CaSE: Cache-Assisted Secure Execution on ARM Processors

N1NG ZHANG, KUN SUN, WENJING LOU, TOM HOU

Who am I ?

- 10 years, working on different security products – data forensic, multi-level security systems

- did my undergrad @ Umass – middle of no where

- did my Ph.D @ VT in DC. – nice area, but I never got to go out !

- back to industry doing interesting things – or not

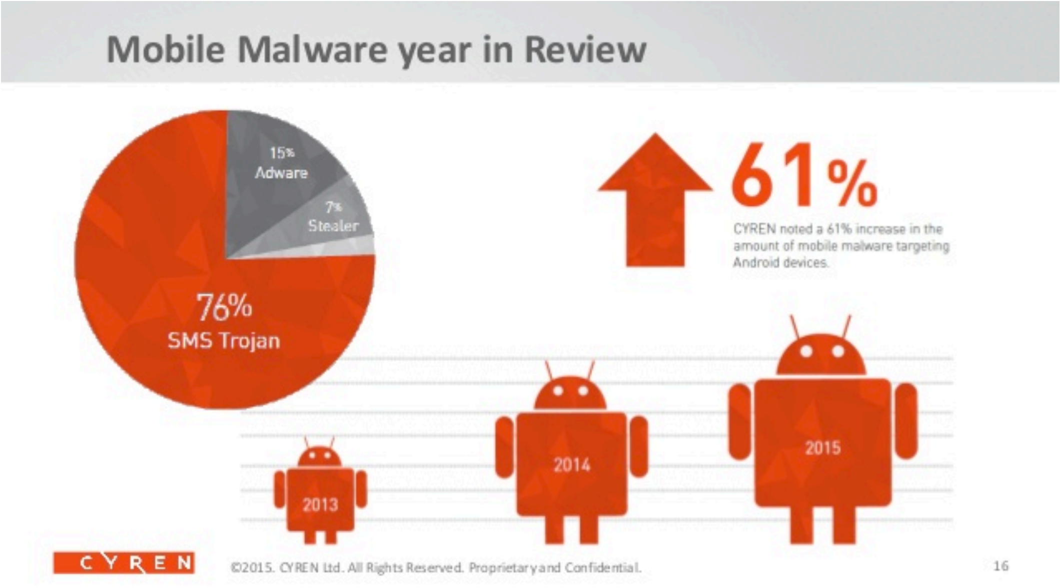
- lastly, I am also an adjunct assistant professor at the complex network and security research laboratory (CNSR) at Virginia Tech

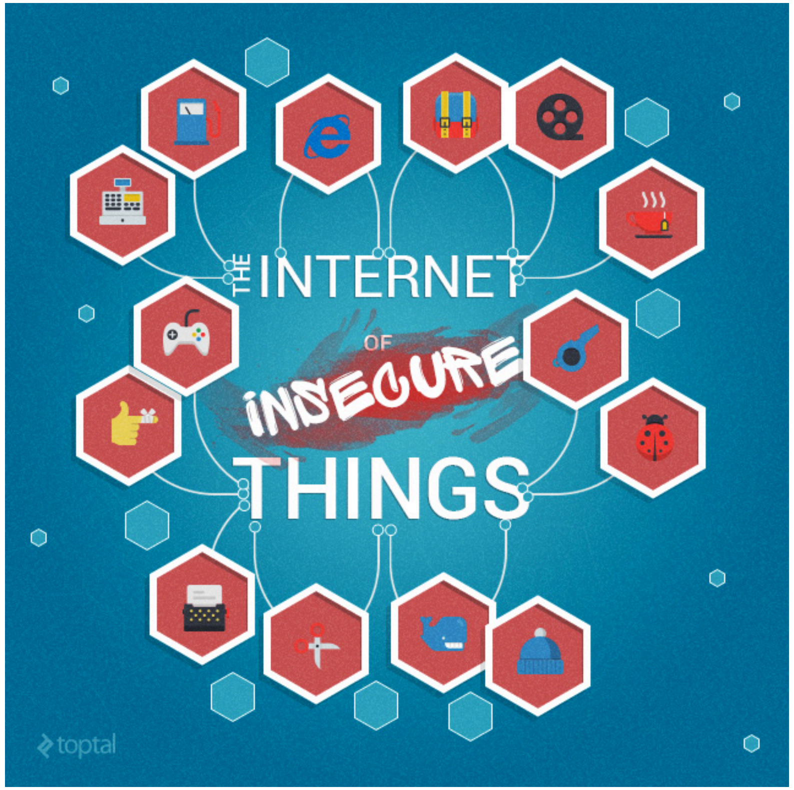
Talk Outline

- ✓ Motivation and Background – Why this work ?
- ✓ Threat Model – What are we defending against ?
- ✓ CaSE: Cache-Assisted Secure Execution – How does it work?
- ✓ CaSE highlight – Challenges ?
- ✓ Evaluation – How did we do ?
- ✓ Conclusion and future Work

