论文Review: OCRAM-Assisted Sensitive Data Protection on ARM-Based Platform (ESORICS 19') CCF-B

Dawei Chu[1,2,3], Yuewu Wang[1,2,3], **Lingguang Lei**[1,2,3 *], Yanchu Li[1,2,3], Jiwu Jing[4], and Kun Sun[5]

1,4 中国科学院大学 University of Chinese Academy of Sciences

2,3 中国科学院信息工程研究所 INSTITUTE OF INFORMATION ENGINEERING,CAS

5 GEORGE MASON UNIVERSITY

Jing Li

Apr. 25, 2021

辛丑三月十四

Huairou, Beijing

中国科学院信息工程研究所 INSTITUTE OF INFORMATION ENGINEERING,CAS

Executive summary

Problem：

- **TEE难防御硬件侧信道且具体实现有漏洞**
  - 基于cache的防御方法（CaSE）性能开销大，不够轻量
  - 现有片上存储解决方案不能有效加固敏感I/O信息保护

Key Idea：

- **用iRAM存储隔离关键安全信息取代L2 cache方案，配合软件层优化和工程直觉方案实现安全免疫**

Mechanisms ：

- **工程实现Oath: 增加iRAM的使用管理机制，修改原先load方法**

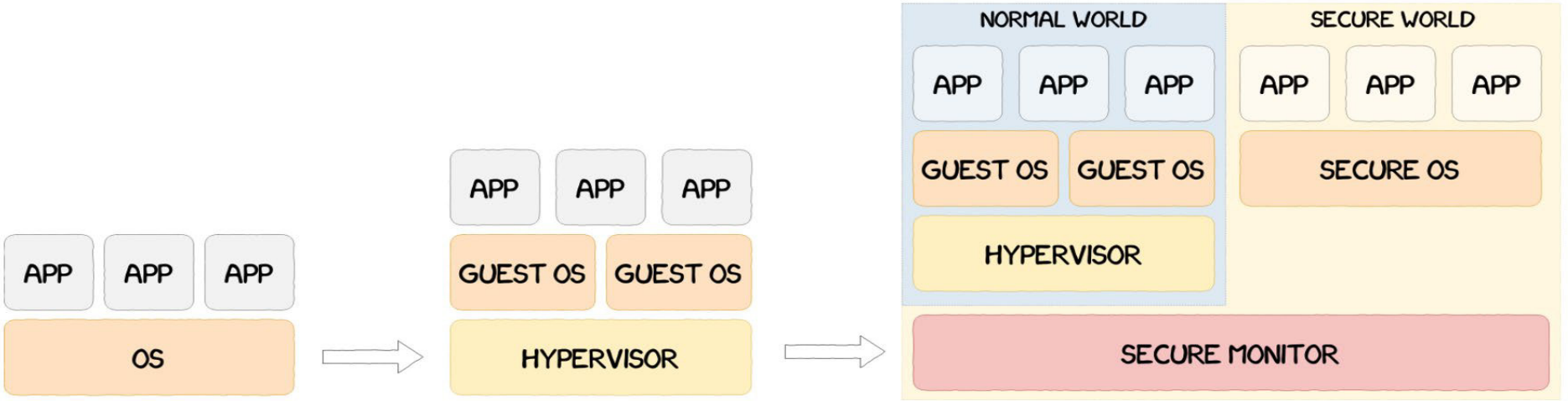
Results ：

- **靠700行代码，性能开销极低，对功能基本无影响**
- **成功免疫软硬件攻击，成功避免敏感数据特别是I/O数据泄露**

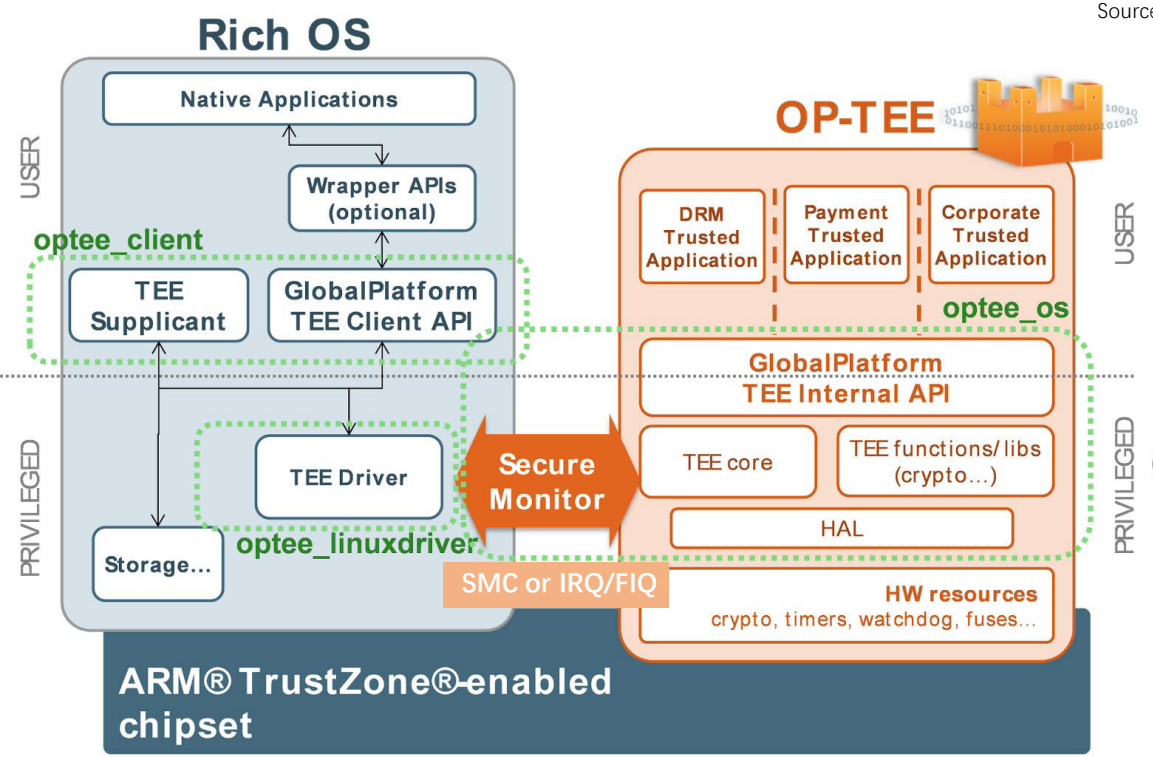
Overview

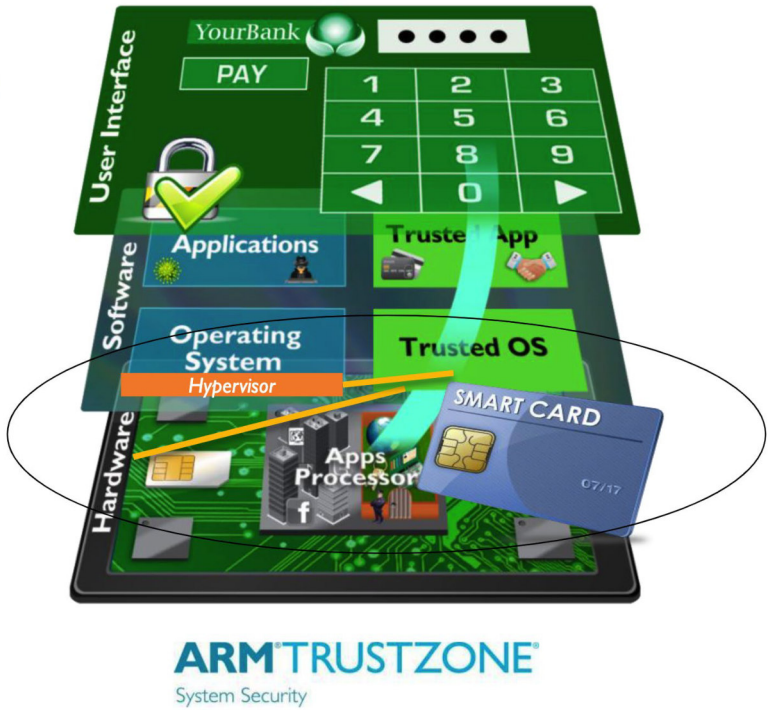
Background, Problem & Goal

Recap: TEE & OP-TEE

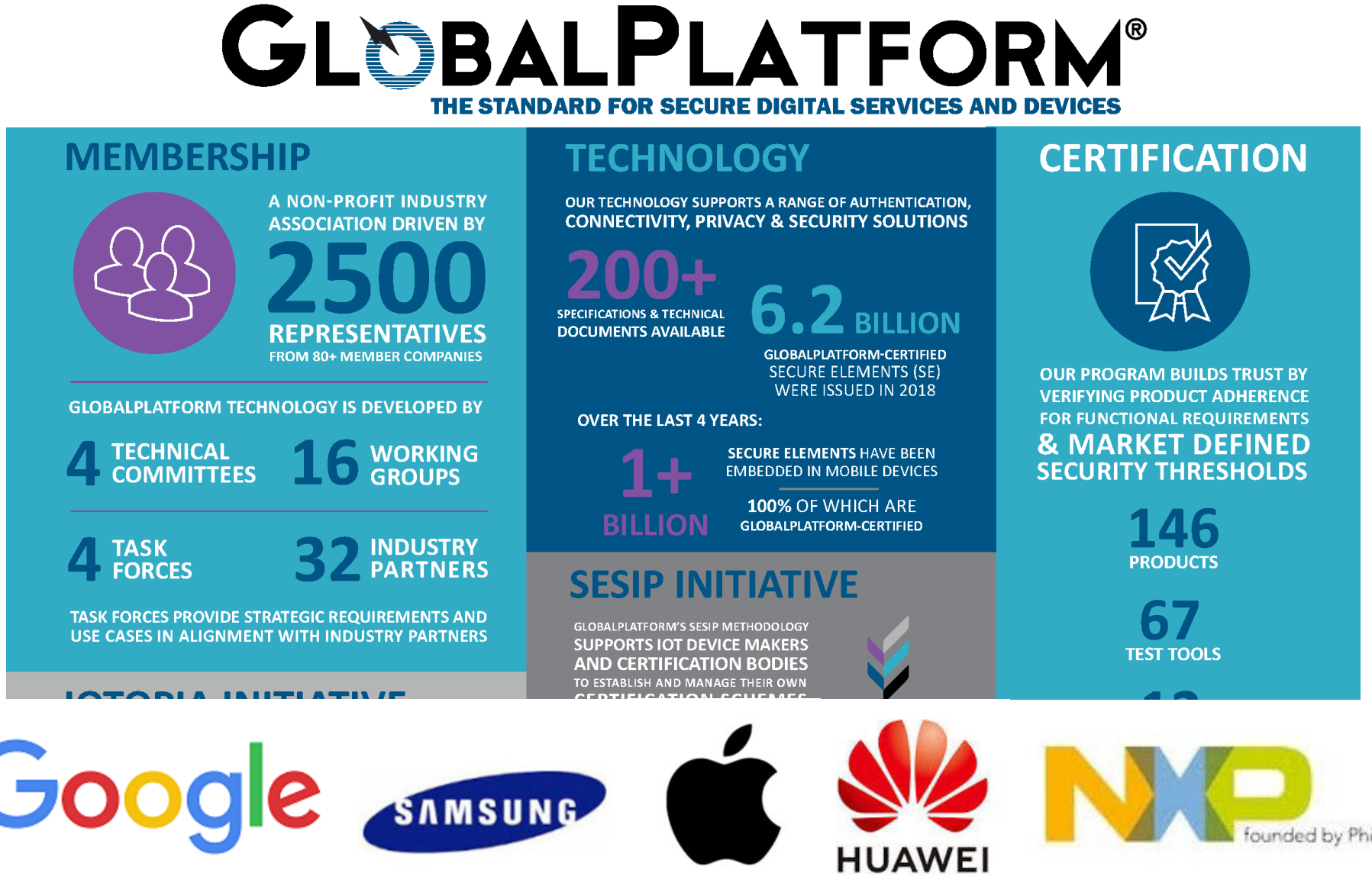


Source: https://blog.quarkslab.com/a-deep-dive-into-samsungs-trustzone-part-1.html





TEE已被工业界应用


GLOBALPLATFORM®
