Preliminary Study of Trusted Execution Environments on Heterogeneous Edge Platforms

Zhenyu Ning, Jinghui Liao, Fengwei Zhang, **Weisong Shi**

COMPASS Lab
Wayne State University

October 27, 2018

Outline

WAYNE STATE UNIVERSITY

- Introduction
- Trusted Execution Environment (TEE)
  - Intel Software Guard eXtension (SGX)
  - ARM TrustZone Technology
  - AMD Secure Encrypted Virtualization Technology
- Edge Computing with TEE
- Conclusion and Future Work

Outline

WAYNE STATE UNIVERSITY

Why moving to Edge from Cloud?

Edge Computing

WAYNE STATE UNIVERSITY

Why moving to Edge from Cloud?

- Reduced network latency for time-sensitive tasks.
  E.g. *Real-time monitoring for transportation [1]*.

Edge Computing

WAYNE STATE UNIVERSITY

Why moving to Edge from Cloud?

- Reduced network latency for time-sensitive tasks.
  E.g. *Real-time monitoring for transportation [1]*.

- Increased efficiency for performance-sensitive tasks.
  E.g. *Video analytics for public safety [2]*.

Edge Computing

WAYNE STATE UNIVERSITY

Why moving to Edge from Cloud?

- Reduced network latency for time-sensitive tasks.
  E.g. *Real-time monitoring for transportation [1]*.
- Increased efficiency for performance-sensitive tasks.
  E.g. *Video analytics for public safety [2]*.
- Increased privacy for sensitive data.
  E.g. *Data of home security cameras [3]*.

What about the security?

Edge Computing

WAYNE STATE UNIVERSITY

What about the security?

- Close to end-user
  $\Rightarrow$ Close to manipulation
- Distributed deployment
  $\Rightarrow$ Lacking centralized protection

Solution?

Outline

WAYNE STATE UNIVERSITY

- Introduction
- Trusted Execution Environment (TEE)
  - Intel Software Guard eXtension (SGX)
  - ARM TrustZone Technology
  - AMD Secure Encrypted Virtualization Technology
- Edge Computing with TEE
- Conclusion and Future Work

Trusted Execution Environment (TEE)

- An isolated execution environment that remains secure even when the system software is compromised.
- Using hardware-assisted protection to guarantee the security.
- Different hardware vendors use different protection mechanisms.

Intel Software Guard eXtension (SGX)

WAYNE STATE UNIVERSITY

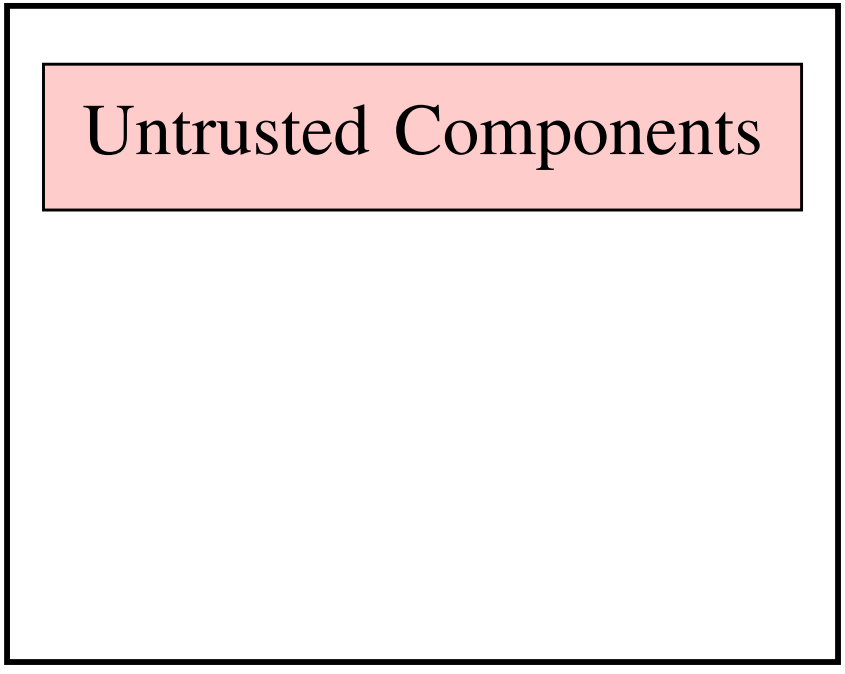
Intel Software Guard eXtension (SGX) is proposed via three research papers in 2013 [4, 5, 6].