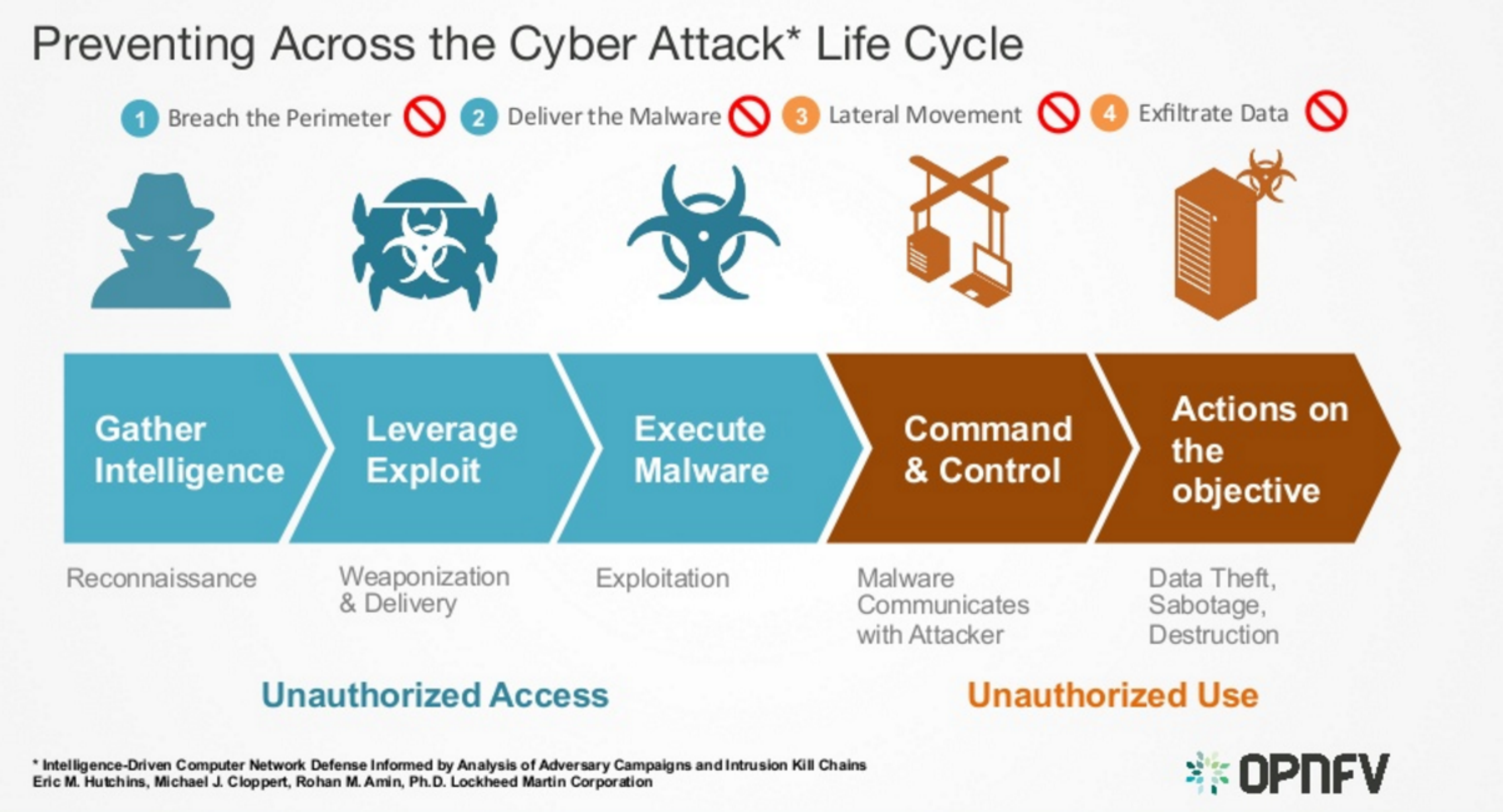
Cyber Attacks

Threat to Mobile devices

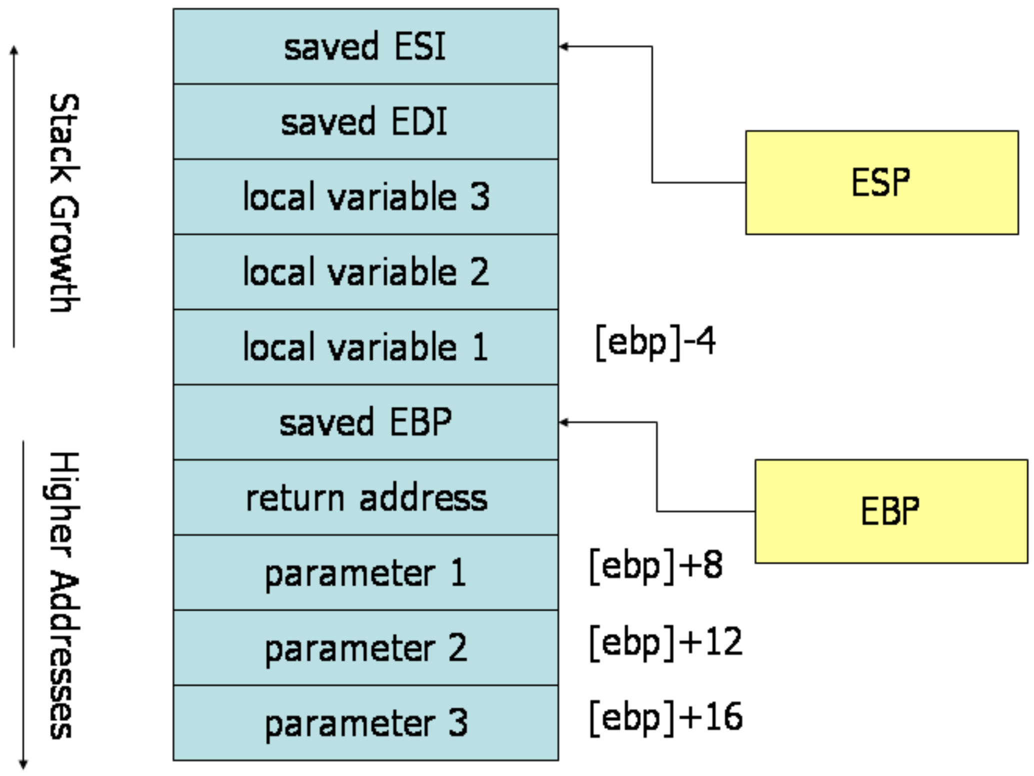




But how does it really work ?



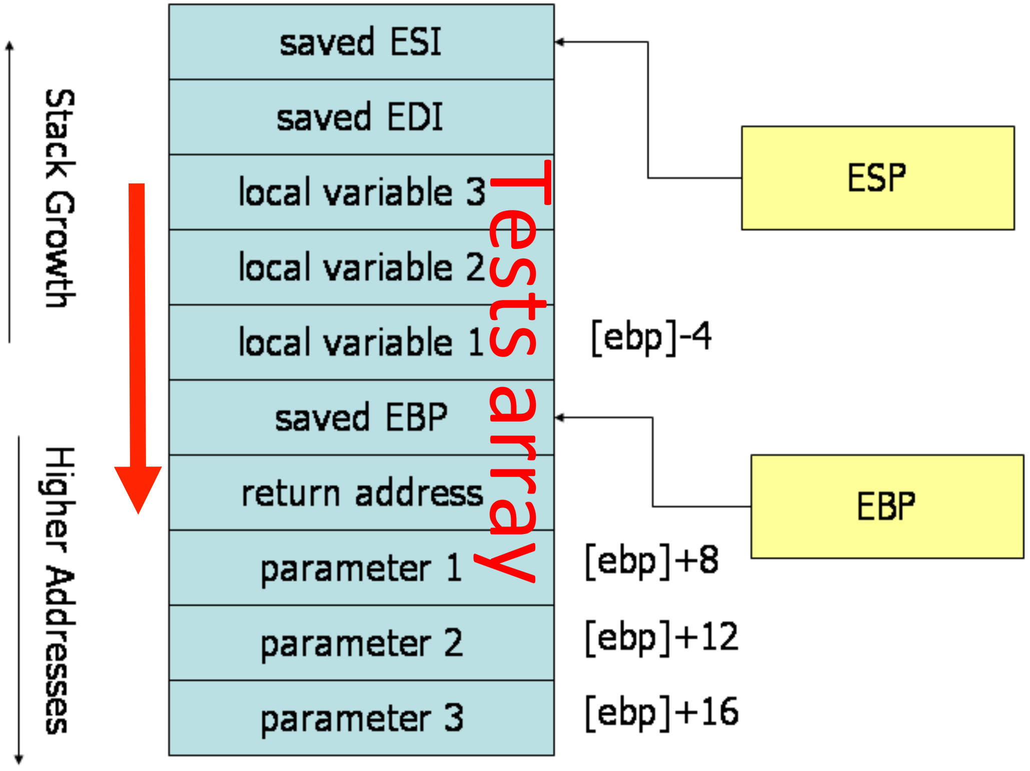
Buffer overflow - What is a software stack