THE STANDARD FOR SECURE DIGITAL SERVICES AND DEVICES
MEMBERSHIP
A NON-PROFIT INDUSTRY ASSOCIATION DRIVEN BY
2500
REPRESENTATIVES
FROM 80+ MEMBER COMPANIES
GLOBALPLATFORM TECHNOLOGY IS DEVELOPED BY
4 TECHNICAL COMMITTEES
16 WORKING GROUPS
4 TASK FORCES
32 INDUSTRY PARTNERS
TASK FORCES PROVIDE STRATEGIC REQUIREMENTS AND USE CASES IN ALIGNMENT WITH INDUSTRY PARTNERS
TECHNOLOGY
OUR TECHNOLOGY SUPPORTS A RANGE OF AUTHENTICATION, CONNECTIVITY, PRIVACY & SECURITY SOLUTIONS
200+
SPECIFICATIONS & TECHNICAL DOCUMENTS AVAILABLE
6.2 BILLION
GLOBALPLATFORM-CERTIFIED SECURE ELEMENTS (SE) WERE ISSUED IN 2018
OVER THE LAST 4 YEARS:
1+ BILLION
SECURE ELEMENTS HAVE BEEN EMBEDDED IN MOBILE DEVICES
100% OF WHICH ARE GLOBALPLATFORM-CERTIFIED
SESIP INITIATIVE
GLOBALPLATFORM'S SESIP METHODOLOGY SUPPORTS IOT DEVICE MAKERS AND CERTIFICATION BODIES TO ESTABLISH AND MANAGE THEIR OWN
CERTIFICATION
OUR PROGRAM BUILDS TRUST BY VERIFYING PRODUCT ADHERENCE FOR FUNCTIONAL REQUIREMENTS & MARKET DEFINED SECURITY THRESHOLDS
146
PRODUCTS
67
TEST TOOLS
Google
SAMSUNG
HUAWEI
NXP
founded by Philips
www.globalplatform.org
UPDATED JUL 2020

