浙江大学网络系统安全与隐私实验室
NETWORK SYSTEM SECURITY & PRIVACY LAB

hexhive

SyzTrust: State-aware Fuzzing on Trusted OS Designed for IoT Devices

Qinying Wang, Boyu Chang, Shouling Ji, Yuan Tian, Xuhong Zhang, Binbin Zhao, Gaoning Pan, Chenyang Lyu, Mathias Payer, Wenhai Wang, Raheem Beyah

浙江大學 ZHEJIANG UNIVERSITY

EPFL

UCLA

Georgia Tech

Motivation

Trust Execution Environments (TEEs) are **essential** to IoT security.

Smart Lock

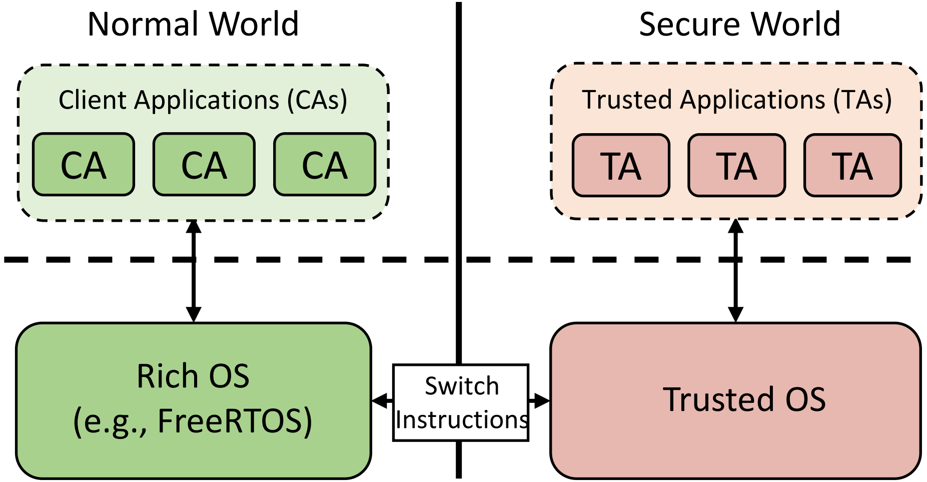
Drone

FIDO Security Key

Smart Locker

Motivation

A TEE is an **isolated** and **trusted** environment shielded against local attacks.

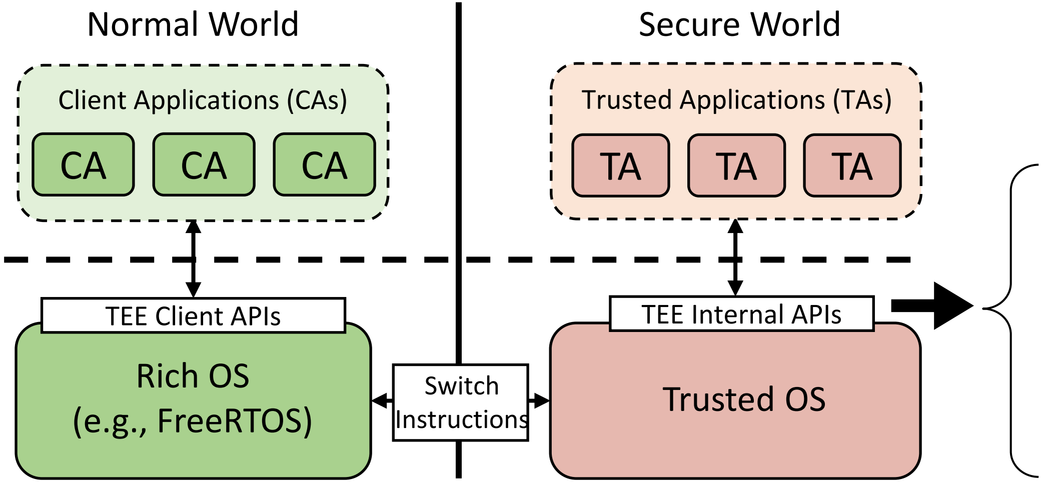


TrustZone-M based TEE

Sensitive data is stored, processed, and protected.

