Prototyping a Secure Controller for Trusted Heterogeneous Disaggregated Architectures

Felix Gust
Advisor: Dr. Atsushi Koshiba
Chair of Distributed Systems and Operating Systems
https://dse.in.tum.de/

15.02.23 – 15.08.23

Motivation

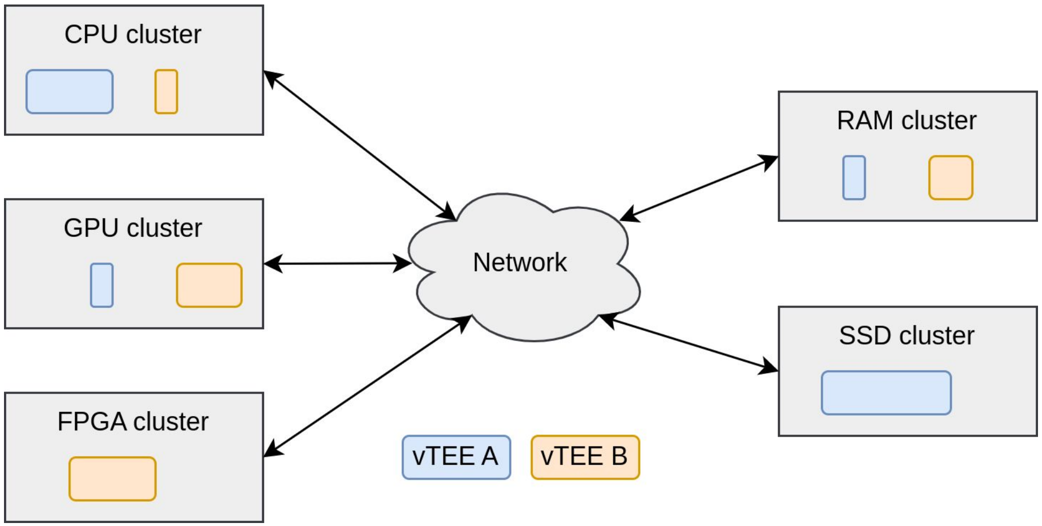
- Data center architectures are becoming more
  - Heterogeneous: CPUs, GPUs, FPGAs, ASICs, …
  - Disaggregated: Devices in racks connected to the network
- Workloads involve sensitive data
- New security challenges
- Trusted isolated environment?

State-of-the-art

- Trusted Execution Environments
  - CPU-centric and vendor-specific (Intel SGX/TDX, AMD SEV, ARM TrustZone)
  - Device-specific (Graviton [1], ShEF [2])
- Distributed operating systems
  - LegoOS [3]: Supports Linux applications, CPU-centric
  - FractOS [4]: Own programming model, execution graph
- Distributed TEEs
  - HETEE [5]: Centralized security controller, limited to one rack
  - [6]: Similar to HETEE with multiple security controllers

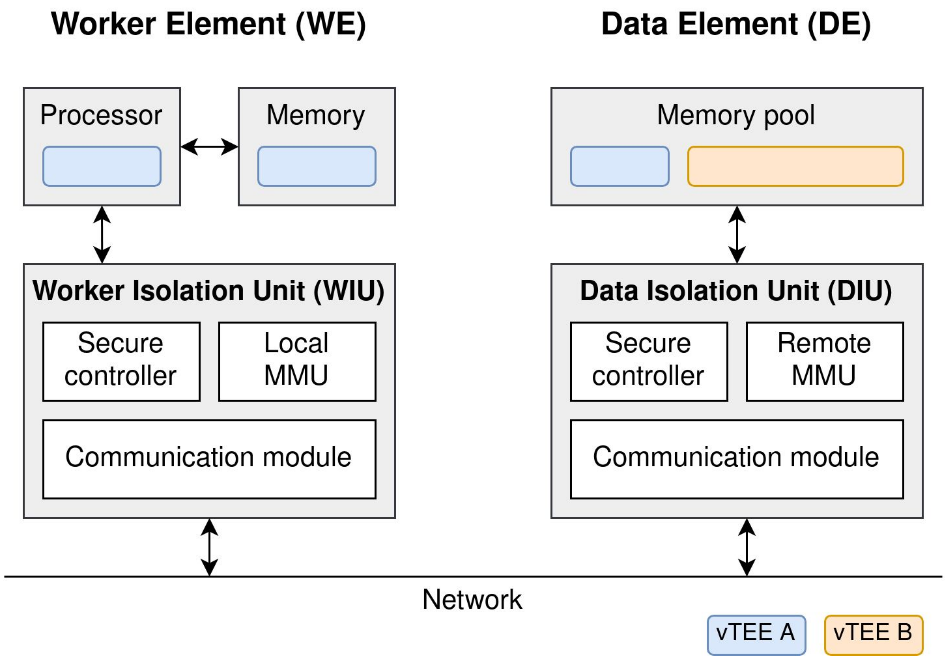
Research gap

How to establish virtual Trusted Execution Environments spanning multiple heterogeneous disaggregated resources?