- The user-level application creates an *enclave* to act as a TEE.
- The memory inside an *enclave* is encrypted by a hardware memory encryption engine.
- Memory access from the outside to the *enclave* is prohibited.

Intel Software Guard eXtension (SGX)

WAYNE STATE UNIVERSITY

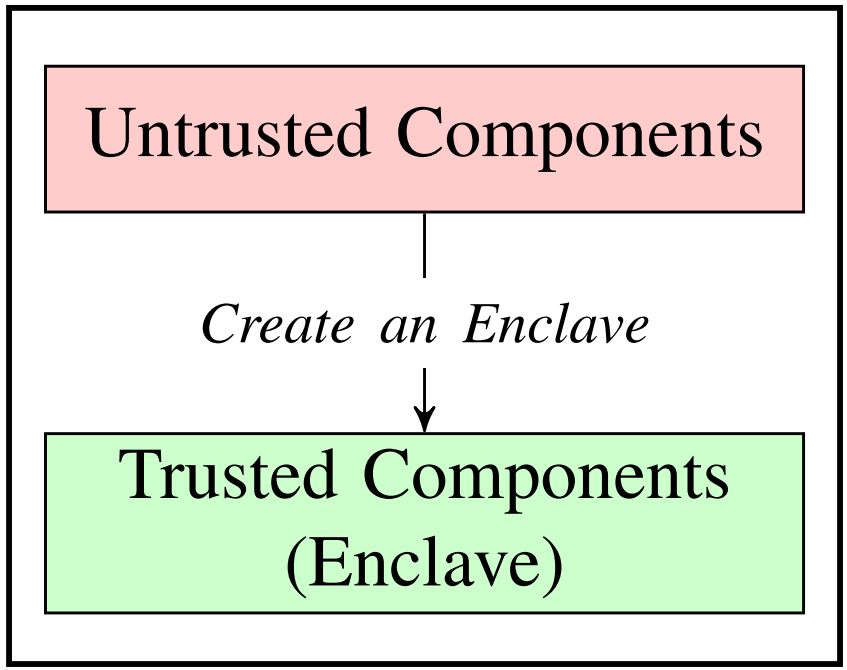
Securing Application in Untrusted OS



Intel Software Guard eXtension (SGX)

WAYNE STATE UNIVERSITY

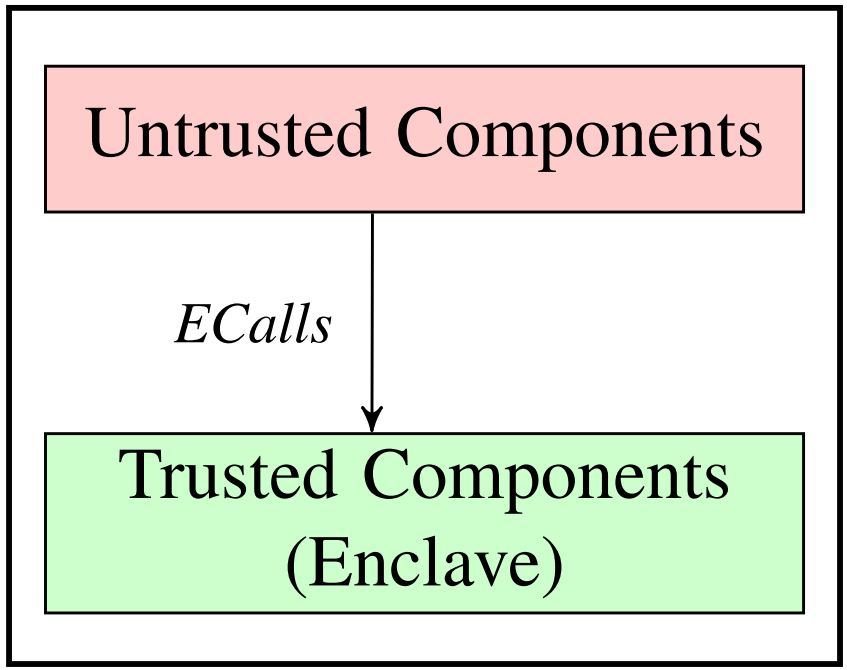
Securing Application in Untrusted OS



Intel Software Guard eXtension (SGX)

