Synthesis of Hardware Sandboxes for Trojan Mitigation in Systems on Chip - Hardware Demo Proposal

Christophe Bobda*, Taylor JL Whitaker*, Charles Kamhoua**, Kevin Kwiat**, and Laurent Njilla**
* Computer Science and Computer Engineering Department, University of Arkansas, Fayetteville, AR
** Air Force Research Laboratory, Cyber Assurance Branch, Rome, NY

Hardware Sandboxing was proposed in [1] as a means to efficiently mitigate some classes of hardware trojans from causing potential damage to trusted system resources. Partitioning the system to trusted and untrusted sectors, we can control the interactions of untrusted modules placed within the physically separated sandbox with the trusted system sector. The use of checkers allow us to monitor interactions in real-time and detect deviations from the IP interface behavioral specification. To further separation from critical resources, hardware sandboxes provide virtualized resources to sandboxed IP to satisfy any resource requirements and nullify any attempts to modify physical resources. Implementing hardware sandboxes was originally difficult, for each IP has a unique interface and resource requirements. Thus, we proposed a design flow for automatically generating these hardware sandboxes for easy integration of our secure isolation method to the system design and integration process. Our current design flow caters to the man design methodologies, however we focus on VHDL and Xilinx Vivado to demonstrate the ability of our tool to alleviate many tedious tasks of hardware sandbox implementations.

Hardware Demo

We intend to demonstrate our design flow for automatically wrapping untrusted IP with a hardware sandbox to allow system integrators to include questionable IP in critical systems. There are three phases of the generation process we wish to describe with display: 1) Specification of IPs, 2) IP Model Optimization, and 3) Sandbox IP Output and Integration.

1) Specification of IP - The first step in preparing a sandbox for a set of IP is to provide the specification of their interfaces, interface behaviors, and unauthorized actions. We capture these with two configuration files. We intend to discuss the sections of each configuration file for the case of Trusthub trojan benchmarks for the RS232 UART protocols. The variations sections of the configuration files will be outlined in a simple slide show that with references to the complementing poster. Considering our work was accepted to the HOST as a poster, we intend to leverage the poster for referencing the formalism that underlies the configuration files.

2) IP Model Optimization - The sandbox generation process is carried out by a simple command line tool. The interface of the tool will be briefly outlined before explaining the underlying computations being performed. This again will reference the accompanying poster for visual aids for the optimizations performed.

3) Sandbox IP Output and Integration - After running our tool, we will have generated a sandbox that can be immediately used in Xilinx Vivado Software and integrated into an existing system design. We intend to briefly cover the contents of the produced VHDL components in a slideshow to ensure clear visibility of important code segments. We then intend to show the process of integration of the sandbox and untrusted IP to an existing design in Xilinx Vivado. A pre-synthesized design of the system will be running on a Zybo FPGA to demonstrate the ability of our hardware sandboxes to detect the Trusthub benchmarks.

Conclusion

To conclude, the hardware demo is expected to last approximately five minutes to cover each of the three phases. The audience will be introduced to concepts used in our tool's generation process and will be able to see the ease of integration of the produced sandboxes. With the use of a single FPGA, a laptop for software/slideshow explanations, and an accompanying poster, the physical setup of our demo would require only an outlet for power, an easel for poster display, and a table large enough for a single laptop computer.

[1] J. Mead, C. Bobda and T. J. Whitaker, "Defeating drone jamming with hardware sandboxing," *2016 IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*, Yilan, Taiwan, 2016, pp. 1-6. doi: 10.1109/AsianHOST.2016.7835557

Approved for Public Release; Distribution Unlimited: 88ABW-2016-4561, dated 17 Sep 2016.