Recap: Cold Boot Attack

Source: https://www.pinterest.com/pin/304204149827181943/

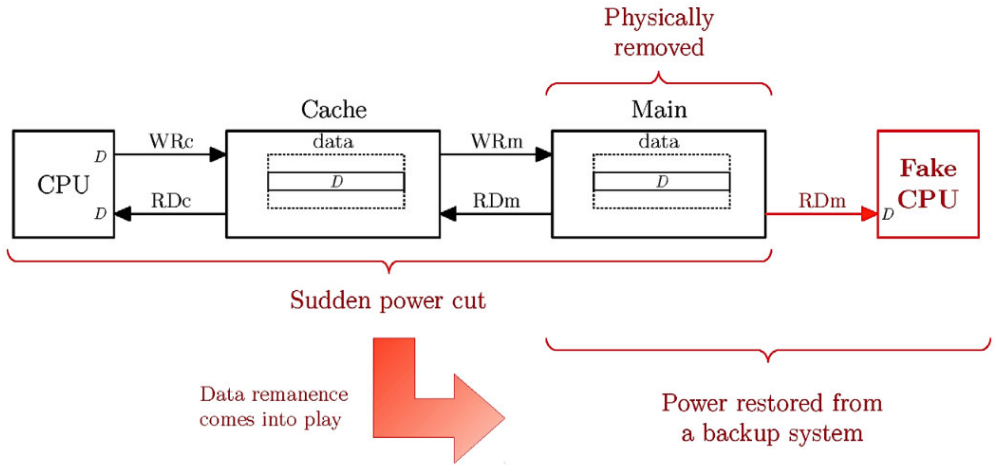
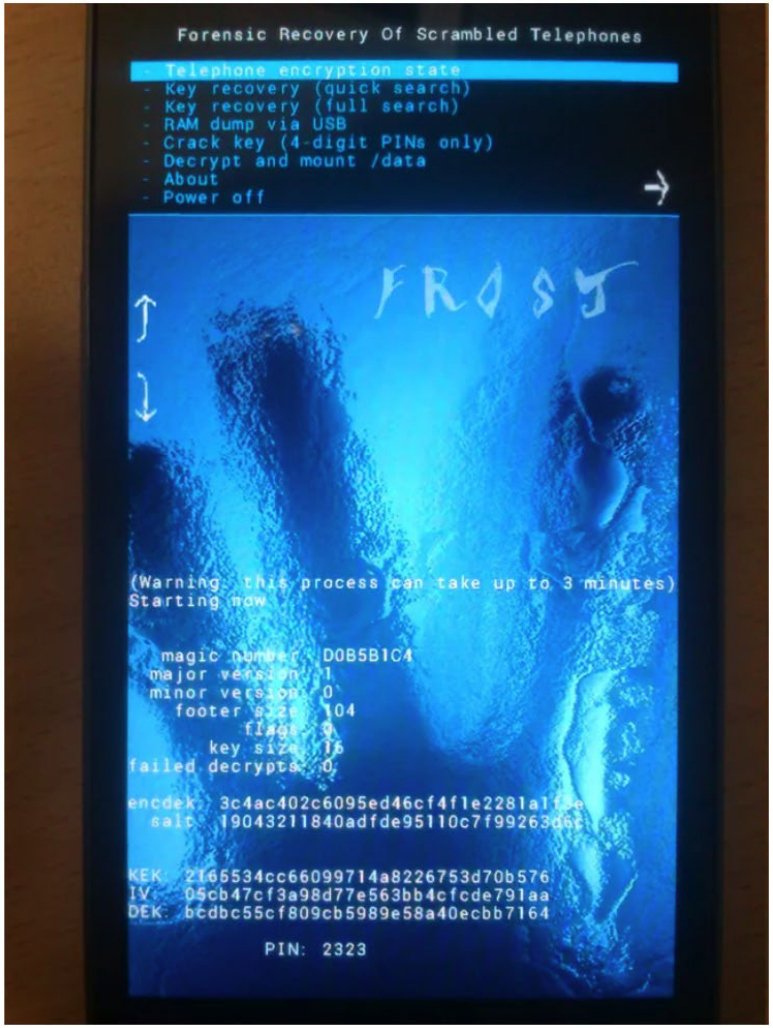


Fig. 10.3. *Cold-boot* attack on main memory.

In book: Advances in Microelectronics: Reviews, Vol. 2



The FROST method requires freezing an Android phone for roughly 1 hour.

Friedrich-Alexander University

Source: https://www.cs1.tf.fau.de/research/system-security-group/frost/

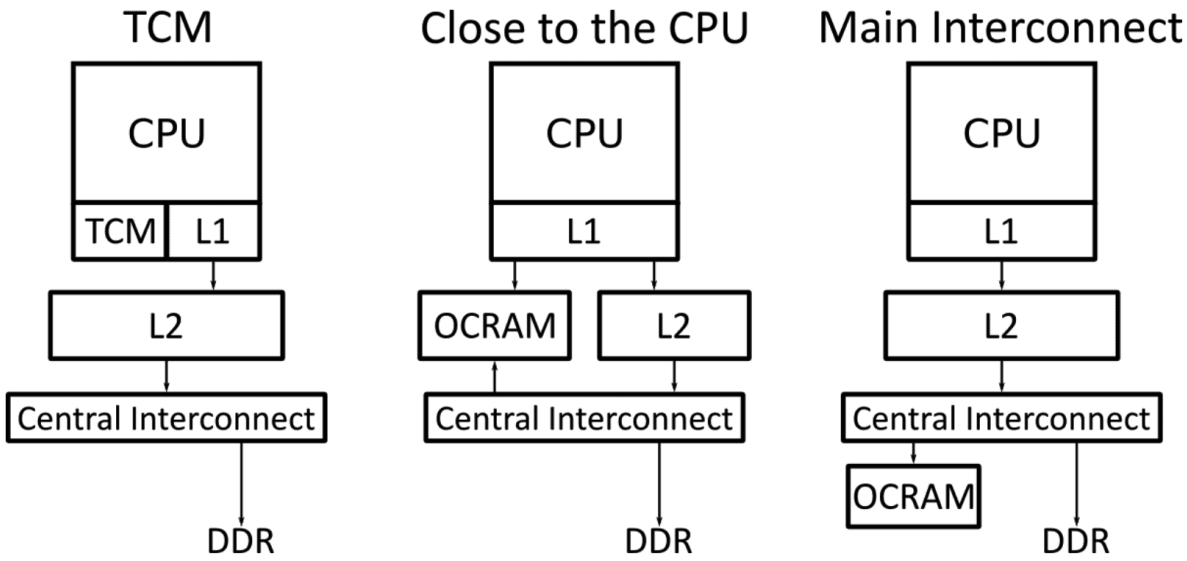
TEE的缺陷

- 难防御硬件攻击
  - Cold Boot Attack等侧信道
  - 总线监听攻击 (CPU to DRAM)
- 实现漏洞
  - 敏感数据直接明文存储在DRAM只读段中（数据机密性缺失）iRAM 或 TrustZone 几乎无法保护，攻击者可以直接分析静态图像。

已有片上存储安全解决方案的问题

- 能够加固上述TEE的部分缺陷
  - Cold Boot Attack
  - 总线监听攻击 (CPU to DRAM)
- 但是敏感I/O操作仍难全面防护
  - 基于软件层仍有漏洞

OCRAM, on-chip RAM (internal aka iRAM)



Diagrams of possible OCRAM location.

