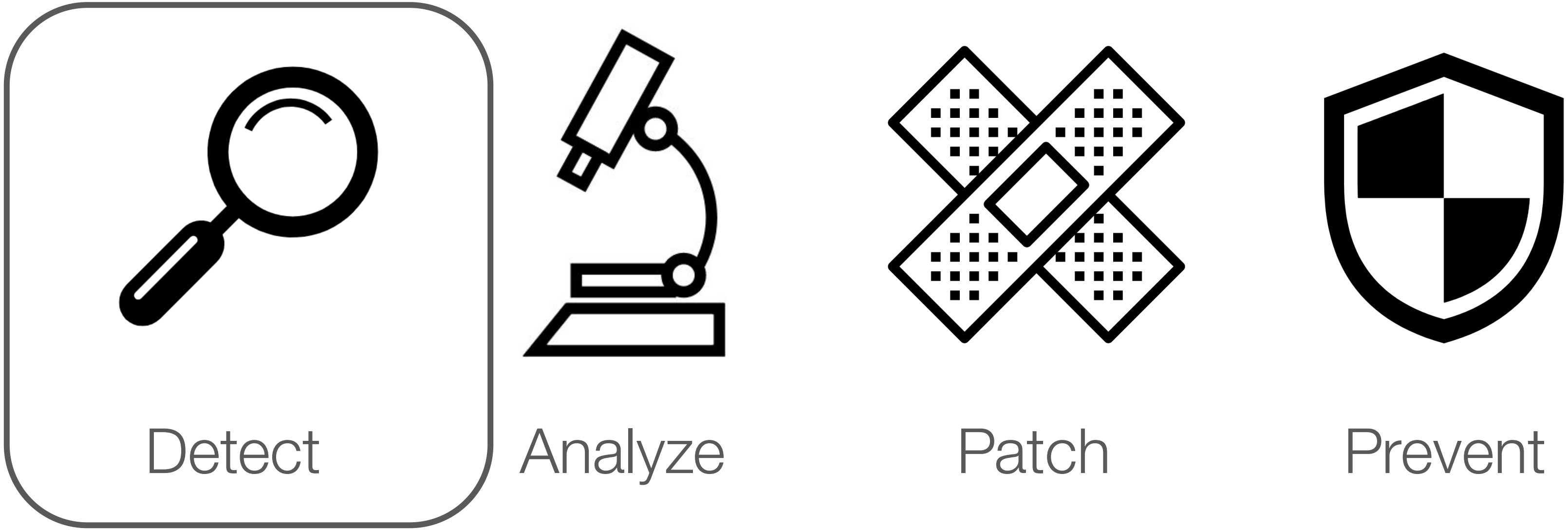
UNIVERSITY OF TWENTE.