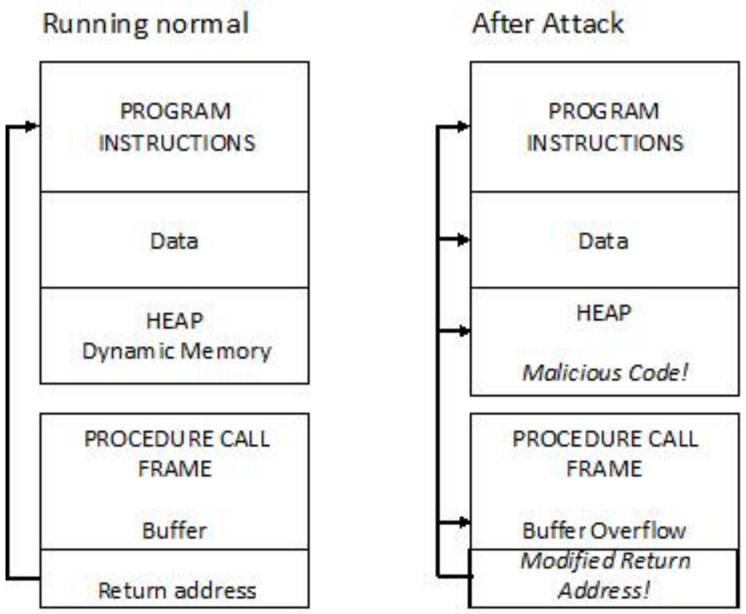
Software Exploits – Can you spot the bug ?

```
#include <iostream>
using namespace std;
  int main(void)
  {
      int tests[10];
      int test;
      int num_elems;

      cout << "How many numbers? ";
      cin >> num_elems;

      for (int i = 0; i < num_elems; i++)
          {
              cout << "Please type a number: ";
              cin >> test;
              tests[i]= test;
          }
      return 0;
  }
```

What happened ?

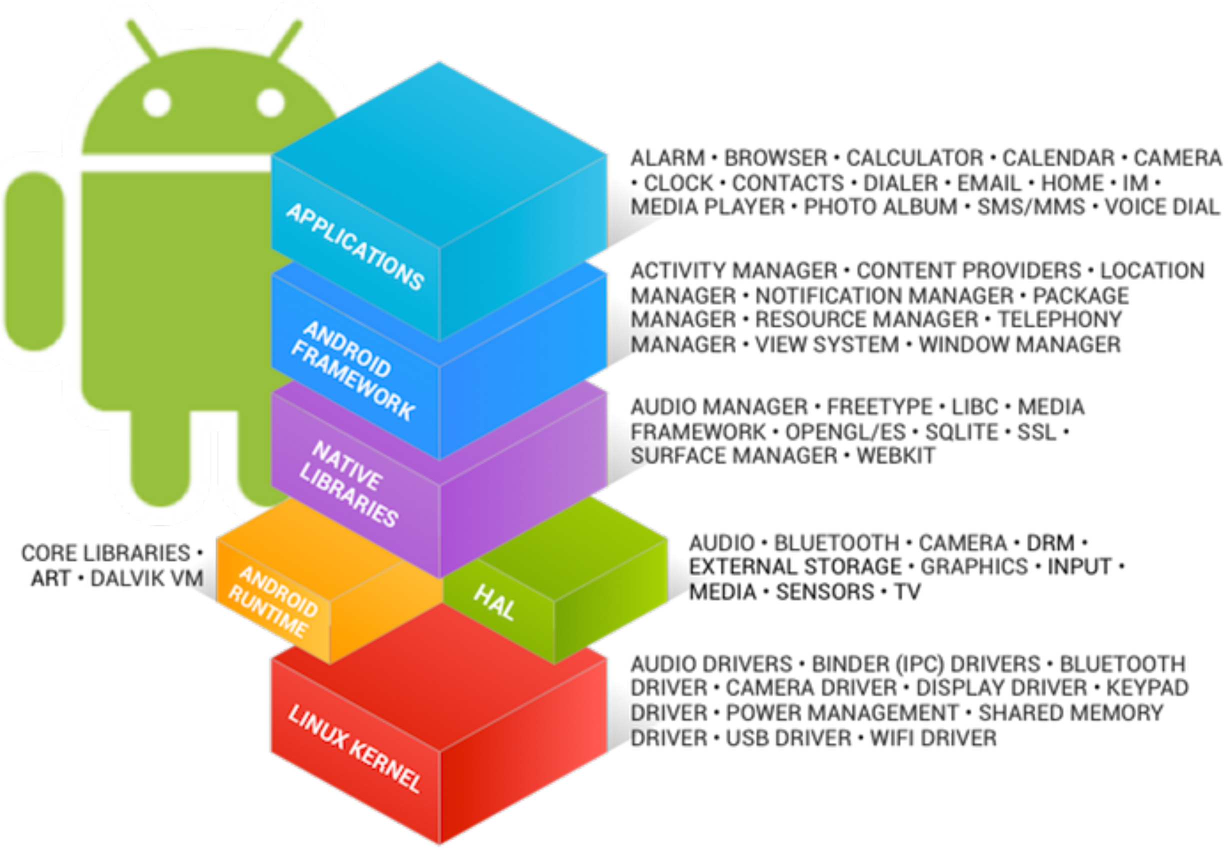


Before and After ☺



Attacker plants code that overflows buffer and corrupts the return address. Instead of returning to the appropriate calling procedure, the modified return address returns control to malicious code, located elsewhere in process memory.

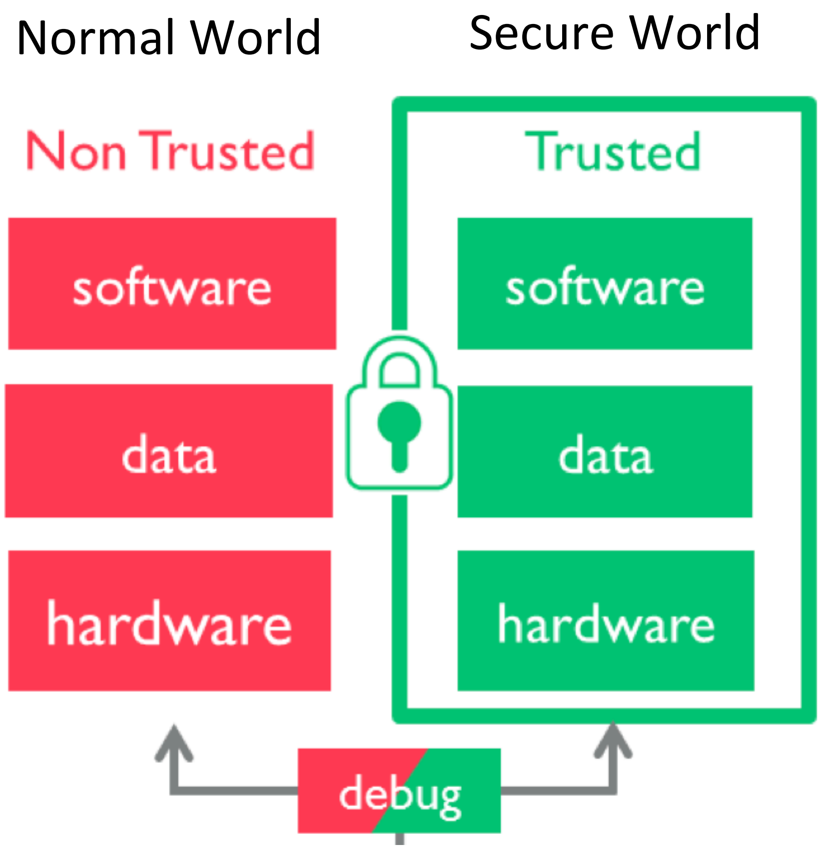
So are we doomed ? The best you can do ?