Source: https://www.jblopen.com/introduction-ocram/

体积：

- Normal 128 KB to 2 MB iRAM memory
- Cortex A8 and A9 ARM processors usually have 128 KB and 256 KB iRAM

- 常用于：
  - 存储如系统挂起/恢复代码，DDR频率修改代码等信息；
  - 加速多媒体处理
- 安全性：
  - 由原生机制：TrustZone Memory Adapter (TZMA) and TrustZone Protection Controller (TZPC)，利用之保护iRAM的安全性，TZMA分隔安全与非安全的空间，TZPC作为控制器

Goal

- ✓ 假定TrustZone的保护使得攻击者不能非法访问secure world
- ✓ 假定攻击者不能物理地窃听芯片上内存数据（侧信道）
- ✓ 假定敏感数据存储在可写内存

现有的片上存储解决TEE缺陷的方案无法解决软件攻击和I/O操作，较小改动硬件，实现一个软件方法，以保护敏感数据的机密性和完整性，包括用户的I/O数据，抵抗软件攻击和物理内存泄露攻击

Input：i.e. 加密密钥

Display: i.e.触屏输入的PIN码

Key Approach & Ideas

iRAM-assisted

- Approach：
  - 借助iRAM硬件，建立相关软件机制，实现将敏感数据存储在iRAM，在安全CPU中运行处理，实现免疫
  - Do it
- Reason：
  - 比cache好用，和DRAM类似
  - ARM中广泛应用