WAYNE STATE UNIVERSITY

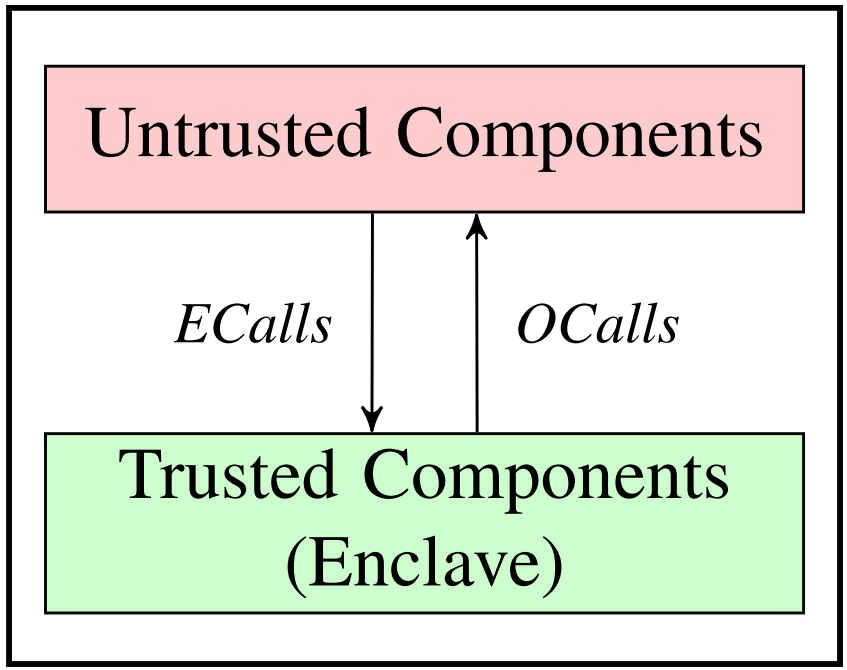
Securing Application in Untrusted OS



Intel Software Guard eXtension (SGX)