E2E Automated Vulnerability Research

Andrea Continella
University of Twente
SCS group

14/06/2023

AVR Lifecycle



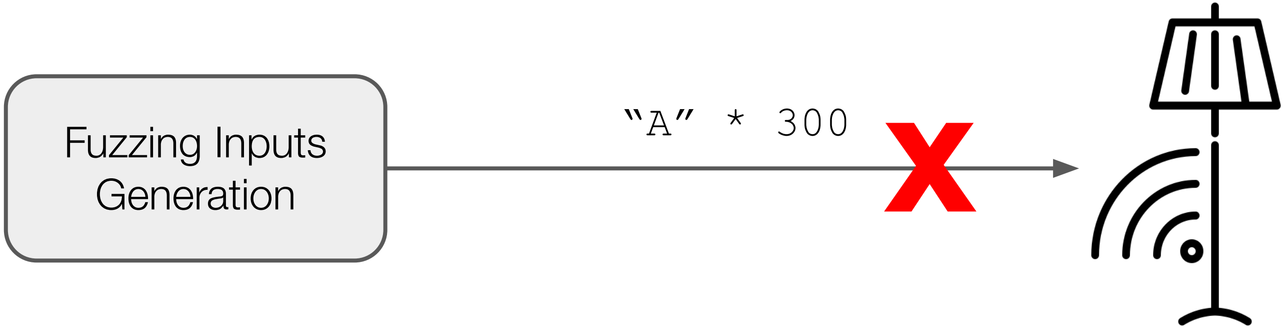
Detect

Analyze

Patch

Prevent

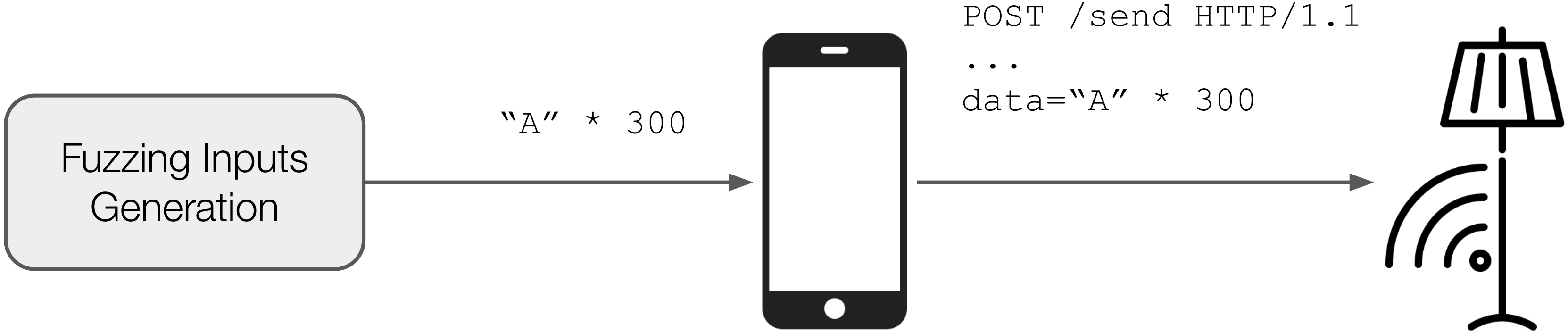
Black-box Fuzzing for IoT Devices



Fuzzing Inputs Generation

`“A” * 300`

Smarter Black-box Fuzzing

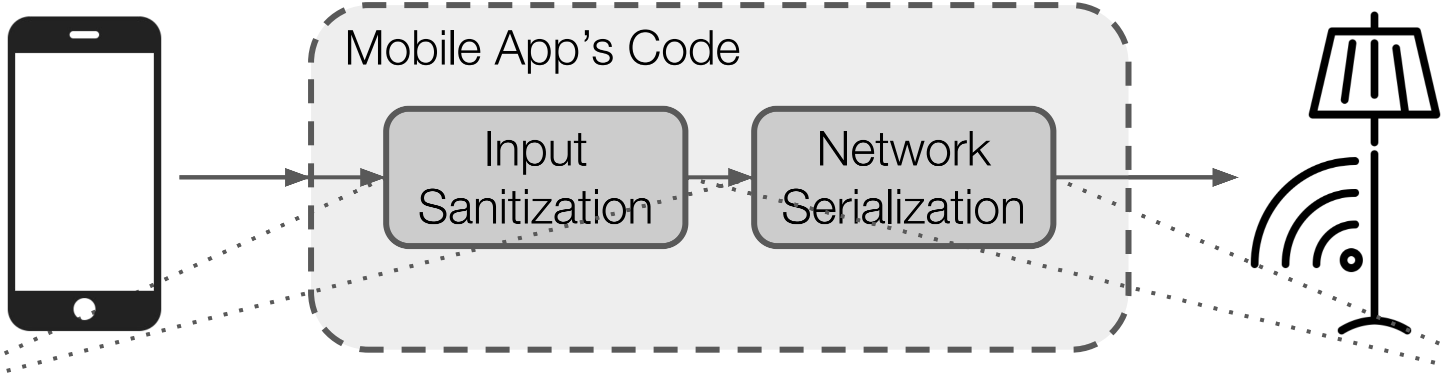


Fuzzing Inputs Generation

```
“A” * 300
```

```
POST /send HTTP/1.1
...
data=“A” * 300
```