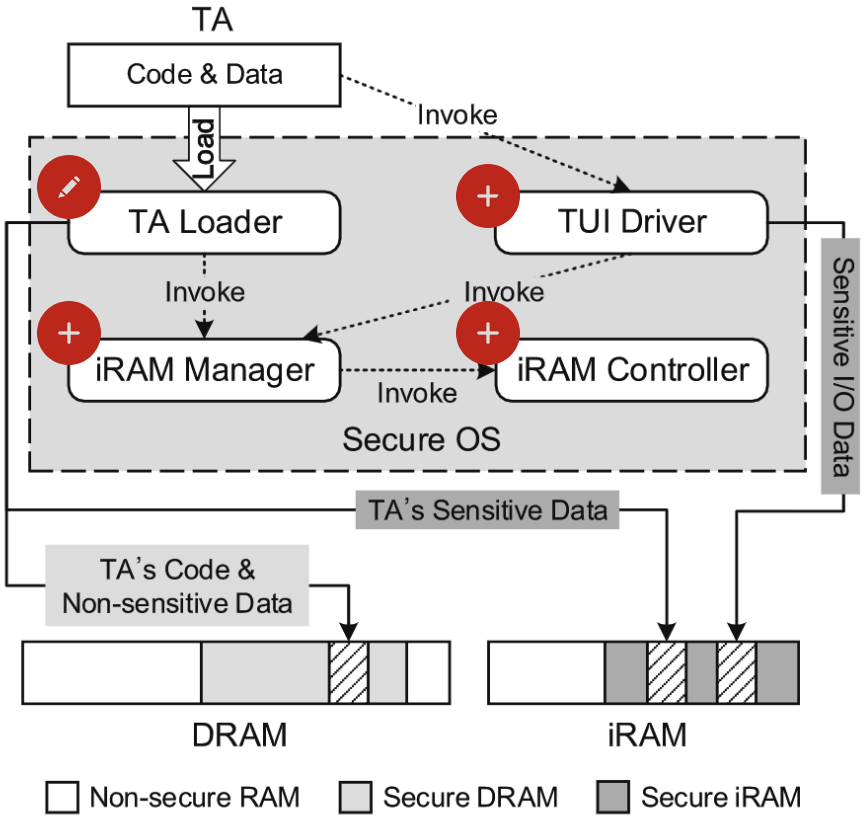


Fig. 2. Architecture of Oath

Mechanisms & Implementation

Oath实现需解决的问题

- Rich OS默认也会设置iRAM为不安全来争夺使用，需隔离；

  -> 工程上寻找方法

- 单独使用iRAM存储敏感信息，或会对rich OS 产生功能和性能影响；

  -> 动态分配

- iRAM一般都很小，难处理并行TA和大图像；

  -> 只存储敏感数据；双层显示图像

Oath机制架构

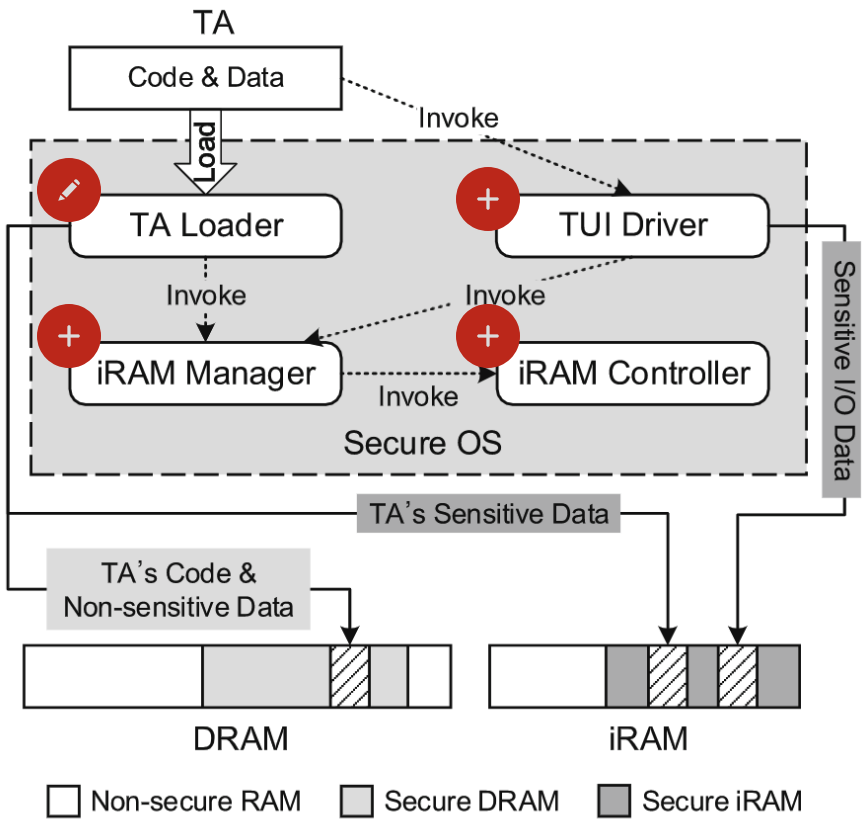


Fig. 2. Architecture of Oath

- 【修改】TA loader 分配堆栈内存并加载TA；

  -> 存储到iRAM，内分分割

- 【新增】iRAM Manager 来管理安全iRAM；

  ->内分动态分配，优化iRAM利用率

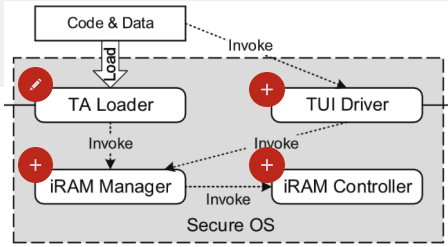
- 【新增】iRAM Controller 控制器；

  -> boot和执行期间及时调整

- 【新增】TUI Driver 支持trust user API用于可信显示和输入

  -> 只存敏感，双显示，加固

TA Loader修改：Memory Splitting



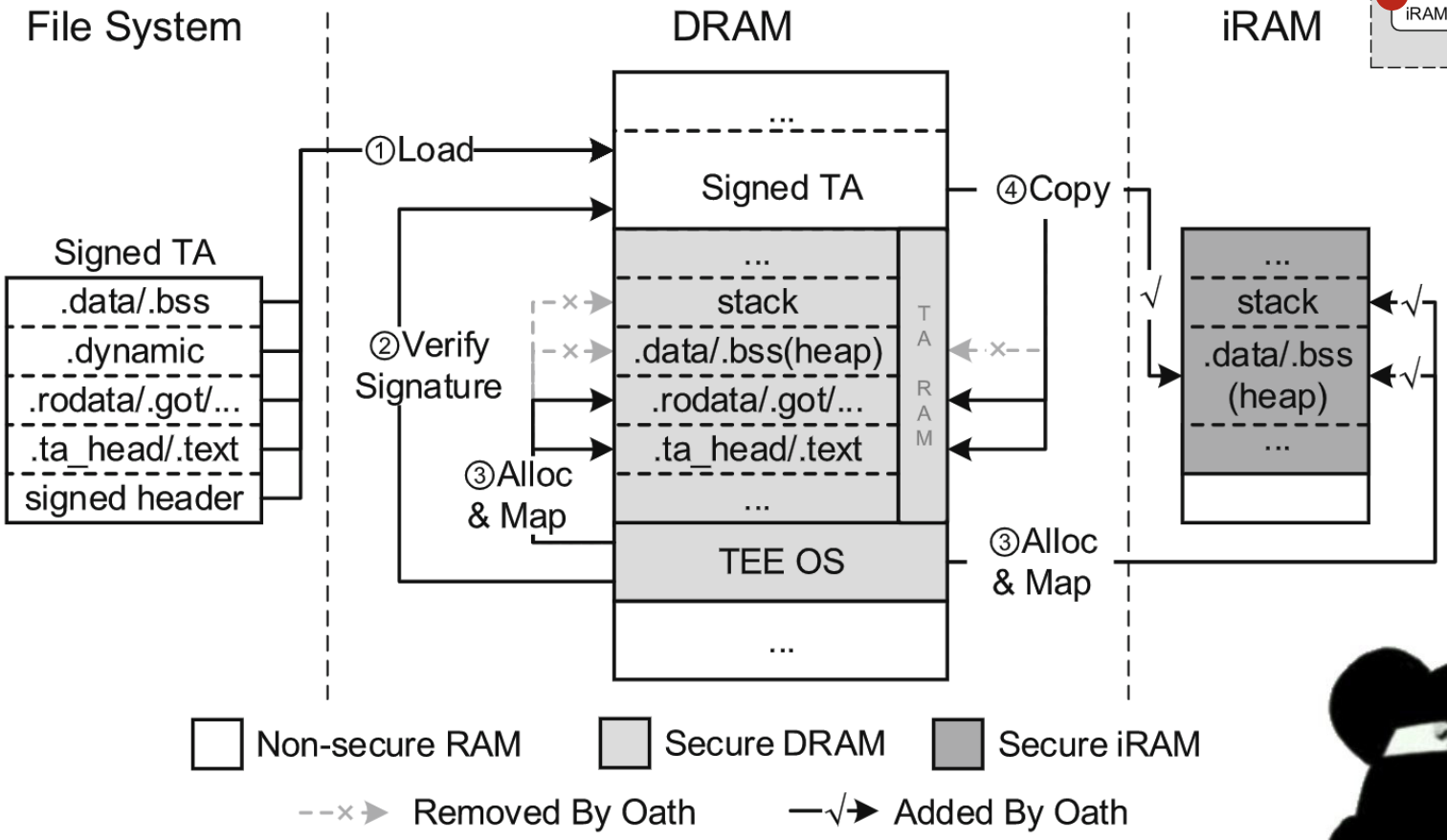
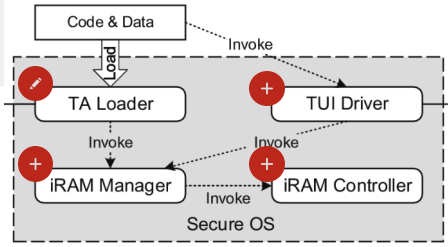


Fig. 4. Procedure to load a TA

iRAM Manager实现



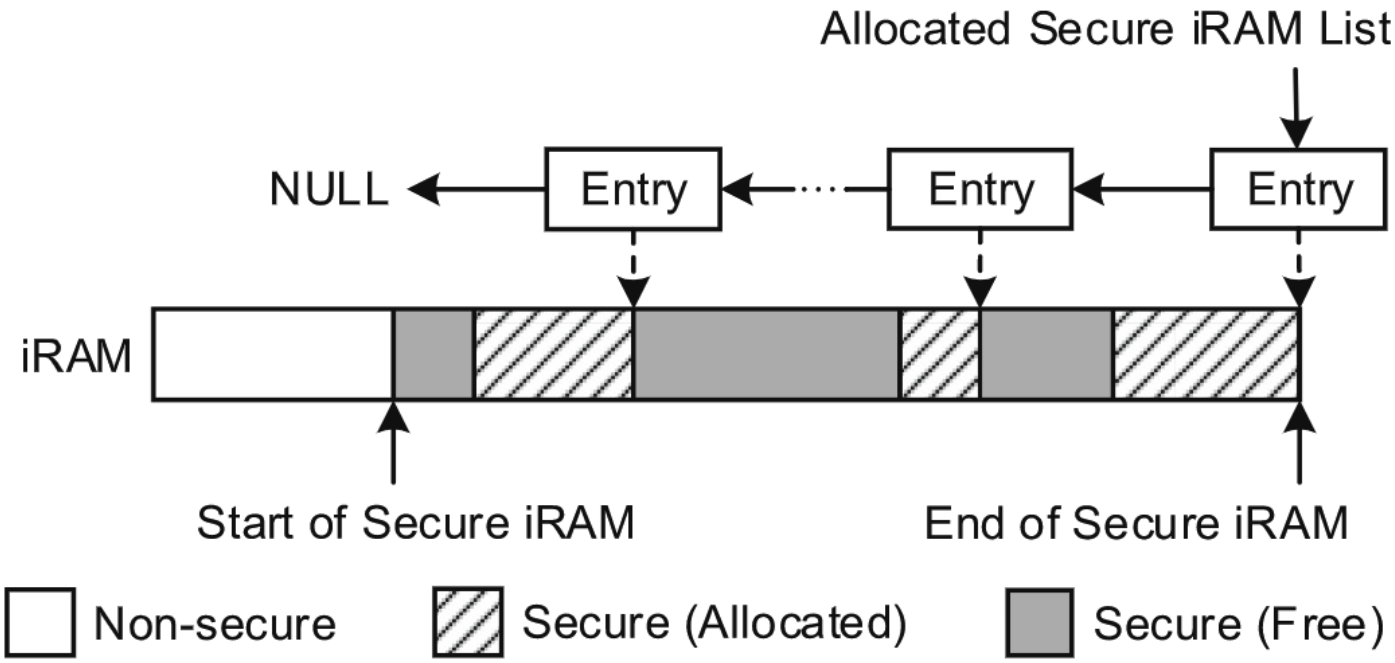
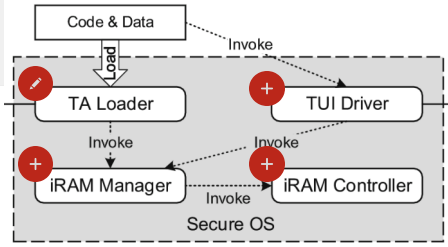