ARM TrustZone – Trusted Execution Environment (TEE)

System Wide Protection

- ✓Divides system resources into two worlds
- ✓Normal World runs the content rich OS
- ✓Secure World runs security critical services
- ✓The protection of resources includes
  - processor, memory and IO devices



Many Products use ARM TrustZone

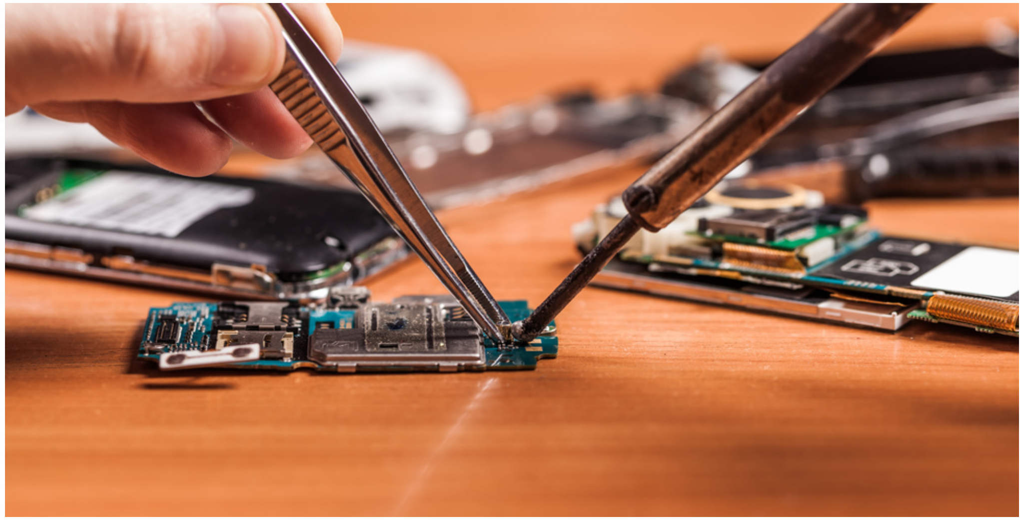
Samsung Knox

TRUSTONIC

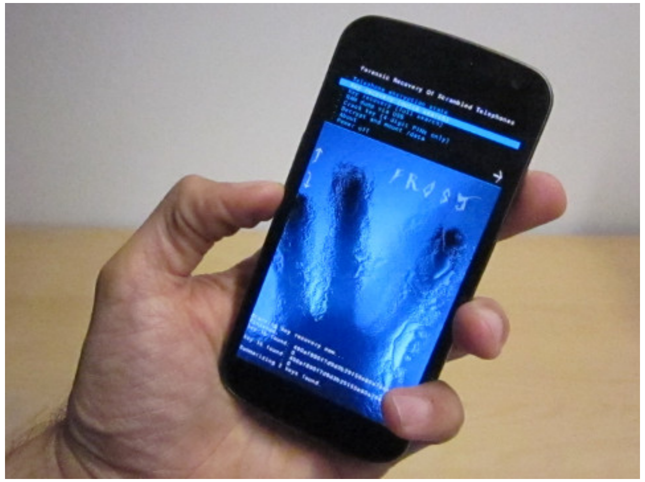
sierraware

Smart Devices Going Mobile

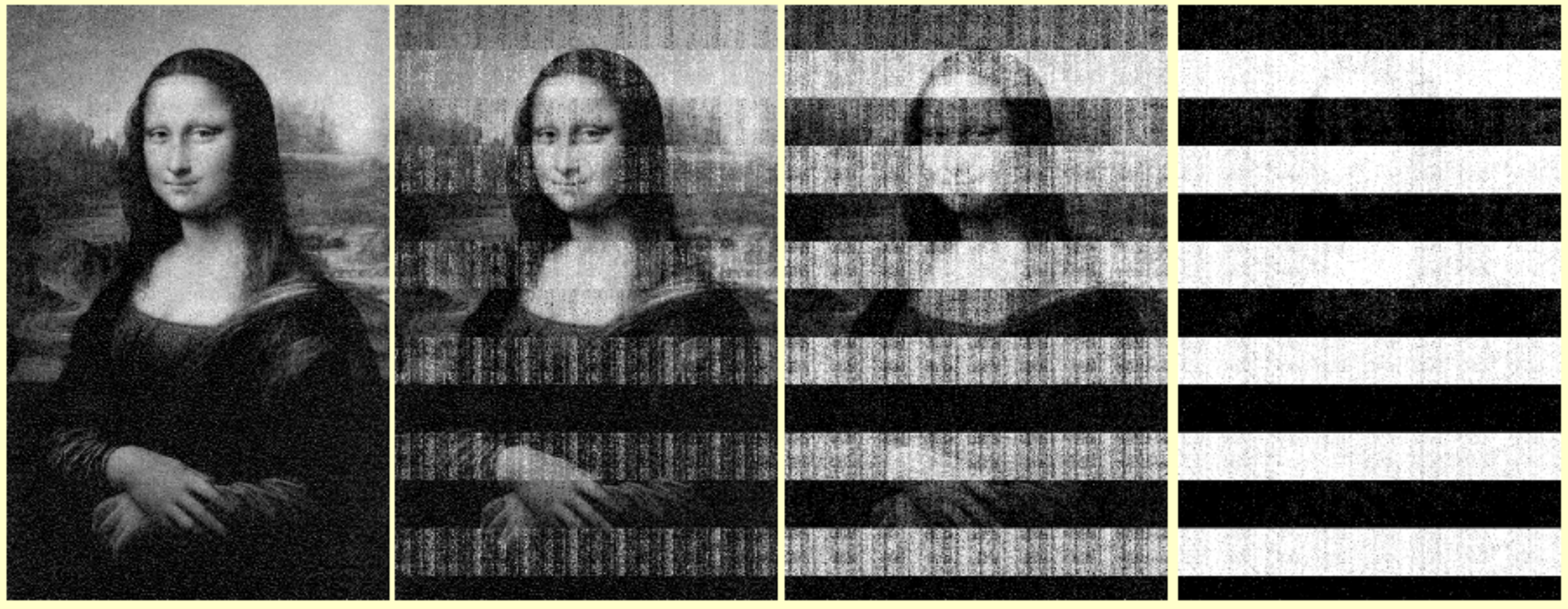
Physical Level Attack



Hardware Attacks - Cold Boot Attack



What can you recover ?



And whatever else that are in memory

Previous Works on Coldboot Defense

TRESOR	Sec	2011 – Register-based RAM-less AES encryption
Copker	NDSS	2014 – Cache-based RAM-less RSA encryption
PixelVault	CCS	2014 – GPU based RAM-less encryption
Sentry	ASPLOS	2015 – Cache-based RAM-less encryption
Mimosa	S&P	2015 – Transactional-based RAM-less encryption

