A secure execution framework for the IoT

Dr. Ing. Alexandru Radovici

Abstract

In this project, we propose an isolated, secure WebAssembly bytecode execution framework for the Internet of Things (IoT).

For increased security, this solution mimics all operating system functionalities so that IoT application software is run in separate processes/threads as if a standard operating system directly executed the instructions.

Running verified bytecode in an isolated framework will assure that all the security enforcement is done by software and does not rely on the security features provided by hardware components.

Security features are essential in IoT applications where replacing or updating hardware components due to security concerns is expensive (or even impossible).

With the proposed framework security flaws are corrected with a straightforward software update. We will provide a toolchain to build and package applications for this framework and, finally we validate the approach by evaluating it (regarding both security and performance) on a few real-world IoT case studies.

Background

Computer, both software and hardware, security is an ever-changing area of active research. The continuous introduction of new software and hardware technologies, coupled with their increased complexity, leads to new threat models and security risks. One of the technologies that has a rapid growth rate and is one of the main drivers of innovation is the Internet of Things (IoT).

In the context of the Internet of Things, the fundamental specific issues that we identify regarding secure IoT software execution are:

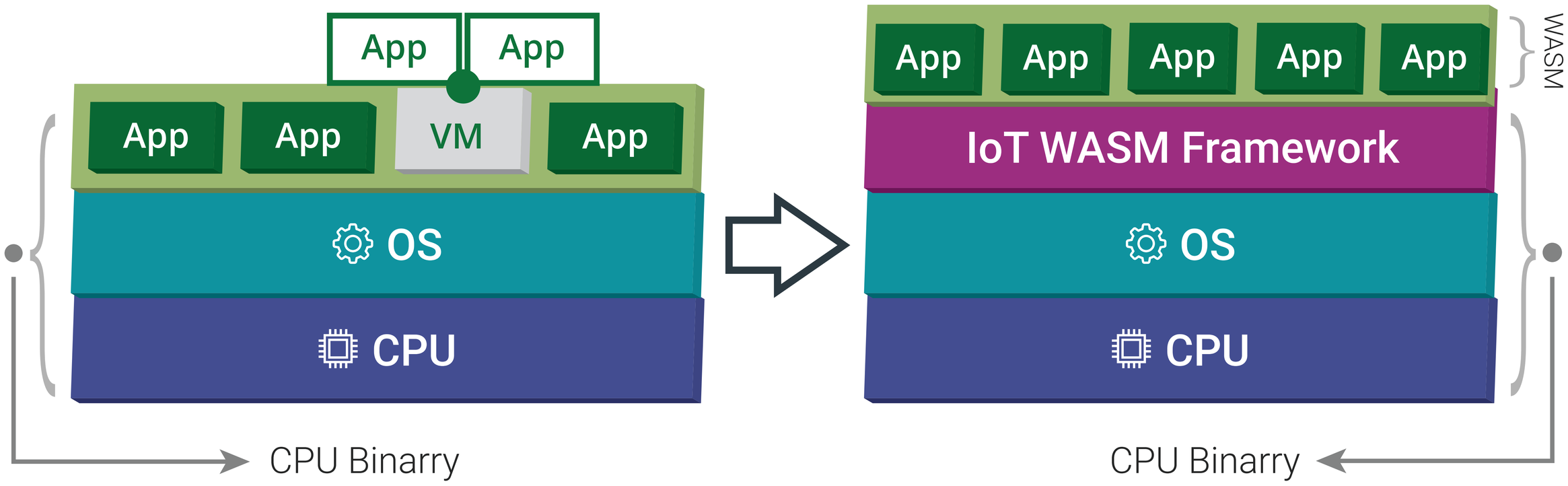
1. The need to protect software running on IoT devices against classic arbitrary code execution vulnerabilities;

2. The need to allow for protection against (or to offer a software-based solution to) hardware design vulnerabilities since, in general, we are unable (or it is unpractical) to update all IoT hardware when a new security issue is found at that level;

3. Provide software developers with a transparent code execution platform for the IoT where they can easily control the program flow.

4. Undertake a performance analysis of the proposed framework, since solutions to some of the problems enumerated above may significantly slow down the performance of IoT systems;

5. Test the proposed platform on several real-world IoT systems where security features are a design priority.



The solution

The aim is to provide a framework for running only WebAssembly (wasm) bytecode. WebAssembly bytecode has been designed to be able to run existing software, written in any language and compiled into WebAssembly, in a browser environment, for the Web. As the browser is a sandboxed environment, several security checks are necessary to implement WebAssemebly (wasm). We believe we can benefit from its security by design and turn it into a general purpose executable, outside the browser.

Our research will focus on building a whole coherent set of security features around the WebAssembly framework. Then, with the framework in place, we focus on getting applications into the proposed wasm execution format. Finally, after the framework and the toolchain are in place, the next step is to build several well-known libraries into the new proposed format.

Objectives

RT 1. A user space framework which runs isolated WebAssembly bytecode outside a browser and provides means to load binaries and libraries.

RT 2. A toolchain for building WebAssembly bytecode applications outside of the browser and any Javascript machine.

RT 3. Port several libraries from native (binary) code to WebAssembly bytecode.

RT 4. Package and securely distribute applications on IoT platforms.

RT 5. Analyze the performance impact of the proposed framework in real-world IoT applications and scenarios.

References

1. A. Haas, A. Rossberg, D. L. Schuff, B. L. Titzer, M. Holman, D. Gohman, L. Wagner, A. Zakai and J. F. Bastien, Bringing the web up to speed with WebAssembly, Proceedings of the 38th ACM SIGPLAN Conference on Programming Language Design and Implementation, Barcelona, Spain, June 18 - 23, 2017

2. E. Reshetova, J. Karhunen, T. Nyman and N. Asokan, Security of OS-Level Virtualization Technologies, Nordic Conference on Secure IT Systems, 2014

3. D. Sin and D. Shin, Performance and Resource Analysis on the JavaScript Runtime for IoT Devices, International Conference on Computational Science and Its Applications, 2016

4. D. Tiwari and Y. Solihin, Architectural characterization and similarity analysis of sunspider and Google's V8 Javascript benchmarks, IEEE International Symposium on Performance Analysis of Systems and Software, New Brunswick, NJ, USA, April 1 - 3, 2012

5. D. Gruss, C. Maurice, K. Wagner and S. Mangard, Flush+ Flush: a fast and stealthy cache attack, International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, June 12, 2016

6. C. Watt, Mechanising and Verifying the WebAssembly Specification, Procee dings of 7th ACM SIGPLAN International

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

UNIVERSITATEA POLITEHNICA DIN BUCUREȘTI UPB 1818