Motivation

Trusted OS is the **primary** component to enable the TEE to use security techniques.

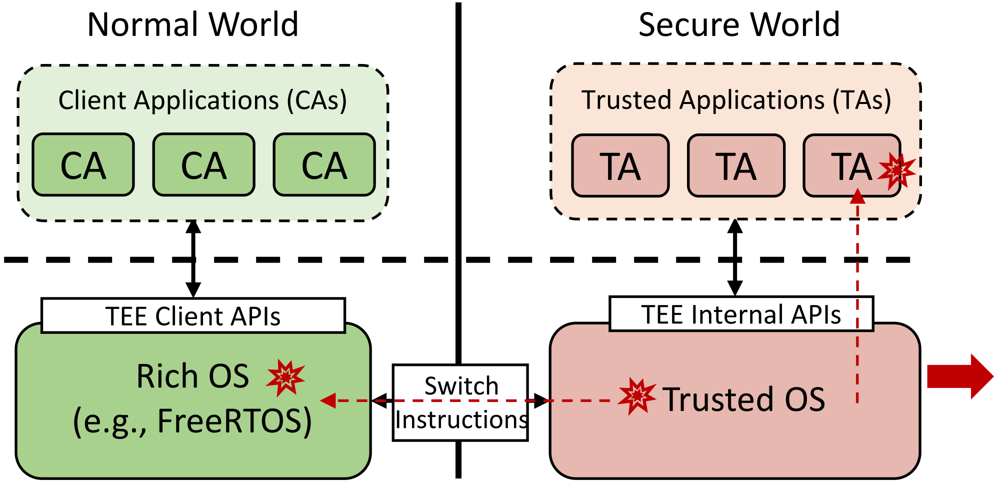


- Trusted storage APIs
- Cryptographic operations APIs
- Time APIs
- Peripheral and event APIs

TrustZone-M based TEE

Motivation

The flaws in Trusted OS lead to **severe** results, which can be further utilized to **exploit** other TAs and even the whole system.