Problem statement

Develop a prototype of a secure controller for trusted heterogeneous disaggregated architectures



Background

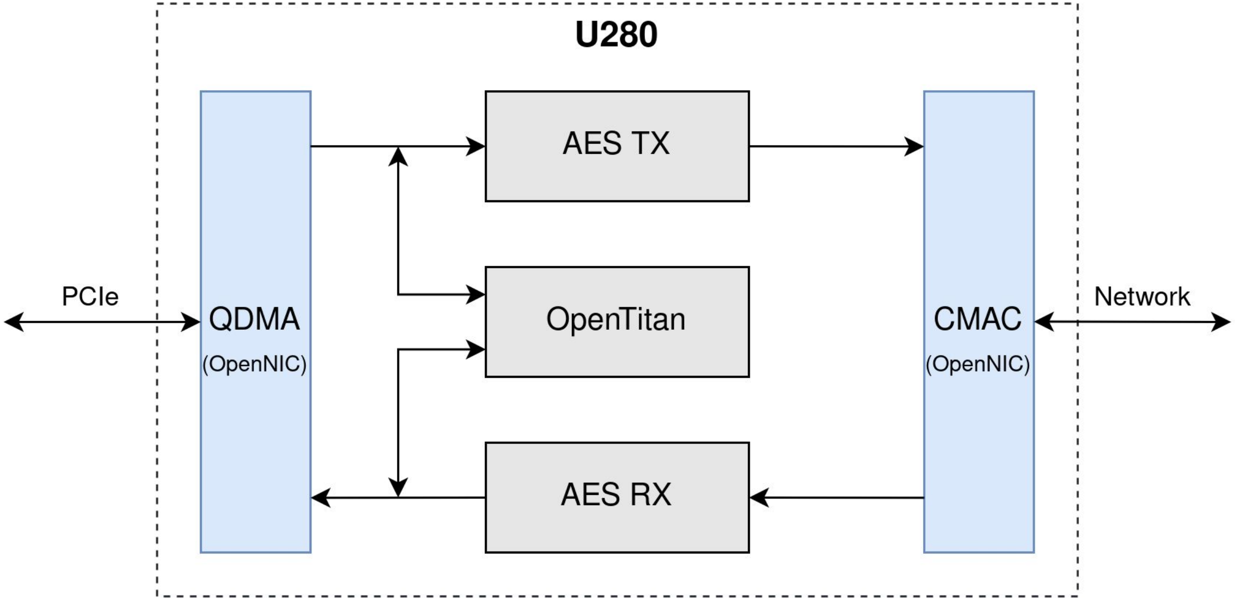
- OpenTitan
  - Open-source silicon Root of Trust
  - Officially supports one FPGA development board
- AES accelerator
  - Performance of the OpenTitan AES module is insufficient
  - Two modes: communication with OpenTitan and passthrough
- Xilinx Alveo U280
  - PCIe FPGA card
  - 100 Gbit/s network interface