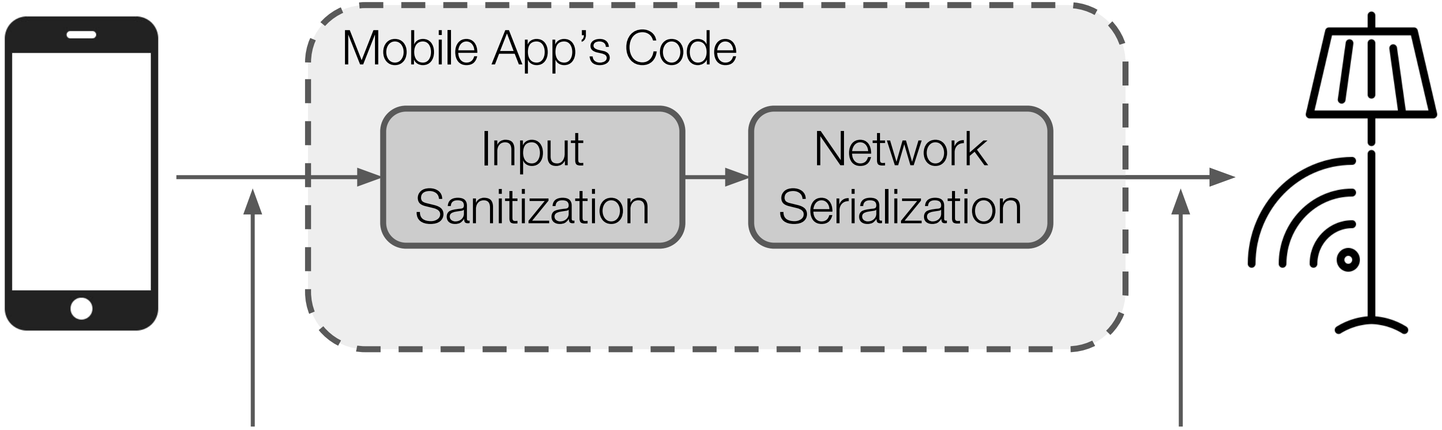
Fuzzing IoT Devices ©



```
...
String json = "{\"op\": \"auth\", \"pass\":" + adminPwd "}";
String encoded = Base64.encode(json);

httpSend(DEVICE_IP; encoded);
```