Fig. 3. Management of the secure iRAM

- 一个链表维护分配内存；
- 用 first-fit （首次适应）算法遍历空闲空间，减少查找时间；
- Boot初始化时做了一些优化，rich OS与secure OS交互时动态调整

iRAM Controller实现：平台原生配置 + 自定义iRAM动态分配

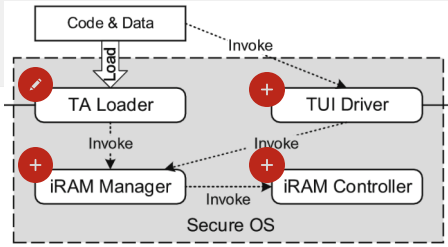


- 配置一下：
  - i.MX6Quad平台原生支持TrustZone的TZMA和TZPC机制，具体实现分别集成于I/O Multiplexer Control (IOMUXC)和Central Security Unit (CSU)，控制General Purpose Register 10 (GPR10) 开启；
  - 通过GPR10中两个字段(OCRAM _TZ _ADDR and OCRAM_TZ_EN)，控制 iRAM 大小；为防止越界，通过另外两个字段加锁限制(LOCK _OCRAM _TZ _ADDR and LOCK _OCRAM _TZ _EN)；
  - CSU控制外设访问权限，通过Config Security Level(CSL6 and CSL27)限制IOMUXC和HDMI；
  - 手册不明确，还做了一个二分枚举测试以得到具体对应的API
- DIY一个：
  - 仅配置不能动态调整 iRAM 分配
  - 不再通过加锁限定 iRAM 起始地址
  - 移植normal OS中的IOMUXC操作以和secure OS中API同步来增强安全性

Listing 1.1. Secure Read And Write Methods

```
//return value of the register "regAddr"
secure_readl(regAddr);
//write register "regAddr" with value "val"
secure_writel(val, regAddr);
```

TUI Driver



- 大尺寸的图像和小容量的 iRAM 内存之间的矛盾
- Display时需要安全的 iRAM 从 DMA 控制器的访问

- Trust Input时，多外设协同需要开关中断时，及时同步修改安全位；
- Trust Display时，分成前景（敏感存储）、背景（非安全存储）双显示，

可信显示时**直接挂起rich OS**，或者**临时**放行，或者放一个**共享主外设**

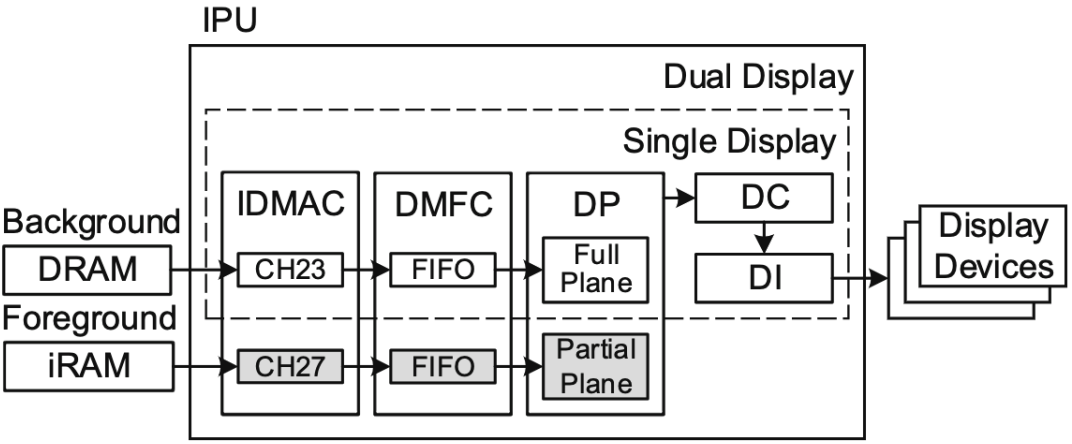


Fig. 5. Image displaying through IPU

Listing 1.2. Configuring IPU for Trusted Display

```
//width, height: foreground width and height
//xp, yp: x and y position of foreground
//trans: transparency of foreground
ConfigIPUForTrustedDisplay(width,height,xp,yp,trans) {
    //Preparation
    OriginalIPUContextSave();
    LookUpTableCreate();
    //Configuration
    Fg_IDMAC_Config(FrameBufAddr, width, height, LUTMode)
        ;
    DMFC_Config(FgCH27);
    DP_Config(xp, yp, trans);
    WaitForStop();
    //Restoration
    OriginalIPUContextRestore();
}
```

Novelty

创新要素

- 免疫思维
  - 站在巨人的肩膀上（ARM TrustZone, OP-TEE, NXP, CaSE）
- 旧瓶装新酒，触类旁通
  - Sensitive TA data is now read from or written to iRAM rather than DRAM
  - 比基于cache的方案高到不知道哪里去了
- 讲求实惠，精细划清安全界限，不该花的一毫不拔
  - The dual display technology adopted by Oath to achieve trusted display
  - iRAM memory splitting
- 隐身衣有时比金钟罩效果好

Results & Evaluation

评估环境及方法

- 实验测试环境
  - 硬件： FreeScale i.MX6Quad sabre development board
    - 1.2Ghz quad-core ARM Cortex-A9 processor
    - 1GB DDR3 SDRAM & 256KB onboard iRAM
  - 软件：
    - Secure World： OP-TEE OS 2.2.0
    - Normal World： Freescale Android 6.0.1 system with a 4.1.15 Linux kernel
  - 统计方法：每个测试样例1000次迭代取平均值
- 功能性影响
  - 使用原生开源测试suite：xtest(optee_test)
- 性能影响
  - 测试benchmark