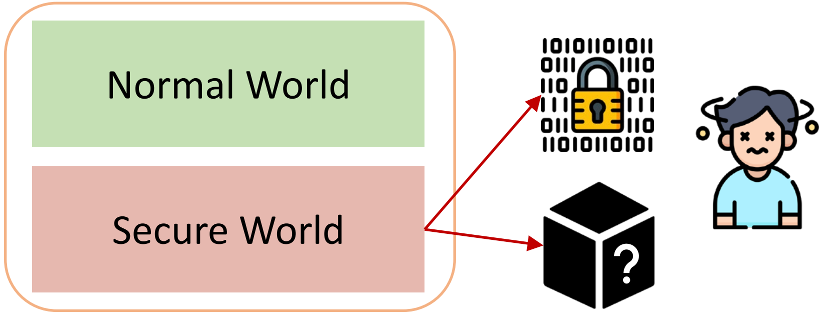


- Gaining control
- Extracting confidential data
- Causing system-wide crashes

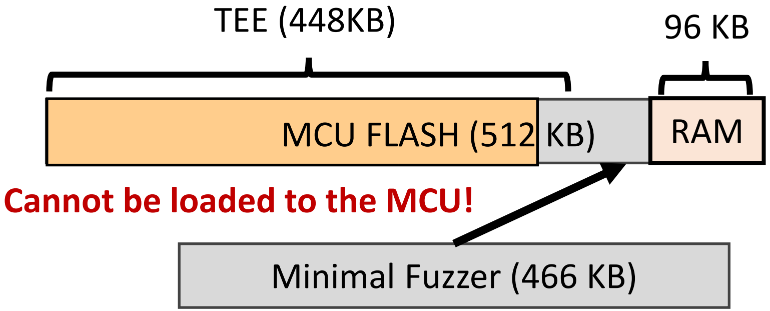
TrustZone-M based TEE

Challenges of Fuzzing Trusted OSes

Challenge 1: Inability of instrumentation and constraint resource



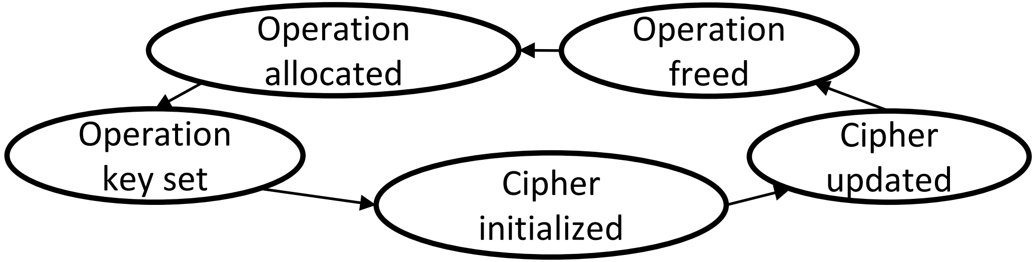
Close source and encrypted



Resource limited

Challenges of Fuzzing Trusted OSes

Challenge 2: Stateful workflow and complex structure



Stateful workflow

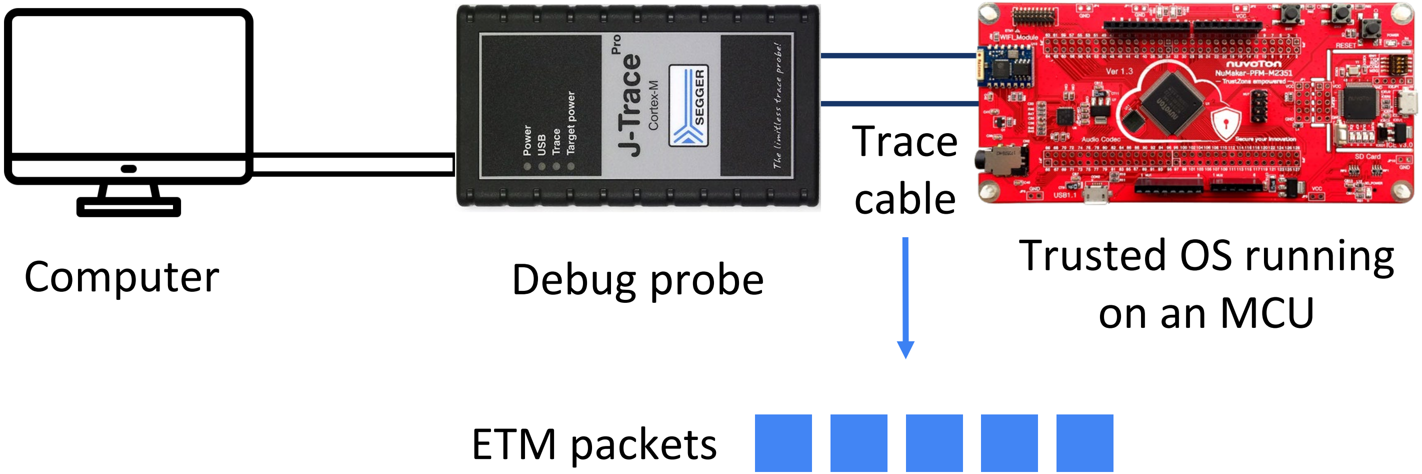
```
struct TEE_OperationHandle{
        uint32_t  algorithm,
        uint32_t operationState,
          TEE_ObjectHandle key…
}
```

Complex structure to control the execution contexts

Key Observations towards a Solution

Challenge 1

ARM Coresight Embedded Trace Macrocell (ETM) provides real-time **instruction tracing**, where we can **obtain code coverage**.