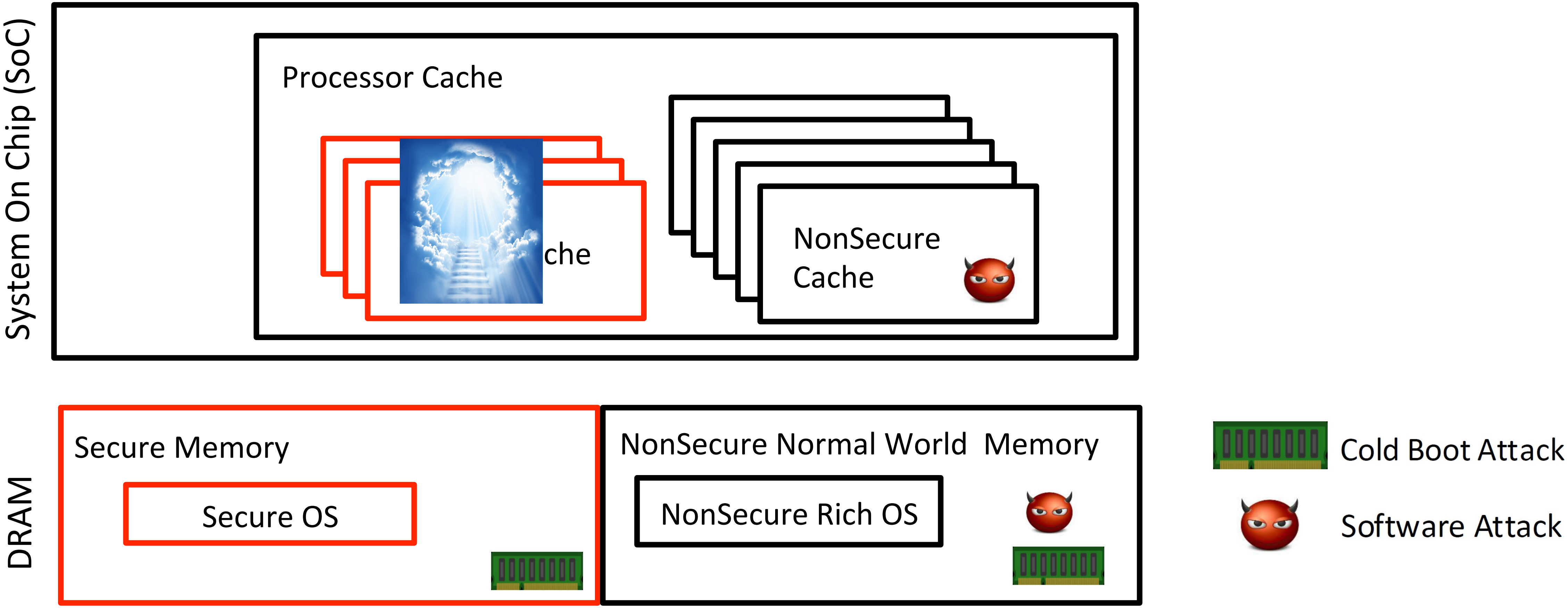
Multi-vector Adversary

Introducing CaSE - Goals

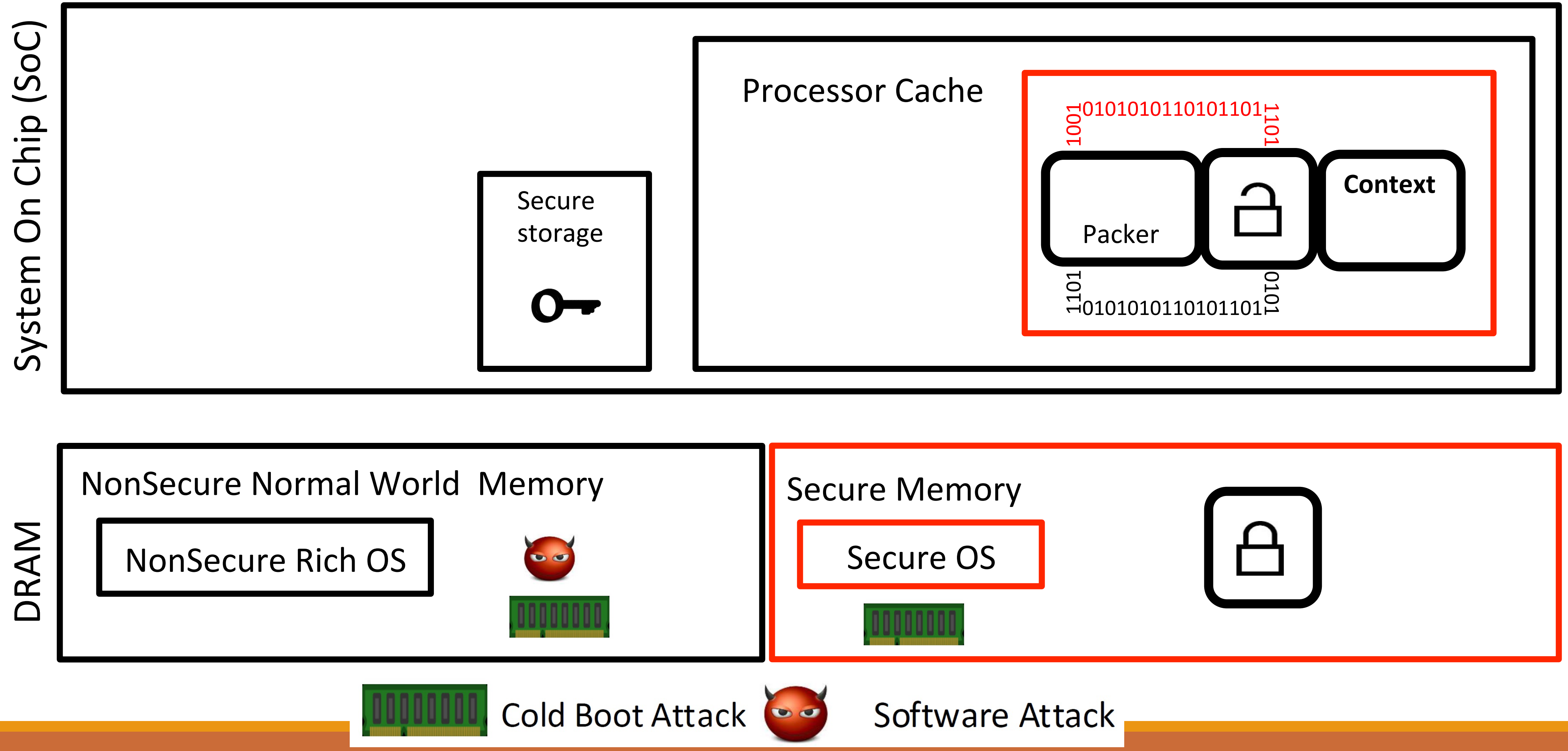
- ✓ Defense against Multi-Vector adversary
  - ✓ Physical memory disclosure attack – Cold boot
  - ✓ Compromised rich OS

- ✓ Provide confidentiality and integrity to both the code and data of the binaries in TEE
  - ✓ Confidentiality – Protects IP, secret code, sensitive data
  - ✓ Integrity – Program behavior