Design Goals

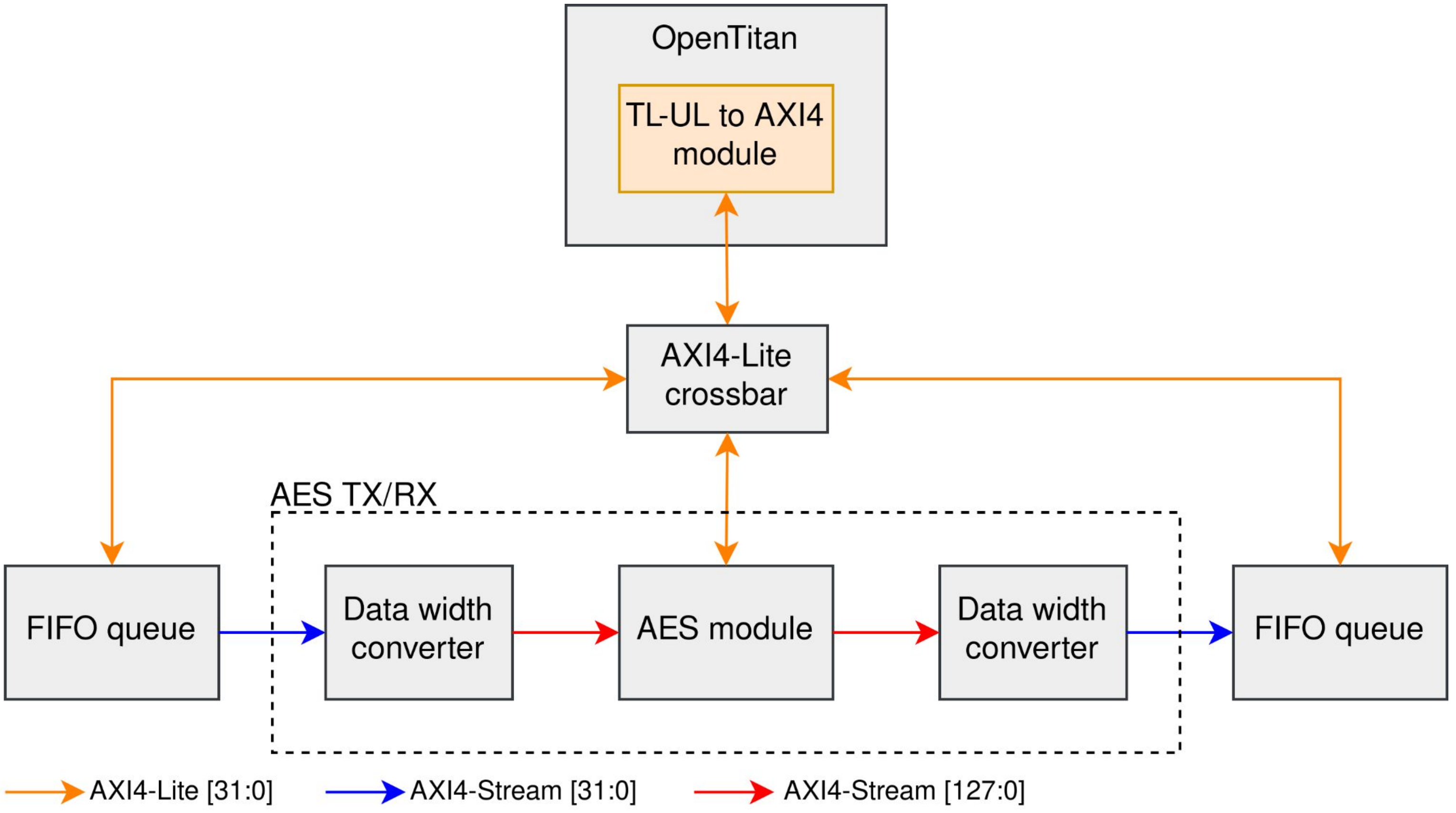
- Implementation on the U280
- OpenTitan as Root of Trust and main CPU
- High-speed symmetric encryption
  - vTEE management (OpenTitan ⇔ OpenTitan)
  - vTEE execution (passthrough, e.g. CPU ⇔ GPU)

Design

- OpenTitan as Root of Trust
- OpenNIC for communication via PCIe and network
- AES modules controlled by OpenTitan



Implementation



Implementation

- Porting the OpenTitan to U280
  - Change config files and constraints
  - Package as Vivado IP ⇒ easy integration into a larger project
- OpenTitan AXI4 module
  - Based on ToAXI4 module from Rocket Chip project[1]
  - Converts internal TL-UL bus to external AXI4 bus
- AXI4-Lite crossbar to connect multiple modules to OpenTitan