Computer

Debug probe

Trace cable

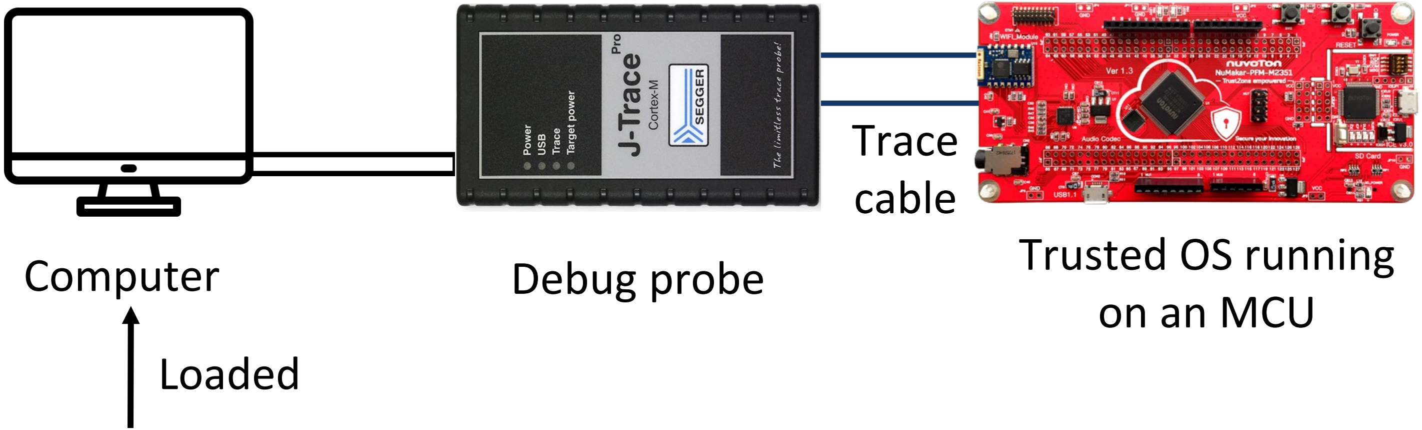
Trusted OS running on an MCU

ETM packets

Key Observations towards a Solution

Challenge 1

We can **decouple** execution to offload heavy-weight tasks to our computer.