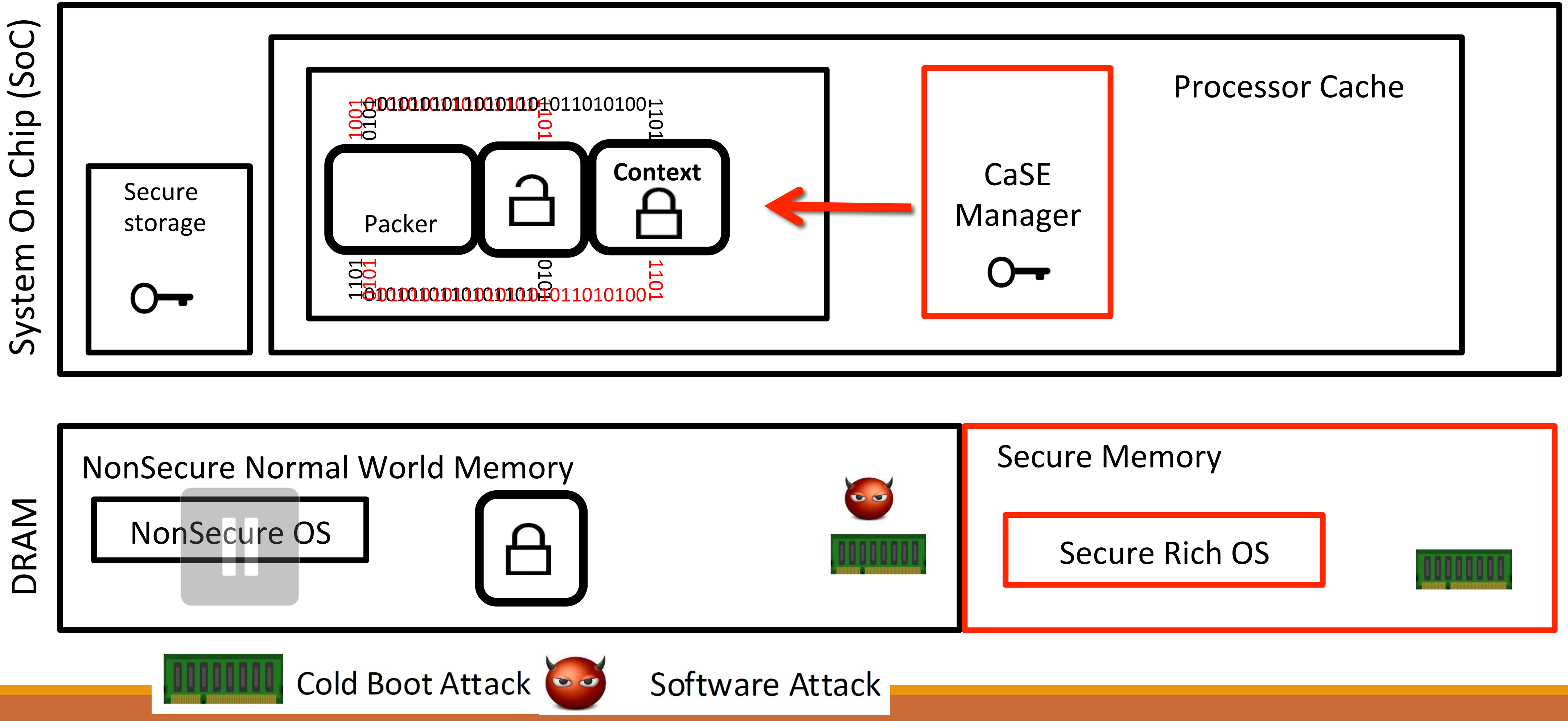
Threat Model



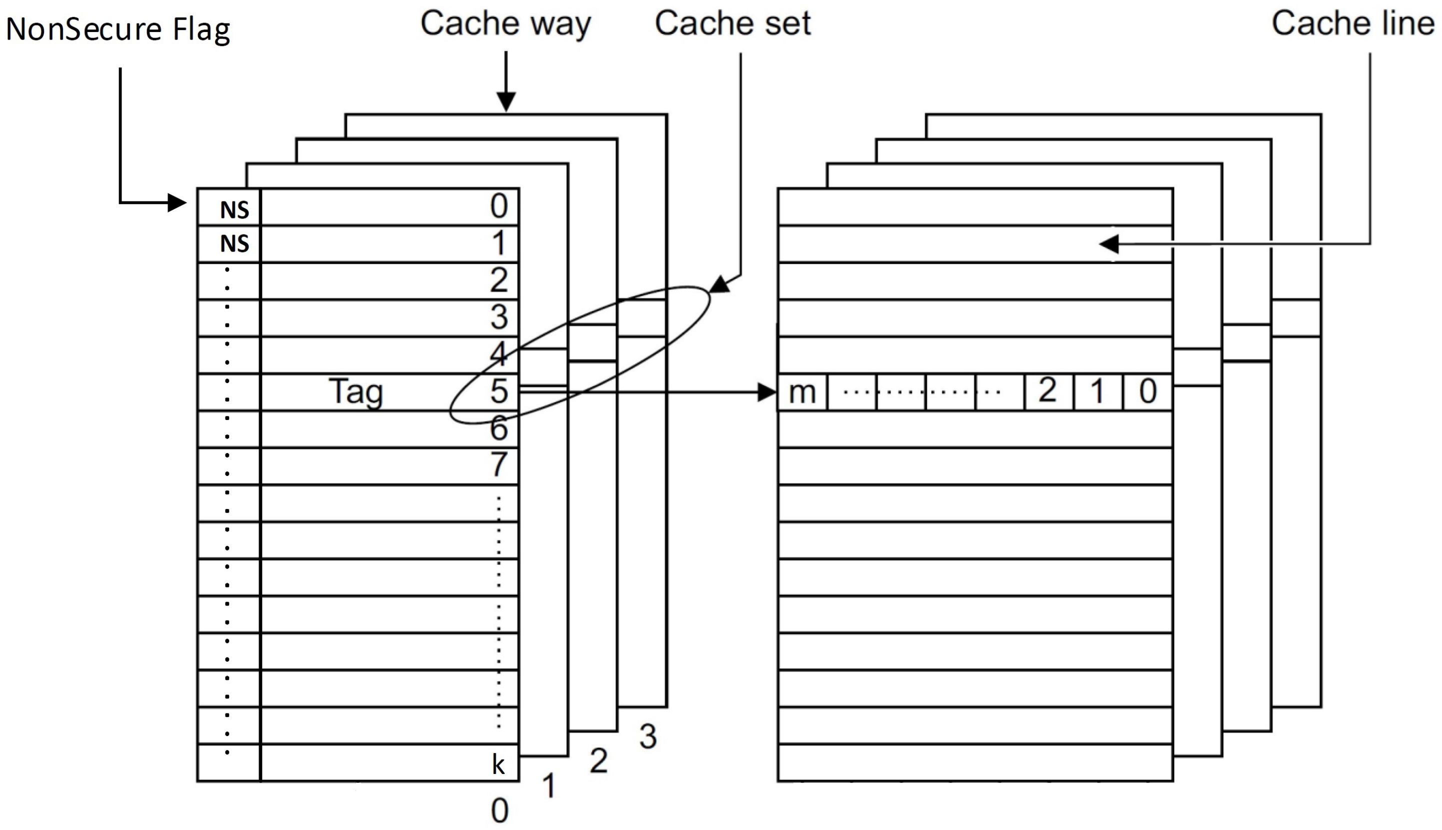
Case-Assisted Execution in Secure World



Case-Assisted Execution in Normal World



Cache Architecture Details



Controlling the Cache

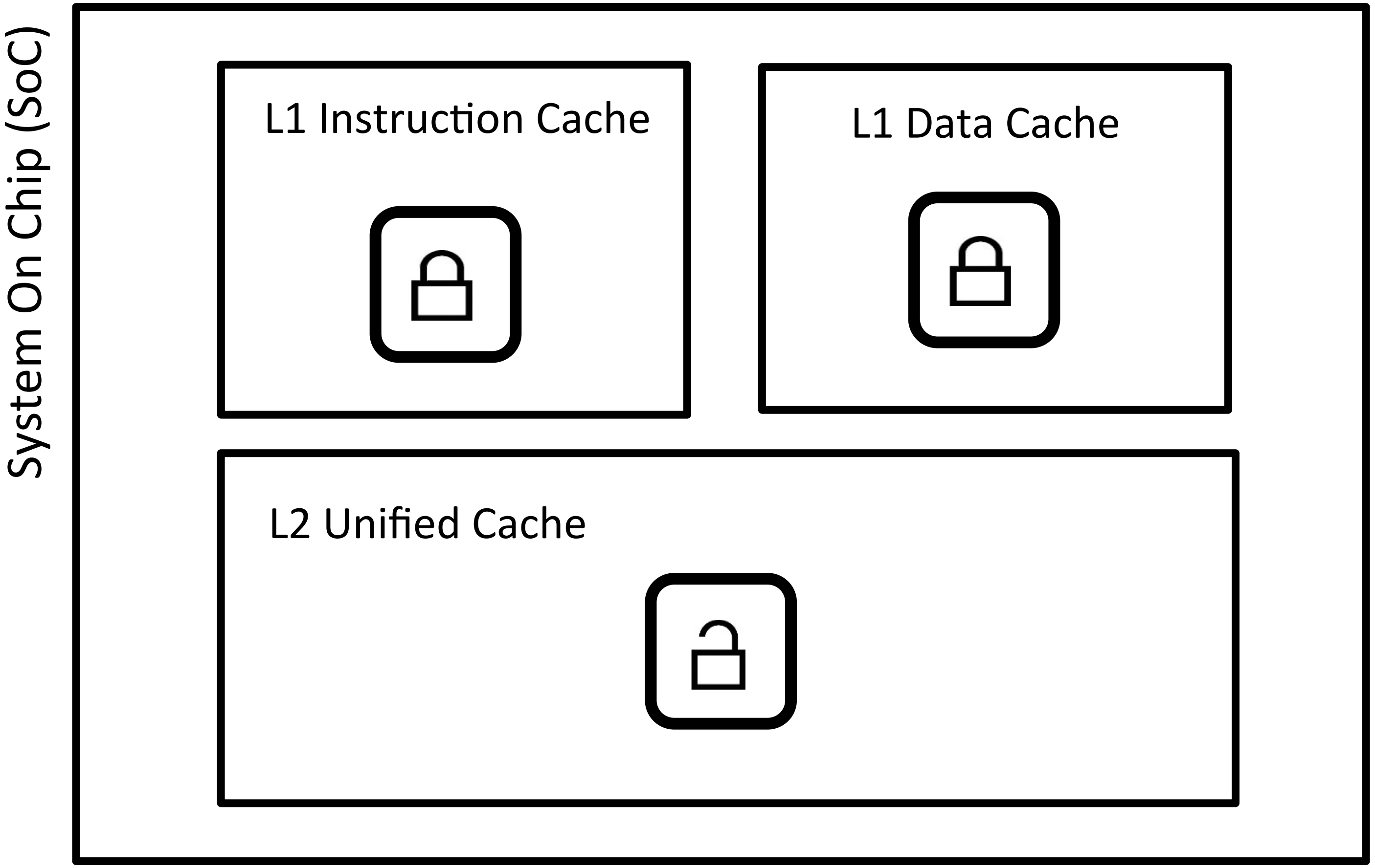
- ✓ Cache Locking is available through L2 cache lockdown CP15 coprocessor
- ✓ The granularity of locking is per cache way
- ✓ On Cortex-A8, which has 8 way total 256KB L2 unified cache

SoC-Bound Execution – Cache Locking

```
disable_local_irq();
enableCaching(memArea);
disableCaching(loaderCode);
disableCaching(loaderStack);
invalidate_cache(virtual address of memArea);
unlockWay(wayToFill);
lockWay(allWay XOR wayToFill);
while(has more to load in memArea)
      LDR r0, [memArea + i];
lockWay(wayToFill);
unlockWay(allWay XOR wayToFill);
```


root@raspberrypi:~/ > git clone --verbose git://github.com/Hexxeh/rpi-firmware.git --depth=1
Cloning into 'rpi-firmware'...
remote: Counting objects: 1673, done.
remote: Compressing objects: 100% (1347/1347), done.
remote: Total 1673 (delta 286), reused 1291 (delta 206)
Receiving objects: 100% (1673/1673), 27.08 MiB | 306 KiB/s, done.