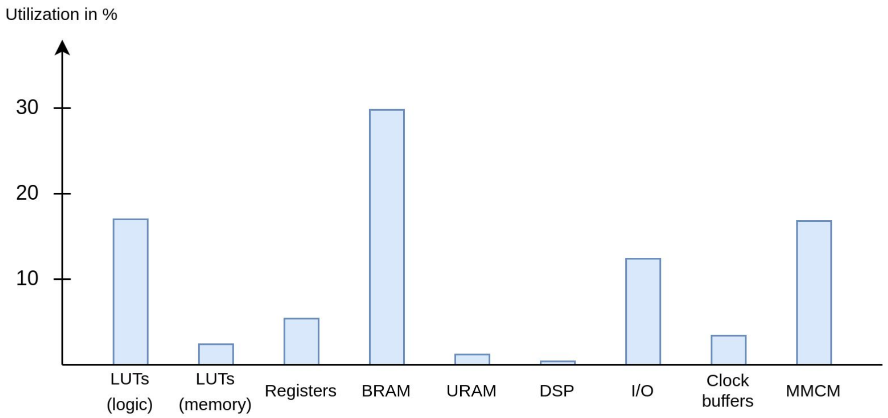
[1] https://github.com/chipsalliance/rocket-chip/blob/master/src/main/scala/tilelink/ToAXI4.scala

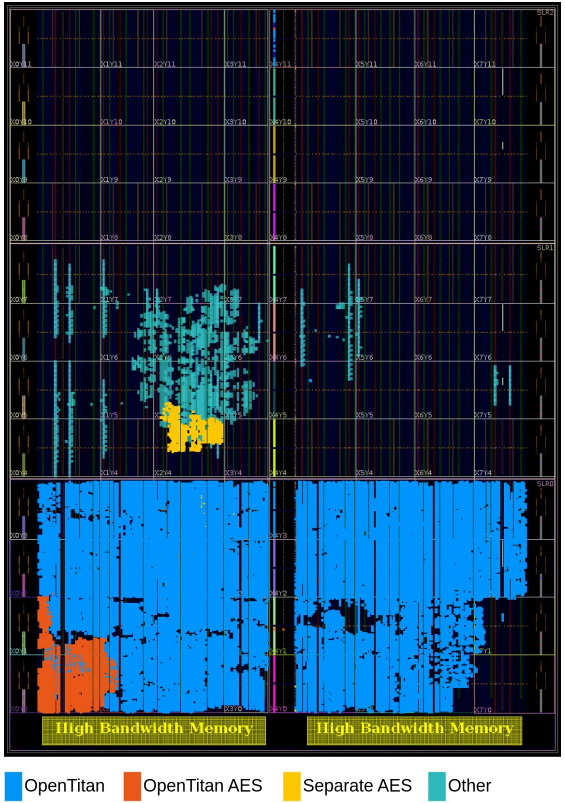
Implementation

- AES module
  - Based on AES module form Xilinx Vitis RTL kernel tutorial[1]
  - Operates on AXI4-Stream traffic
  - AES function only ⇒ ECB mode!
- Data width converters
  - AXI4-Stream data widths: FIFOs 32 bit, AES 128 bit

[1] https://github.com/Xilinx/Vitis-Tutorials/blob/2021.2/Hardware_Acceleration/Design_Tutorials/05-bottom_up_rtl_kernel/doc/krnl_aes.md

Evaluation: FPGA Utilization

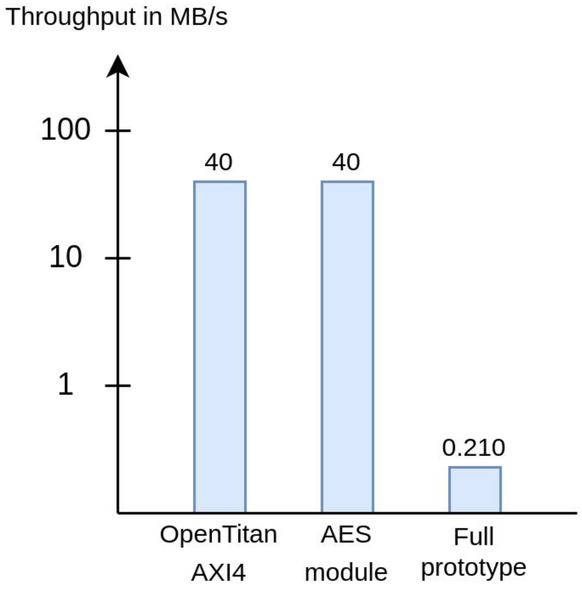




Utilization is low enough to add additional modules (e.g. OpenNIC)