Trace cable

Computer

Debug probe

Trusted OS running on an MCU

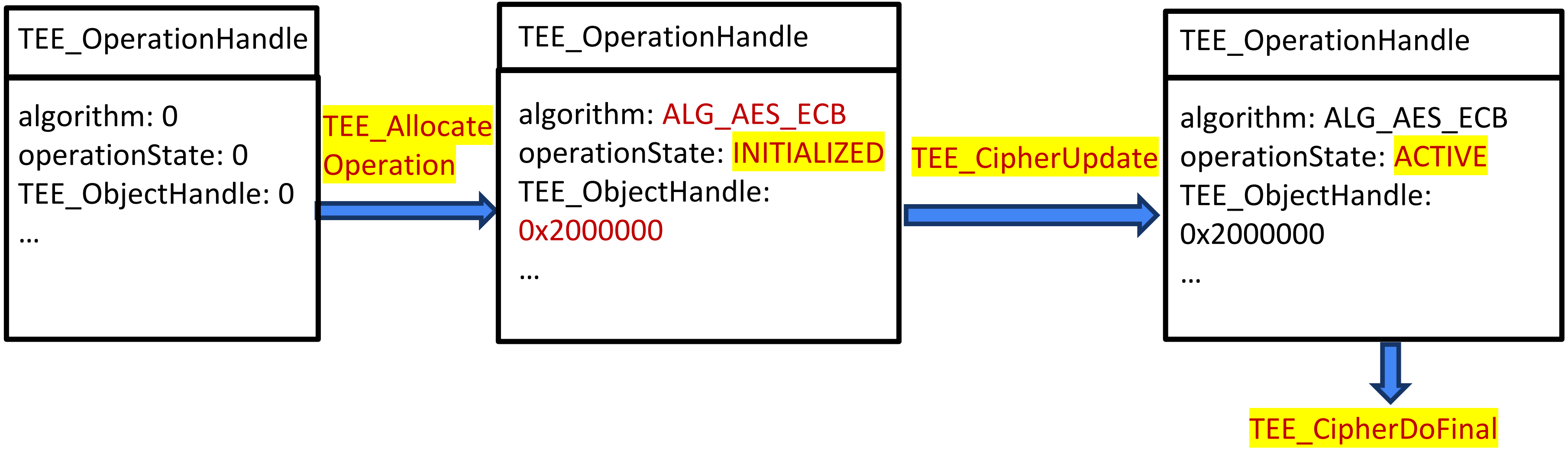
Loaded

Heavy-weight tasks

Key Observations towards a Solution

Challenge 2

Several variables in **handle structures** determine and store the Trusted OS’ **state**.



TEE_OperationHandle

algorithm: 0
operationState: 0
TEE_ObjectHandle: 0
…

TEE_Allocate Operation

TEE_OperationHandle

algorithm: ALG_AES_ECB
operationState: INITIALIZED
TEE_ObjectHandle: 0x2000000
…

TEE_CipherUpdate

TEE_OperationHandle

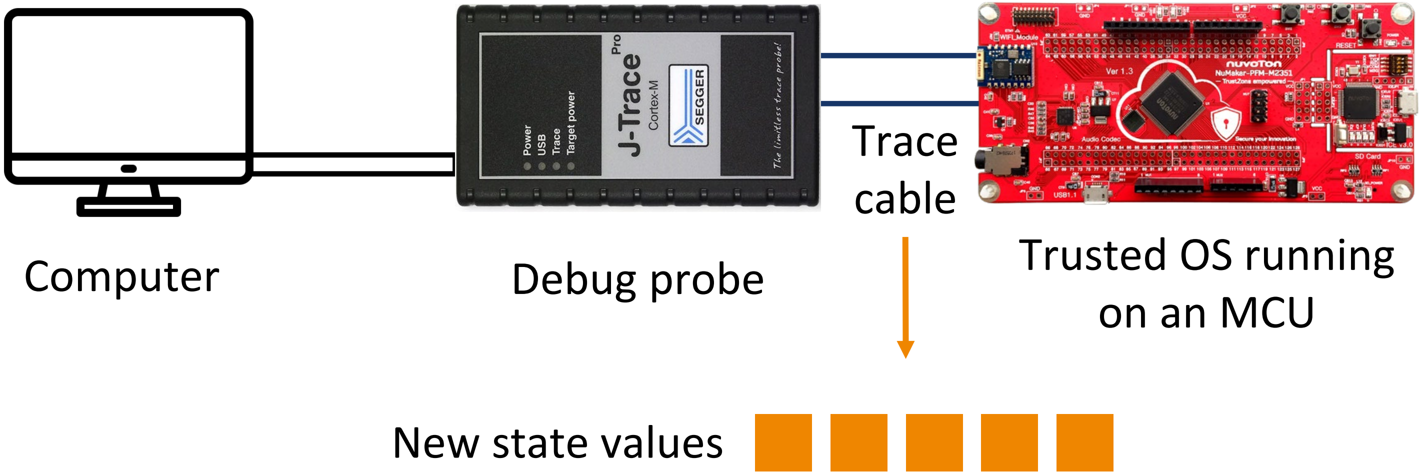
algorithm: ALG_AES_ECB
operationState: ACTIVE
TEE_ObjectHandle: 0x2000000
…

TEE_CipherDoFinal

Key Observations towards a Solution

Challenge 2

We can stream the state variable values via a debug probe and **calculate state coverage**.