Resolving deltas: 100% (286/286), done.
[1461.679215] ------------[cut here]------------
[1461.692804] kernel BUG at drivers/tty/vt/vt.c:2838!
[1461.706496] Internal error: Oops - BUG: 0 [#1] PREEMPT ARM
Entering kdb (current=0xc5e04360, pid 1326) Oops: (null)
due to oops @ 0xc0227cc8
Pid: 1326, comm: agetty
CPU: 0 Tainted: G C (3.6.11 #375)
PC is at con_shutdown+0x30/0x34
LR is at queue_release_one_tty+0x20/0x54
pc : [<c0227cc8>] lr : [<c02125e0>] psr: 60000013
sp : c7bedd20 ip : 00000000 fp : 00000000
r10: 00000000 r9 : 00000000 r8 : c78a41d8
r7 : 00000002 r6 : c7bec000 r5 : 00000000 r4 : c769a000
r3 : c0227c98 r2 : 00000000 r1 : 00000000 r0 : c769a000
Flags: nZCv IRQs on FIQs on Mode SVC_32 ISA ARM Segment user
Control: 00c5387d Table: 03e50008 DAC: 00000015
[<c0013a7c>] (unwind_backtrace+0x0/0xf0) from [<c0072a80>] (kdb_dumpregs+0x28/0x50)
[<c0072a80>] (kdb_dumpregs+0x28/0x50) from [<c0074e04>] (kdb_main_loop+0x3a8/0x6fc)
[<c0074e04>] (kdb_main_loop+0x3a8/0x6fc) from [<c00774e8>] (kdb_stub+0x154/0x380)
[<c00774e8>] (kdb_stub+0x154/0x380) from [<c006e61c>] (kgdb_handle_exception+0x1f8/0x668)
more> _