Evaluation: Performance

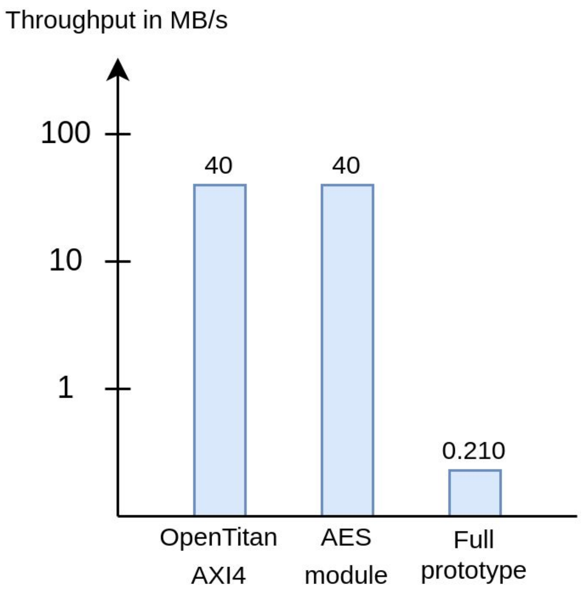
- OpenTitan AXI4 module: 40 MB/s
  - Limited by 10 MHz TL-UL bus
  - Sufficient for intended use case
- AES module: 40 MB/s
  - Limited by 10 MHz AXI4 clock
  - Too slow for high-speed network traffic
  - Xilinx benchmark with higher clock: 390 MB/s



AES performance is sufficient for OpenTitan but insufficient for communication between worker elements

Evaluation: Performance

- Full prototype: 210 KB/s
  - OpenTitan ⇒ FIFO ⇒ AES ⇒ FIFO ⇒ OpenTitan
  - Extrapolated from 4 KB per run
  - Limited by
    - AXI4 clock
    - Data width converters
    - FIFO copying



Full prototype performance limited by multiple factors, greatly lags behind raw AES module performance

Summary

Data center architectures are becoming more heterogeneous and disaggregated

- Goal: distributed virtual Trusted Execution Environments (vTEEs)

Secure controller prototype

- Root of Trust OpenTitan ported to U280 & extended with AXI4 module
- AES module for encrypting network traffic controlled by OpenTitan
- Evaluation: resource utilization low, AES performance lacking

Code

- OpenTitan: https://github.com/TUM-DSE/TDA-opentitan
- Full prototype: https://github.com/TUM-DSE/TDA-testbed

References

[1] S. Volos, K. Vaswani, and R. Bruno. “Graviton: Trusted Execution Environments on GPUs.” In: 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18). 2018, pp. 681–696. isbn: 978-1-939133-08-3.

[2] M. Zhao, M. Gao, and C. Kozyrakis. “ShEF: Shielded Enclaves for Cloud FPGAs.” In: Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems. Feb. 28, 2022, pp. 1070–1085. doi:10.1145/3503222.3507733. arXiv: 2103.03500 [cs].

[3] Y. Shan, Y. Huang, Y. Chen, and Y. Zhang. “LegoOS: A Disseminated, Distributed OS for Hardware Resource Disaggregation.” In: 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18). 2018, pp. 69–87. isbn: 978-1-939133-08-3.

[4] L. Vilanova, L. Maudlej, S. Bergman, T. Miemietz, M. Hille, N. Asmussen, M. Roitzsch, H. Härtig, and M. Silberstein. “Slashing the Disaggregation Tax in Heterogeneous Data Centers with FractOS.” In: Proceedings of the Seventeenth European Conference on Computer Systems. EuroSys ’22. New York, NY, USA: Association for Computing Machinery, Mar. 28, 2022, pp. 352–367. isbn: 978-1-4503-9162-7. doi: 10.1145/3492321.3519569.

[5] J. Zhu, R. Hou, X. Wang, W. Wang, J. Cao, B. Zhao, Z. Wang, Y. Zhang, J. Ying, L. Zhang, and D. Meng. “Enabling Rack-scale Confidential Computing Using Heterogeneous Trusted Execution Environment.” In: 2020 IEEE Symposium on Security and Privacy (SP). 2020 IEEE Symposium on Security and Privacy (SP). San Francisco, CA, USA: IEEE, May 2020, pp. 1450–1465. isbn: 978-1-72813-497-0. doi: 10.1109/SP40000.2020.00054.

[6] A. Dhar, S. Sridhara, S. Shinde, S. Capkun, and R. Andri. Empowering Data Centers for Next Generation Trusted Computing. Nov. 1, 2022. doi: 10.48550/arXiv.2211.00306. arXiv: 2211.00306 [cs]. url: http://arxiv.org/abs/2211.00306. preprint.