Computer

Debug probe

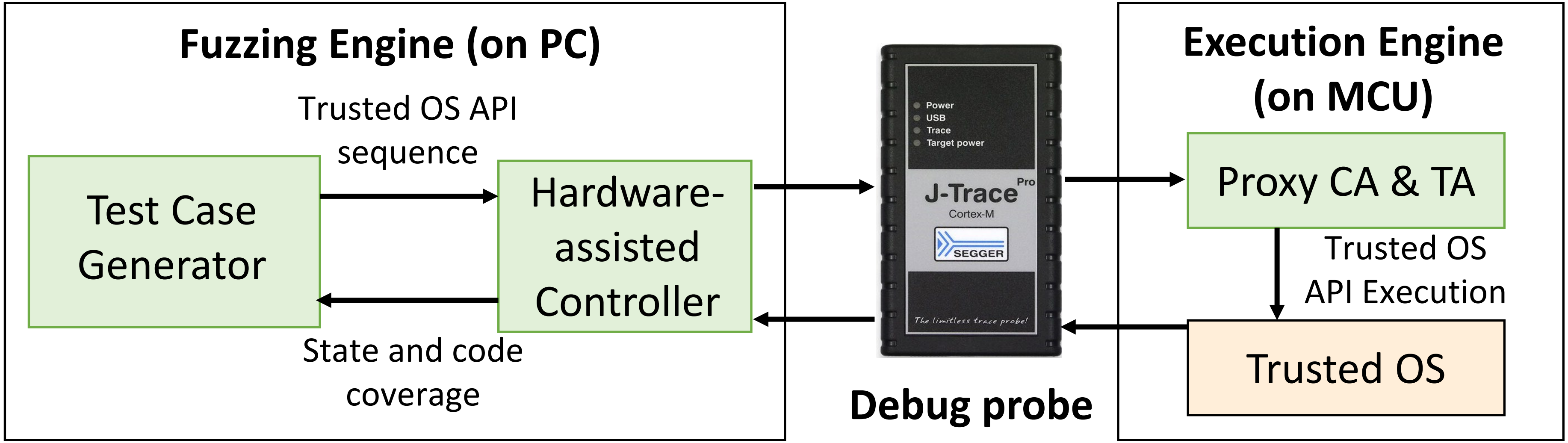
Trace cable

Trusted OS running on an MCU

New state values

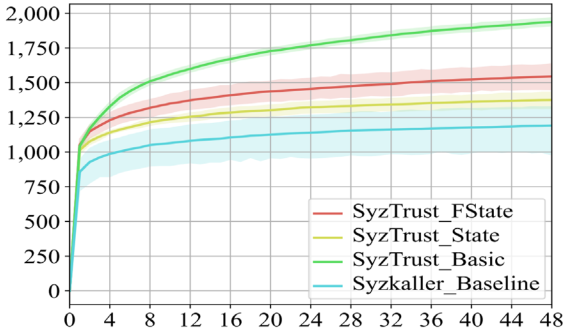
SyzTrust End-to-End

SyzTrust includes two modules, the fuzzing engine and execution engine.

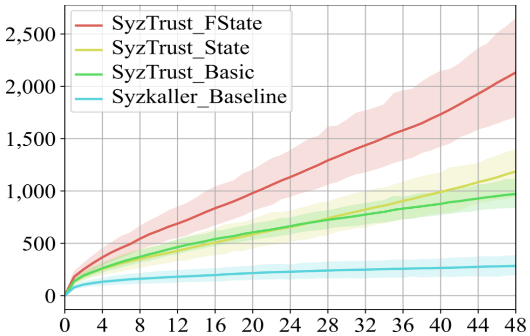


Evaluation - Baseline Comparison

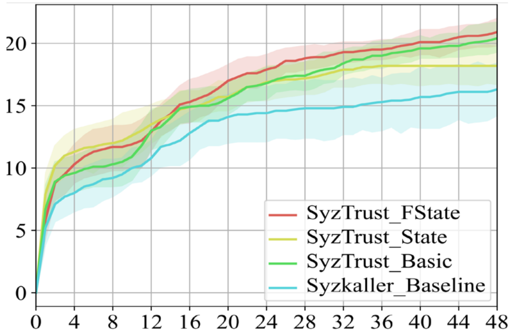
SyzTrust **outperforms** *Syzkaller in terms of branch coverage, state coverage and unique vulnerabilities.



