JackHammer: Rowhammer and Cache Attacks on Heterogeneous FPGA-CPU Platforms

Zane Weissman¹  Thore Tiemann²  Daniel Moghimi³  Evan Custodio⁴  Thomas Eisenbarth²  Berk Sunar¹
¹Worcester Polytechnic Institute  ²University of Lübeck  ³UC San Diego  ⁴Amazon Web Services

Abstract
We studied two new heterogeneous FPGA-CPU platforms from Intel: the integrated Arria 10 GX which shares a chip with its host CPU, and the Arria 10 GX PAC expansion card which connects the FPGA to the CPU via a PCIe interface.

We show a cache covert channel between FPGA and CPU and JackHammer, which is a Rowhammer attack from FPGA against a host's main memory. It performs **twice as fast** as conventional CPU Rowhammer and causes **four times as many faults**.

Caching Behavior
The memory access latency of the FPGA depends on the location answering a memory request. FPGA memory reads do **not alter the caching state** or location. FPGA memory writes update the CPU's last level cache state and data.

Covert Channel
We constructed a covert channel with the FPGA as the sender and a cooperative CPU program as receiver. The FPGA sends binary messages by writing to a cache line when transmitting a One and staying quiet otherwise. The receiver **continuously probes the cache set** to detect access latency fluctuations to receive the messages. While using heavily redundant encoding, we still achieve a **throughput of 94.98 kBit/s**.

References

Special thanks to Intel’s Alpa Thrived and Sayak Ray and former Intel’s Evan Custodio for their guidance and support.

Contact
Zane Weissman
zweissman@wpi.edu

Thore Tiemann
ttiemann@uni-luebeck.de