对原始OP-TEE的功能性影响评估

- 开源套件里功能相关test case有22647个
- 因其中28个需要调用Oath所删掉的TA或硬件模块，直接忽略，最终用了其中22619个进行评估

Table 1. Function Impacts on OP-TEE

Category	Original	Oath
Main functions (8815)	√	√
TEE internal API (12894)	√	√
TA storage (268)	√	√
Shared memory (125)	√	√
Key derivation (225)	√	√
Sanity test (292)	√	√

- **基本功能无损**
- **2个测并行的体积太大load不进iRAM 过不了测试，直接改小**

对原始OP-TEE的功能性影响评估（Cont.）

Table 2. Memory Demands of TAs

TA name	Total(T) (KB)	iRAM(I) (KB)	Ratio (1-I/T)
aes_perf	111.04	47.04	57.64%
concurrent	110.95	46.95	57.68%
* concurrent_large	2126.95	2062.95	3.01%
create_fail_test	89.95	45.95	48.92%
crypt	135.52	47.52	64.94%
* os_test	1122.69	926.69	17.46%
rpc_test	96.96	48.96	49.50%
sha_perf	110.95	46.95	57.68%
sims	128.98	80.98	37.21%
storage	114.95	46.95	59.16%
storage2	114.95	46.95	59.16%
storage_benchmark	98.97	46.97	52.54%

* iRAM memory demand exceeds 256 KB.

- the average demand of iRAM memory for each TA is 50.52 KB. **可并行4个TA**
- **Oath中引入的内存拆分机制的有效性，该机制平均减少了56.49%的 iRAM 内存需求**

对Rich OS的功能性影响评估

Table 3. Time breakdown of trusted display

Action	Display (µs)	Restore (µs)
World switching	14408.60	34.46
* Core suspending/resuming	3.47	42.24
* DMA suspending/resuming	3.24	3.02
* Secure iRAM adjusting	0.08	0.20
* IPU context saving/restoring	1.44	4.71
* IPU configuring	2876.21	–
Total	17293.04	84.63

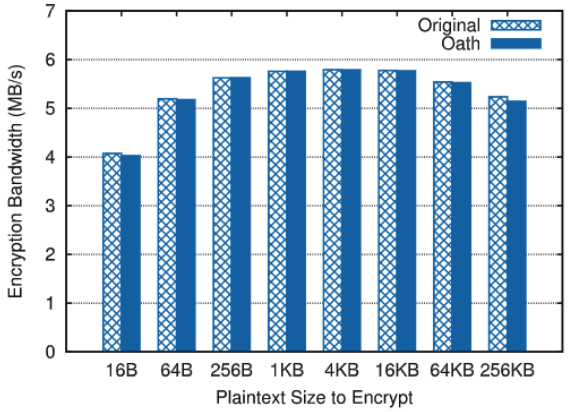
* Rich OS is suspended.

Case中可信显示时总耗时17293.04µs， 为此rich OS挂起全程耗时2884.44µs
还原时上下文切换总耗时84.63µs，期间rich OS挂起耗时50.17 µs
每个可信显示操作所需的rich OS挂起总时间 = 2934.61µs（2884.44+50.17）+ 等待时间

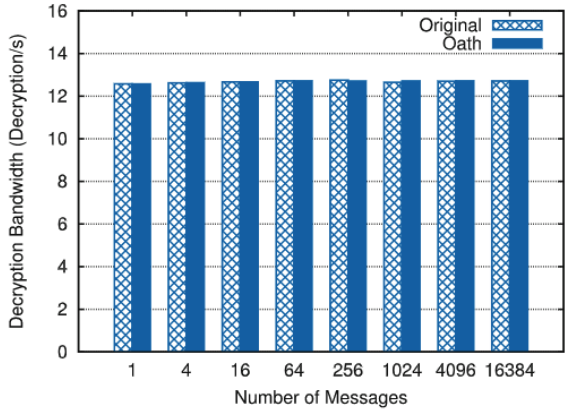
- 不能多并行一些大程序
- 视频解码某些时候需要被迫延迟
- 不太用户体验（有一点点，不能多了）

对原始OP-TEE的性能影响评估

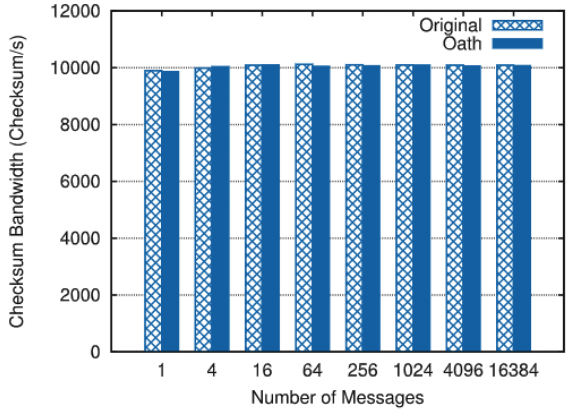
- 迭代取平均
- 基本无损
- TA加载过程损耗：通过PMU进行对比测试10个常见TA，load TA时间分别是111.27ms 和 109.77ms，Oath增加开销1.37%
- TA执行时间损耗：测试3个常见加密TA，差距最大的是AES中256KB加密，增加开销1.8%



(a) AES Speed Comparison



(b) RSA Speed Comparison



(c) SHA256 Speed Comparison

对Rich OS的性能影响评估

- 移植操作引入的两个额外的上下文切换开销
- 在Android OS上设置参数，统计日志中boot时间，增加开销0.16%
- 换了一个综合的benchmark suite：AnTuTu 2.9.4，基本无损，开销小于0.5%

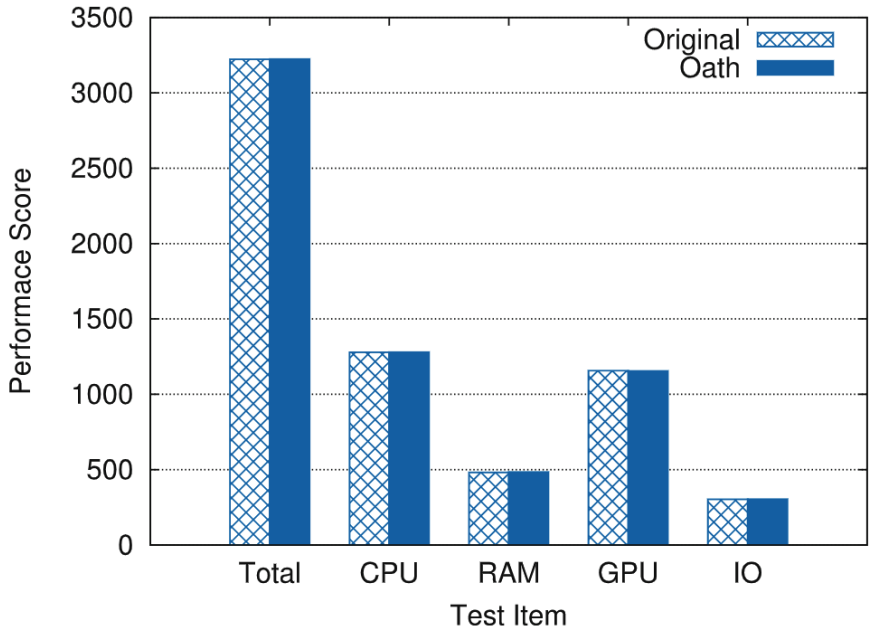


Fig. 7. Comparison of Rich OS Performance

安全性评估

- 敏感数据仅存储在secure iRAM，并在secure CPU Core中执行，故天生免疫软硬件攻击；
- **两个例外**：
  - 可信显示时**上下文切换**，有短暂不安全的framebuffer，较小间隔轮询访问，寄存器中数据可能会泄露，但数据仍然安全（因为中断时直接先刷新清零再关掉）
  - iRAM 从DMA及内存之间**通信时**会临时放行，但由secure CPU处理数据，仍在内部，问题不大