Branch coverage



State coverage

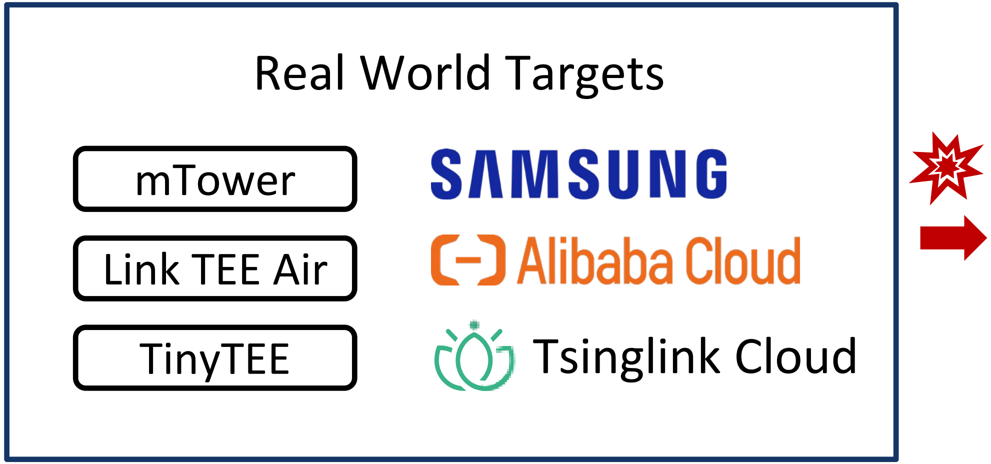


Unique vulnerabilities

*https://github.com/google/syzkaller

Evaluation - Vulnerabilities

SyzTrust discovered **70 unknown** vulnerabilities.



Real World Targets

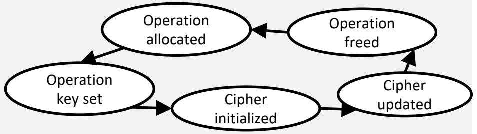
- Buffer overflow
- Missing release of memory after effective lifetime
- Null pointer dereference
- Untrusted pointer dereference

We got **19 CVEs** to date, each rated as **high-impact** vulnerabilities.

SyzTrust: State-aware Fuzzing on Trusted OS Designed for IoT Devices

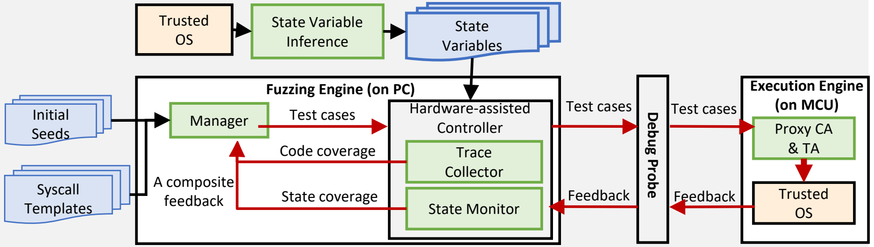
Key Challenges

- Inability of instrumentation
- Limited resources
- Stateful workflow



Summary of SyzTrust

First fuzzing framework for IoT Trusted OSes



Real World Impact

mTower	SAMSUNG
Link TEE Air	Alibaba Cloud
TinyTEE	Tsinglink Cloud

- 70 previously unknow bugs
- 34 bugs have been patched
- 19 CVEs

Email: wangqinying@zju.edu.cn

浙江大学网络系统安全与隐私实验室
NETWORK SYSTEM SECURITY & PRIVACY LAB

hexhive