WAYNE STATE UNIVERSITY

Securing Application in Untrusted OS



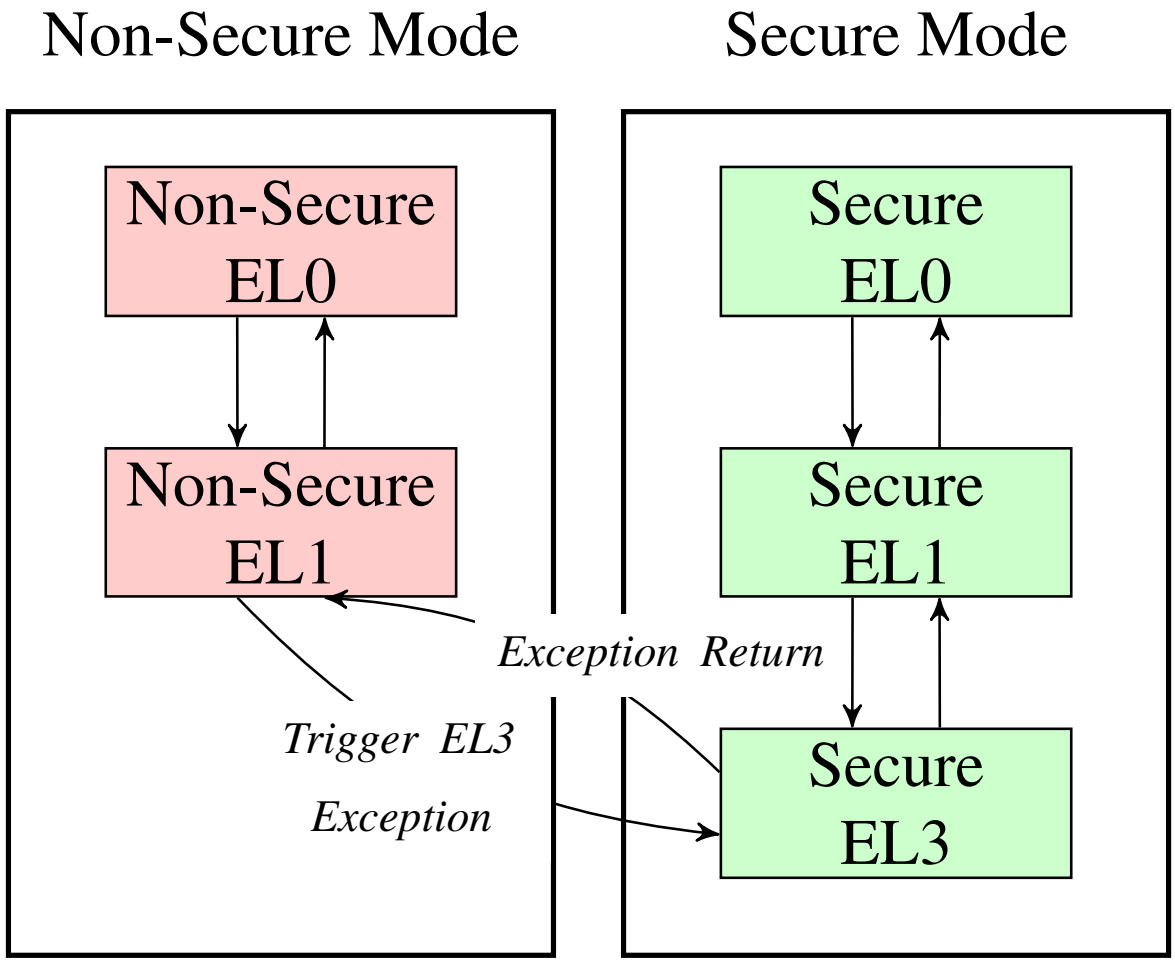
ARM TrustZone Technology

WAYNE STATE UNIVERSITY

ARM proposed the TrustZone Technology [7] since ARMv6 around 2002.

- The CPU has secure and non-secure states.
- The RAM is partitioned to secure and non-secure regions.
- The interrupts are assigned into the secure or non-secure group.
- Hardware peripherals can be configured as secure access only.

ARM TrustZone Technology



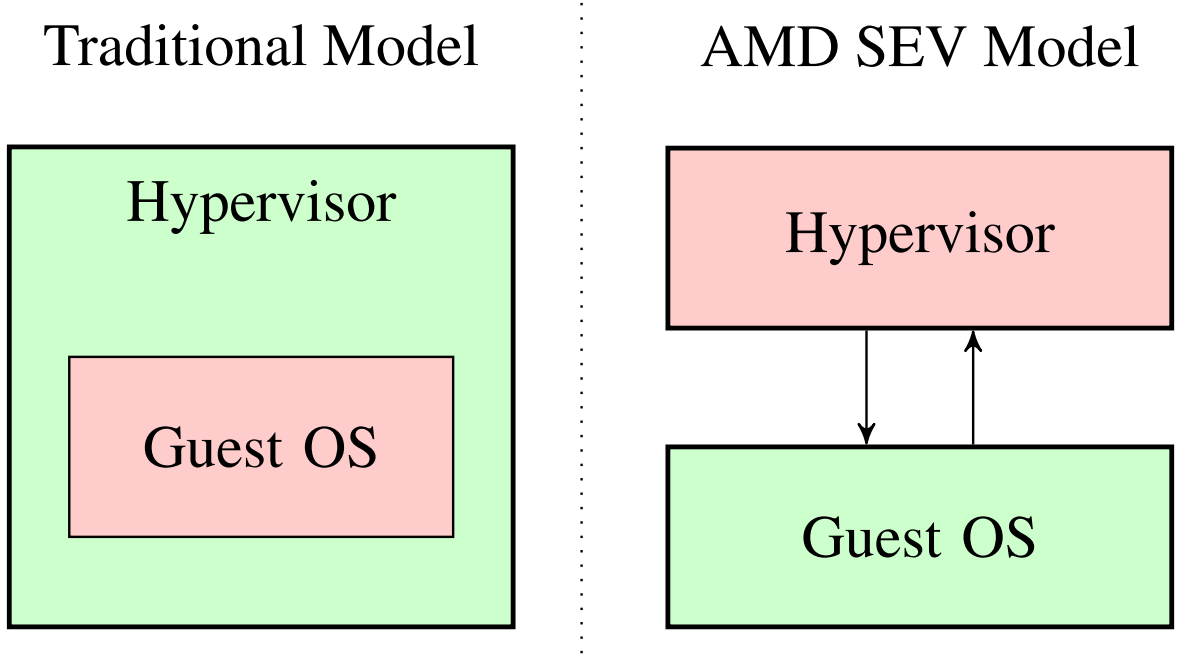
AMD Secure Encrypted Virtualization Technology

WAYNE STATE UNIVERSITY

AMD Secure Encrypted Virtualization (SEV) [8, 9] Technology is released with AMD Secure Memory Encryption (SME) in 2016.