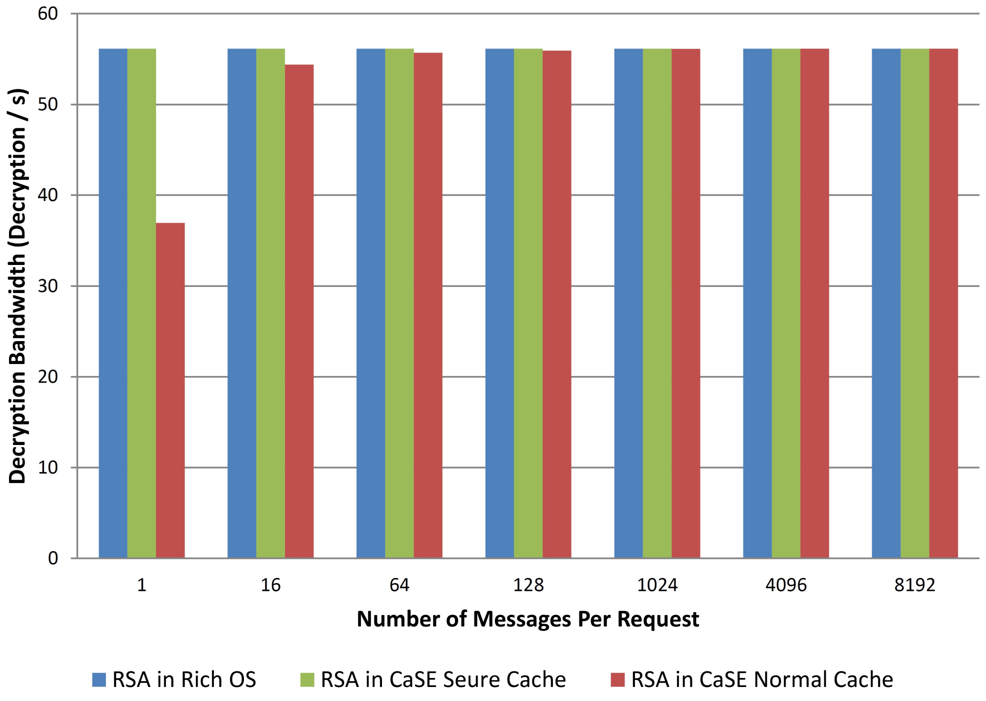
Self Modifying Program

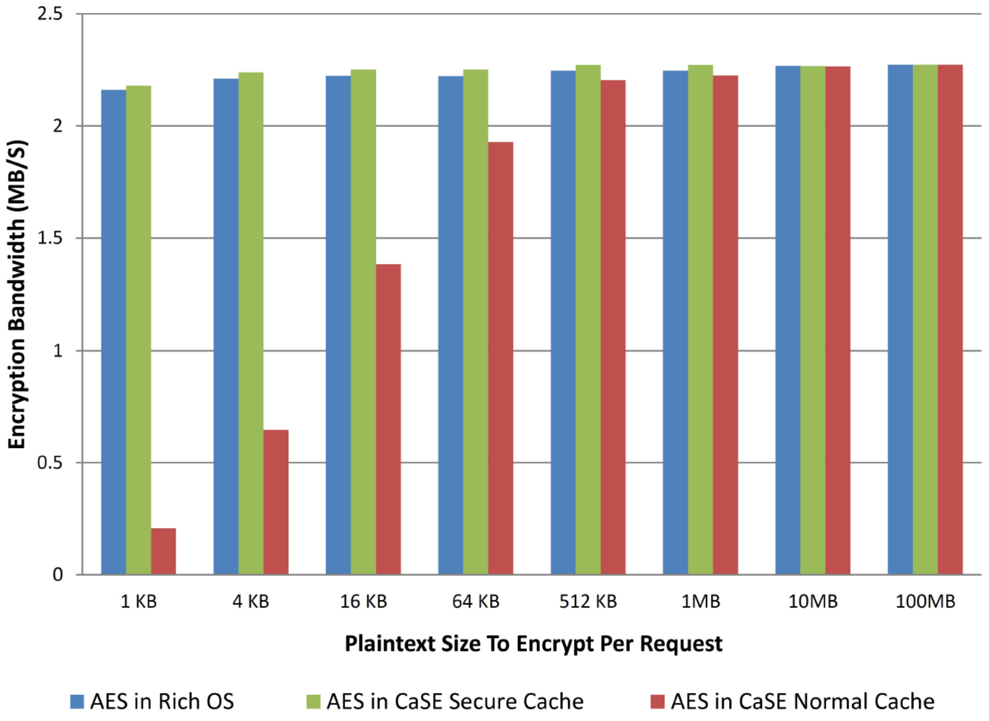


Evaluation
Feasibility of using Cache as Memory

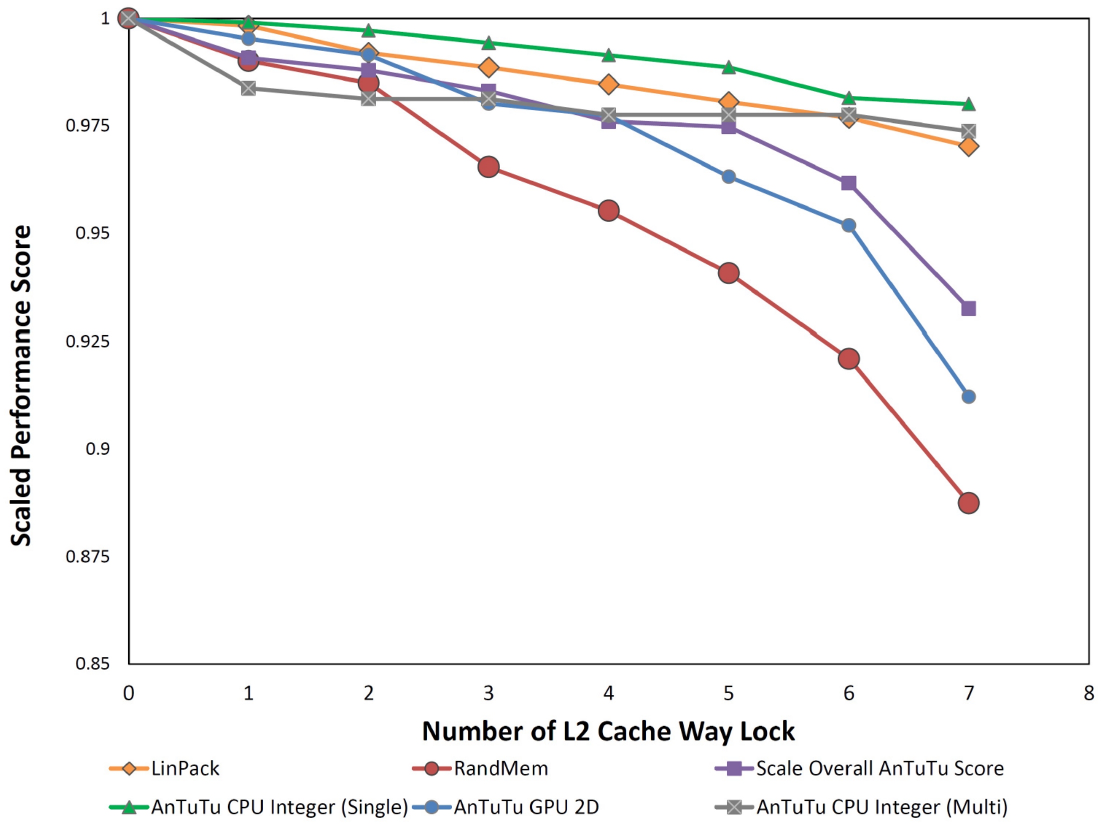
Application	Code+Data (KB)
AES	2.4
RSA	10
SHA1	5
CaSE Crypto Lib	17.4
Kernel Integrity Checker	6.6
CaSE Packer	2.8
Packed CaSE Crypto Lib	20.4
Packed Kernel Checker	9.5

Evaluation
Performance Impact to the Application





Performance Impact to the System



Conclusion

- ✓A secure cache-assisted SoC-bound execution framework
  - ✓Provide confidentiality and integrity to sensitive code and data of applications
  - ✓Protect against both software attacks and cold boot attack.
- ✓In the future, we would like to further study efficient method to provide OS support to the TEE.

What other things did I do ?

- Differential privacy in data mining - ICC 11

- Reverse engineer ASUS BIOS - Trusted Cloud Computing – CNS 14

- Anti-memory forensic framework – HIVES – ASIACCS 15

- Cache-based rootkits – EUROSP 16

- Case – Cached-assisted security execution – SP16

- Augmented reality authentication – TRUSTED – CCS16

Feel free to contact me at Ningzhang.info / ningzh@vt.edu