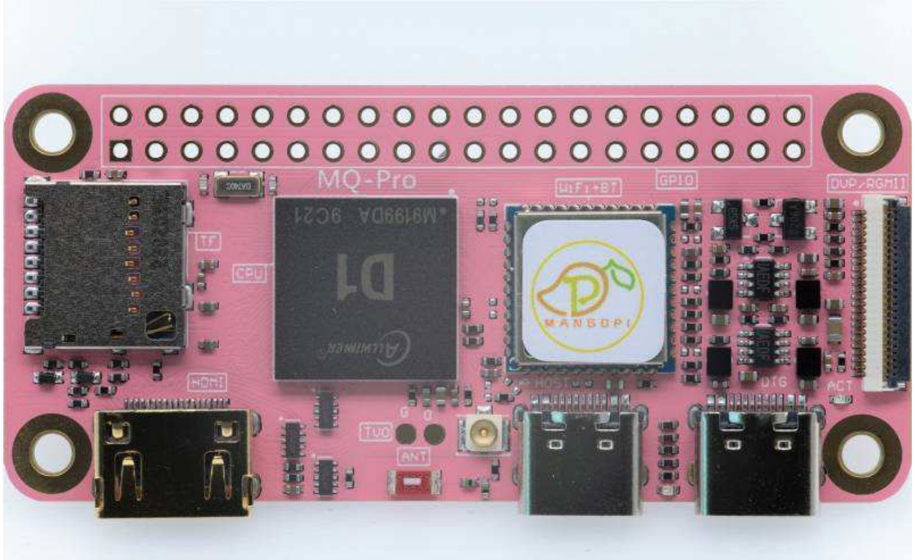

Figura 1.3 MangoPi MQ-Pro

surface of buffer overflow and memory corruption attacks. Therefor the object of this research can be in tree specific objects:

- Develop and implementation of simulation scenarios with the main components of a RISC-V CPU in a IoT paradigm(RISC-V boot sequence for constrained devices)
- Plan and execute buffer overflow and memory corruption attacks on the boot flow of a RISC-V architecture by QEMU and in a bare-metal MangoPi MQ-Pro board device to analyse the result of the attack against our improvement suggestion
- Evaluate the data generated by the tests and organize it for the conclusion of this research and future improvements

1.4 CONTRIBUTIONS

The resulting contributions of this research are:

- Development of a virtual environment for RISC-V boot tests that can be extended for future research.
- Improvements suggestions that the RISC-V community can evaluate and adopt.
- Hardening of the RISC-V boot flow.

1.5 METODOLOGY

The planning of research activities carried out during the project was conducted according to the following logical sequence: (i) conducting a literature review on studies addressing the topics of trusted computing and security in the boot process; (ii) based on the literature review and the secure boot paradigm, defining, modeling, and implementing the proposed solution; (iii) planning and executing initial experiments; (iv) evaluating the partial results obtained in stage iii; (v) based on the results obtained in the previous stages, carrying out the modeling and final implementation; (vi) evaluating the final results obtained in stage (v).

In addition to the research activities, micro activities were carried out at the end of some larger activities. These include: (i) writing and submitting a scientific article after the completion of the evaluation of partial results; (ii) writing a scientific article after the completion of the evaluation of final results; (iii) writing the master's thesis.

1.6 BIBLIOGRAPHY

CREMER, Frank et al. Cyber risk and cybersecurity: a systematic review of data availability. The Geneva Papers on risk and insurance-Issues and practice, v. 47, n. 3, p. 698-736, 2022.

ARBAUGH, William et al. A secure and reliable bootstrap architecture. In: Security and Privacy, 1997. Proceedings., 1997 IEEE Symposium on. IEEE. p. 65-71, 1997.

WELIVESECURITY. BlackLotus UEFI bootkit: Myth confirmed. <https://www.welivesecurity.com/
uefi-bootkit-myth-confirmed/>. Accessed on: November 23, 2023.

RISCV. About. <https://riscv.org/about/>. Accessed on: November 23, 2023.

KLABNIK, Steve; NICHOLS, Carol. The Rust programming language. No Starch Press, 2023.

Y. Chao and Y. Meng-ting, "Security Bootstrap Based on Trusted Computing,"2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, China, 2010, pp. 486-489, doi: 10.1109/NSWCTC.2010.121.

Banik, S., Zimmer, V. (2022). System Firmware Architecture. In: System Firmware. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-7939-74

QEMU. RISC-V System emulator. Choosing a board model. <https://www.qemu.org/docs/master/sys
riscv.html>. Accessed on: November 23, 2023.