- Protecting the VM memory space from the hypervisor.
- Based on AMD Memory Encryption Technology and AMD Secure Processor.
  - Memory Encryption: An AES 128 encryption engine inside the SoC.
  - Secure Processor: A 32-bit ARM Cortex-A5 with TrustZone technology.
- Modification to the application is NOT required.

AMD Secure Encrypted Virtualization Technology



Outline

WAYNE STATE UNIVERSITY

- Introduction
- Trusted Execution Environment (TEE)
  - Intel Software Guard eXtension (SGX)
  - ARM TrustZone Technology
  - AMD Secure Encrypted Virtualization Technology
- Edge Computing with TEE
- Conclusion and Future Work

How to secure the Edge nodes?

Securing the Edge Computing

WAYNE STATE UNIVERSITY

How to secure the Edge nodes?

- Secure the data and computation
  $\Rightarrow$ Using existing TEEs

- Accommodate to heterogeneous Edge nodes
  $\Rightarrow$ Adopting heterogeneous TEEs on different platforms

Performance Concerns

- The switch between the trusted and untrusted components should be efficient.
- The computing power inside the trusted component should be high.
- Introducing the trusted component should not affect the performance of the untrusted components.

Edge Computing with Intel SGX

WAYNE STATE UNIVERSITY

- Testbed Specification
  - Intel Fog Node, which is designed specifically for Fog Computing.
  - Hardware: An octa-core Intel Xeon E3-1275 processor.
  - Software: Tianocore BIOS and 64-bit Ubuntu 16.04.

Experiment Setup

- Context Switch: Use `RDTSC` instruction to record the time consumption of a pair of `ECall` and `OCall` with different parameter sizes.
- Secure Computation: Calculate MD5 of a pre-generated random string with 1024 characters inside the enclave, and record the time consumption.
- Overall Performance: Trigger a secure computation every one second, and use GeekBench [10] to measure the performance score.

Table: Context Switching Time of Intel SGX on the Fog Node (μs).

Buffer Size	Mean	STD	95% CI
0 KB	2.039	0.066	[2.035, 2.044]
1 KB	2.109	0.032	[2.107, 2.111]
4 KB	2.251	0.059	[2.247, 2.254]
8 KB	2.362	0.055	[2.359, 2.366]
16 KB	2.714	0.036	[2.712, 2.716]

Table: Time Consumption of MD5 (μs).

CPU Mode	Mean	STD	95% CI
Normal	4.734	0.095	[4.728, 4.740]
Enclave	6.737	0.081	[6.732, 6.742]

Table: Performance Score by GeekBench.

Sensitive Computation	Mean	STD	95% CI
No	4327.33	17.124	[4323.974, 4330.686]
Yes	4306.46	14.850	[4303.550, 4309.371]

- Testbed Specification
  - ARM Juno v1 development board, which represents ARM's official design purpose.
  - Hardware: A dual-core 800 MHZ Cortex-A57 cluster and a quad-core 700 MHZ Cortex-A53 cluster.
  - Software: ARM Trusted Firmware (ATF) [11] v1.1 and Android 5.1.1.

Experiment Setup

- Context Switch: Use Performance Monitor Unit (PMU) to record the time consumption of the context switch caused by `SMC` instruction.
- Secure Computation: Calculate MD5 of a pre-generated random string with 1024 characters in secure mode, and record the time consumption.
- Overall Performance: Trigger a secure computation every one second, and use GeekBench to measure the performance score.

Table: Context Switching Time of ARM TrustZone (μs).

Step	Mean	STD	95% CI
Non-secure to Secure	0.135	0.001	[0.135, 0.135]
Secure to Non-secure	0.082	0.003	[0.082, 0.083]
Overall	0.218	0.005	[0.218, 0.219]

Table: Time Consumption of MD5 (μs).

CPU Mode	Mean	STD	95% CI
Non-secure	8.229	0.231	[8.215, 8.244]
Secure	9.670	0.171	[9.660, 9.681]

Table: Performance Score by GeekBench.