Strengths & Weaknesses

Strength

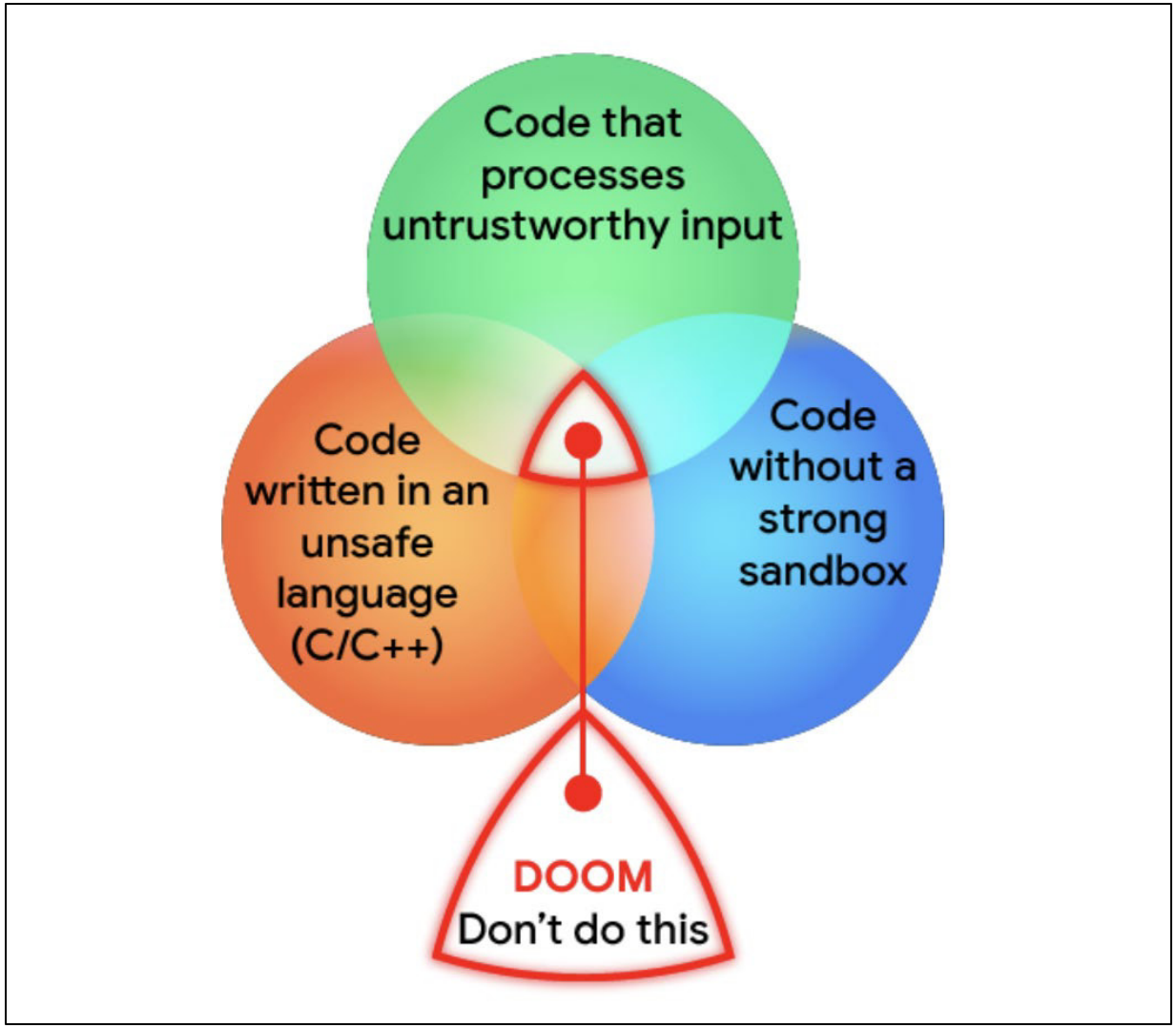
- 是一个完整项目的技术分享
- 论文结构很清晰，多次点题
- Simple, novel mechanism to solve an important problem
- Effective and low hardware overhead
- 借鉴了其他基于iRAM的防御机制(InvisiSpec [MICRO 2018])
- 没有太多改动硬件，对底层工程师友好
- 有一定HW/SW Co-Design
- 做工程的直接思想：内存拆分机制等，不通则变

Weakness

- Also too Theoretically
- 不是原创的 idea，在前人工作上的二创改进
- 没给代码，而且并未充分证明其机制本身的安全性
- 假定的 bug 处，没有通过实验证明其可抵抗
- iRAM内存小，限TA并发程度和安全显示图像的大小
- 对其他硬件平台的可移植性不强
- 可信显示启动时对 rich OS 的瞬时挂起上下文切换带来一定安全风险/性能损耗

Discussion

还有哪些空白可探索


Code that processes untrustworthy input
Code written in an unsafe language (C/C++)
Code without a strong sandbox
DOOM
Don't do this

套娃解决方案是否可持续

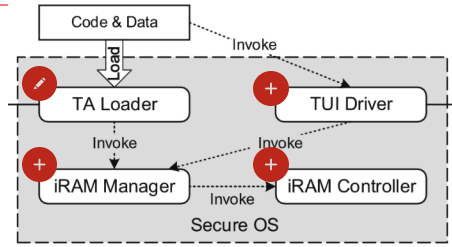
本文提出后续工作可以引入另一级虚拟内存（即利用DRAM作为加密iRAM页面的备份存储）来扩展Oath以支持更大尺寸的TA

和CaSE类似了

感觉是套娃

Problem：

- **TEE难防御硬件侧信道且具体实现有漏洞**
  - ▶基于cache的防御方法（CaSE）性能开销大，不够轻量
  - ▶现有片上存储解决方案不能有效加固敏感I/O信息保护



Key Idea：

- **用iRAM存储隔离关键安全信息取代L2 cache方案，配合软件层优化和工程直觉方案实现安全免疫**

Mechanisms ：

- **工程实现Oath: 增加iRAM的使用管理机制，修改原先load方法**

Results ：

- **靠800行代码，性能开销极低，对功能基本无影响**
- **成功免疫软硬件攻击，成功避免敏感数据特别是I/O数据泄露**

感谢批评指正

THANKS