TEEzz: Fuzzing Trusted Applications on COTS Android Devices

Marcel Busch, Aravind Machiry, Chad Spensky, Giovanni Vigna, Christopher Kruegel, Mathias Payer

{marcel.busch, mathias.payer}@epfl.ch
amachiry@purdue.edu, chad@allthenticate.net
{chris, vigna}@ucsb.edu
Modern TZ-based TEEs on Android Mobile Devices
ARM TrustZone Privilege Levels

Normal World
(Rich Execution Environment)

MEMORY CORRUPTIONS

Secure World
(Trusted Execution Environment)

MEMORY CORRUPTIONS EVERYWHERE

Trust Applications

Trusted OS
Challenges of Fuzzing Trusted Applications

1. Limited introspection

2. Complex input

3. Statefulness
Observations and Intuitions

- Clients located in “normal world”
- Semantics decrease towards lower levels of abstraction
- Control over input increases towards lower levels of abstraction
TEEzz – End-to-End

Client Application Identification
- App
  - AOSP System Service
  - Vendor System Service
  - CAlib₁
  - CAlib₂
  - CAlib₃
  - libteec
  - CAlib consumer

CA Interface Processing
- iface
  - AST
  - DBII Recorder

Seed Recording
- CAlib consumer

Fuzzing Engine
- Mutation Engine
  - DBII Recorder
  - 011101011111...
  - 11101010011...
  - Type- and State-Aware Seeds
Automatically Generating Type-aware Seeds
enum E { x, y, z };  
struct S {  
    int a;  
    enum E e;  
    char b[64];  
}  
int func(struct S* s);  

enum_decl

int func(struct S* s) {  
    onenter  
    x = 0  
    // complex logic  
    onleave  
}

def mutate_enum(enum_decl):
    return random.choice(enum_decl)

enum E
enum E e
char b[64]
```c
int f(param_set_t* P, keyblob_t** K)

param_set_t {param_t[] p, size_t len}

keyblob_t {uint8_t[] k, size_t len}

param_t {enum E, short Magic, blob B}

p[] = 0x1234

E  Magic  B
1  0xCAFE  0xAAAA
2  0xCAFE  0xBBBB
3  0xCAFE  0xABAB

p[0] = 0x1234, len=3

len: 03 00 00 00
p[0]: 01 00 00 00 CA FE AA AA
p[1]: 02 00 00 00 CA FE BB BB
p[2]: 01 00 00 00 CA FE AB AB

len: 00 02 00 00
k: 4d 4b 49 48 b4 6c 22 4a ...

size_t
{enum, short, blob}
{enum, short, blob}
{enum, short, blob}
UNKNOWN (12)

03 00 00 00 01 00 00 00
CA FE AA AA 02 00 00 00
CA FE BB BB 01 00 00 00
CA FE AB AB 00 00 00 00
66 75 7A 7A 74 68 69 73

10 00 00 00 00 02 00 00
00 00 00 00 00 00 00 00
4d 4b 49 48 b4 6c 22 4a ...

6d 74 65 74 65 63 74 6f
72 6f 6d 65 6e 74 69 6d
61 72 79 44 49 52 4c 4f

DBII Recorder

CA Interface

loctl Recorder

Driver Interface

Type Recovery

Seeds / Fuzzing
Templates
```
State-Awareness

int f(param_set_t* P_{in}, keyblob_t** K_{out})

int g(keyblob_t* K_{in})
Evaluation – Finding Bugs

- 4 devices / 2 proprietary TEEs
  - 40 unique crashes
  - CVE-2019-10561
- OPTEE
  - 13 previously unknown bugs
Evaluation – Ground-truth Coverage Experiments
TEEzz: State- and Type-aware Black-box Fuzzing

- 4 devices / 2 proprietary TEEs
  - 40 unique crashes
  - CVE-2019-10561
- OPTEE
  - 13 previously unknown bugs