Sensitive Computation	Mean	STD	95% CI
No	984.70	1.878	[984.332, 985.068]
Yes	983.44	3.273	[982.799, 984.082]

Edge Computing with AMD SEV

WAYNE STATE UNIVERSITY

- Testbed Specification
  - A customized machine with AMD EPYC-7251 CPU.
  - Hardware: 8 physical cores and 16 logic threads.
  - Software: Ubuntu 16.04.5 with SEV-enabled Linux kernel 4.15.10 and KVM 2.5.0.

Experiment Setup

- Context Switch: Use `RDTSC` instruction to record the time consumption of the context switch caused by `VMMCALL` instruction.
- Secure Computation: Calculate MD5 of a pre-generated random string with 1024 characters inside the guest, and record the time consumption.
- Overall Performance: Trigger a secure computation every one second, and use GeekBench to measure the performance score.

Edge Computing with AMD SEV

WAYNE STATE UNIVERSITY

- Context switch in AMD SEV takes about 3.09 μs.

Table: Time Consumption of MD5 (μs).

CPU Mode	Mean	STD	95% CI
Guest OS	3.66	0.126	[3.602, 3.720]
Host OS	0.70	0.005	[0.697, 0.702]

Table: Performance Score by GeekBench.

Sensitive Computation	Mean	STD	95% CI
No	3425.05	41.016	[3417.011, 3433.089]
Yes	3283.15	32.772	[3276.727, 3289.573]

Edge Computing with TEE

WAYNE STATE UNIVERSITY

- The context switch in all tested TEEs is efficient.
- The computing power in the TEEs provided by ARM TrustZone is lower than that out of the TEEs.
- The overall performance overhead of involving Intel SGX, ARM TrustZone, and AMD SEV in Edge Computing is 0.48%, 0.13%, and 4.14%, respectively.

Outline

WAYNE STATE UNIVERSITY

- Introduction
- Trusted Execution Environment (TEE)
  - Intel Software Guard eXtension (SGX)
  - ARM TrustZone Technology
  - AMD Secure Encrypted Virtualization Technology
- Edge Computing with TEE
- Conclusion and Future Work

Conclusion and Future Work

WAYNE STATE UNIVERSITY

- The hardware-assisted TEEs provided by different hardware vendors make it possible to fit the security requirement of heterogeneous Edge nodes.
- Deploying of these TEEs can efficiently improve the security of the Edge nodes with a low performance overhead.
- In the future, we will use Asylo project from Google, an open framework for enclave applications, as a base to further develop a generic framework for TEEs on Edge platforms.

References I

[1] B. Qi, L. Kang, and S. Banerjee, "A vehicle-based edge computing platform for transit and human mobility analytics," in *Proceedings of the 2nd ACM/IEEE Symposium on Edge Computing (SEC'17)*, 2017.

[2] Q. Zhang, Z. Yu, W. Shi, and H. Zhong, "Demo abstract: Evaps: Edge video analysis for public safety," in *Proceedings of the 1st IEEE/ACM Symposium on Edge Computing (SEC'16)*, 2016.

[3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, 2016.

[4] F. McKeen, I. Alexandrovich, A. Berenzon, C. V. Rozas, H. Shafi, V. Shanbhogue, and U. R. Savagaonkar, "Innovative instructions and software model for isolated execution." in *HASP@ ISCA*, 2013, p. 10.

[5] M. Hoekstra, R. Lal, P. Pappachan, V. Phegade, and J. Del Cuvillo, "Using innovative instructions to create trustworthy software solutions." in *HASP@ ISCA*, 2013, p. 11.

[6] F. McKeen, I. Alexandrovich, I. Anati, D. Caspi, S. Johnson, R. Leslie-Hurd, and C. Rozas, "Intel® software guard extensions (intel® SGX) support for dynamic memory management inside an enclave," in *Proceedings of the Hardware and Architectural Support for Security and Privacy 2016*. ACM, 2016, p. 10.

[7] ARM, "TrustZone security," http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.prd29-genc-009492c/index.html, 2009.

[8] D. Kaplan, J. Powell, and T. Woller, "Amd memory encryption," *White paper, Apr*, 2016.

[9] D. Kaplan, "AMD x86 memory encryption technologies." Austin, TX: USENIX Association, 2016.

[10] Primate Labs, "GeekBench," https://www.geekbench.com/, 2016.

[11] ARM, "Trusted firmware," https://github.com/ARM-software/arm-trusted-firmware, 2013.

Thank you!

WAYNE STATE UNIVERSITY

Questions?

weisong@wayne.edu

`http://compass.cs.wayne.edu`