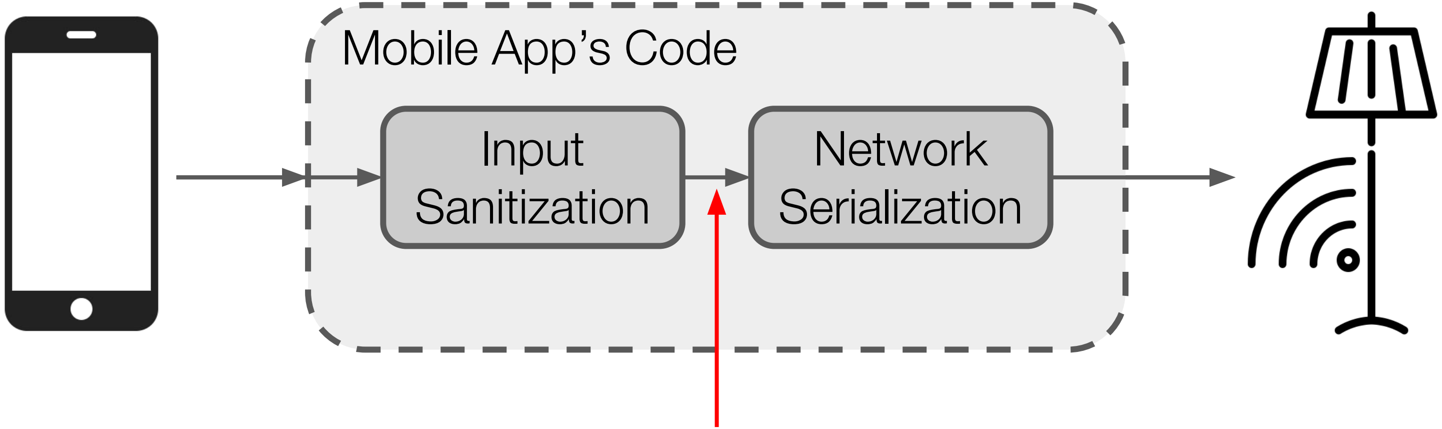
Fuzzing IoT Devices ©



UI-level
Limited by app-side sanitization ❌

Network-level
Invalid inputs ❌

Fuzzing IoT Devices ©

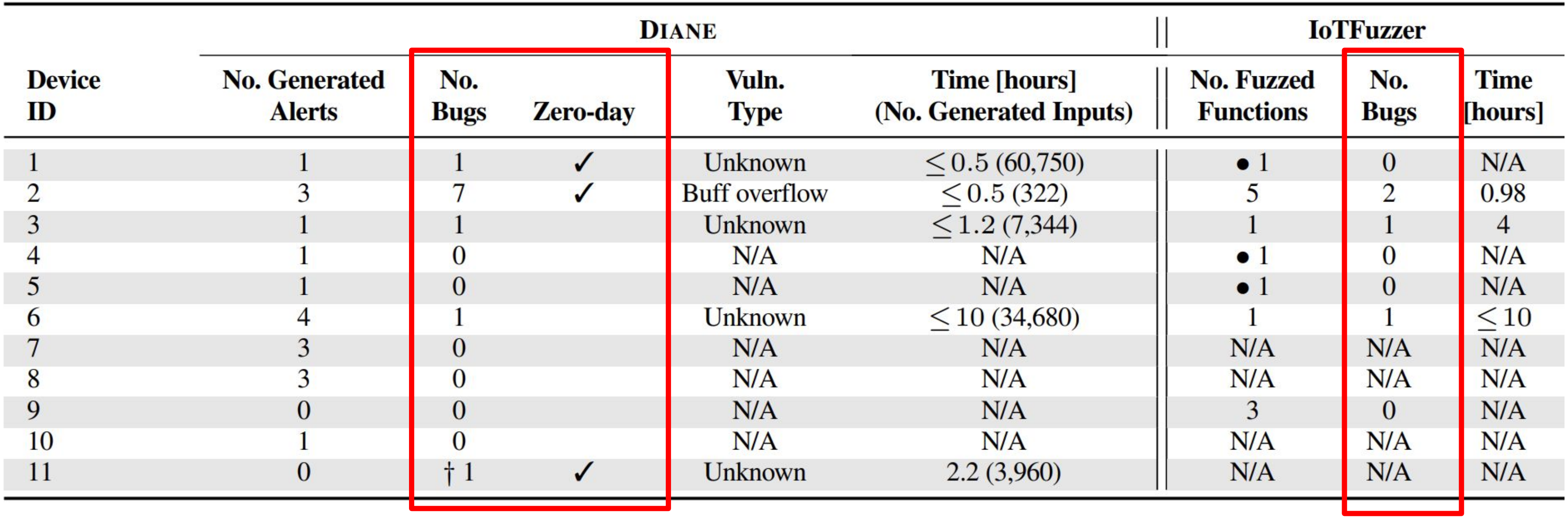


Our Approach

Valid inputs ✓

Not limited by app-side input sanitization ✓

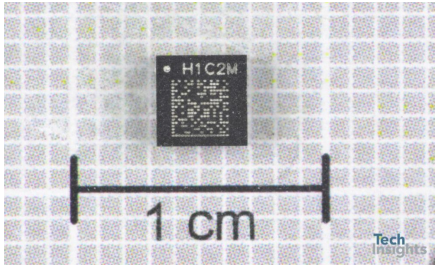
Results & Outcomes



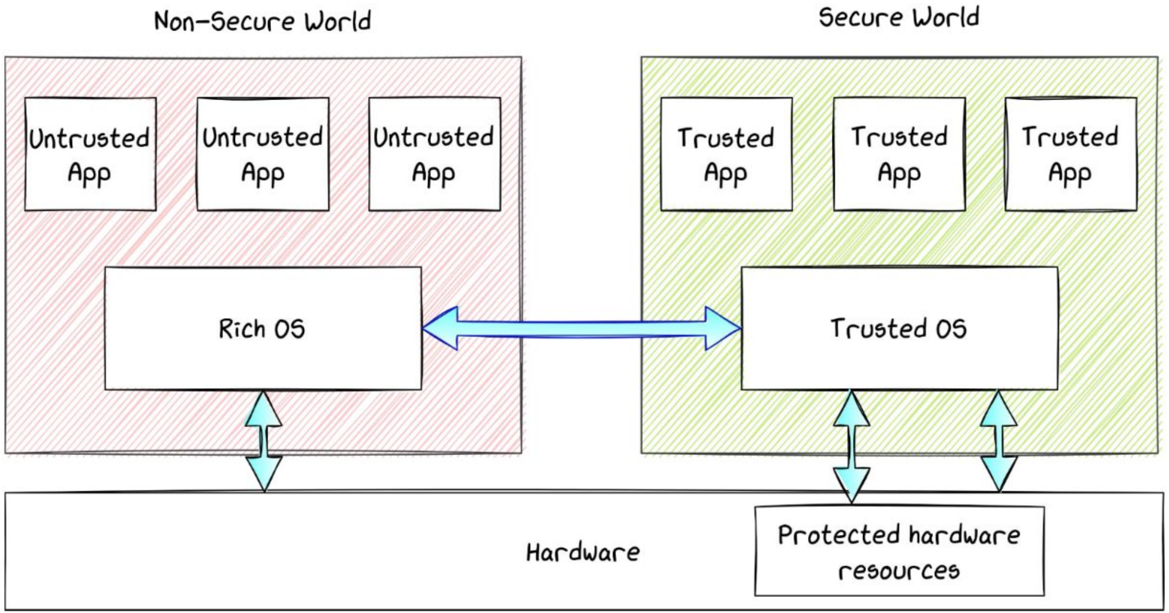
	DIANE					IoTFuzzer		
Device ID	No. Generated Alerts	No. Bugs	Zero-day	Vuln. Type	Time [hours] (No. Generated Inputs)	No. Fuzzed Functions	No. Bugs	Time [hours]
1	1	1	✓	Unknown	$\leq$0.5 (60,750)	● 1	0	N/A
2	3	7	✓	Buff overflow	$\leq$0.5 (322)	5	2	0.98
3	1	1		Unknown	$\leq$1.2 (7,344)	1	1	4
4	1	0		N/A	N/A	● 1	0	N/A
5	1	0		N/A	N/A	● 1	0	N/A
6	4	1		Unknown	$\leq$10 (34,680)	1	1	$\leq$10
7	3	0		N/A	N/A	N/A	N/A	N/A
8	3	0		N/A	N/A	N/A	N/A	N/A
9	0	0		N/A	N/A	3	0	N/A
10	1	0		N/A	N/A	N/A	N/A	N/A
11	0	† 1	✓	Unknown	2.2 (3,960)	N/A	N/A	N/A

DIANE: Identifying Fuzzing Triggers in Apps to Generate Under-constrained Inputs for IoT Devices
In Procs. of the IEEE Symposium on Security & Privacy (S&P), 2021

Google Titan M Chip



External Coprocessor: Trusted Execution Environment (TEE)



Results & Outcomes

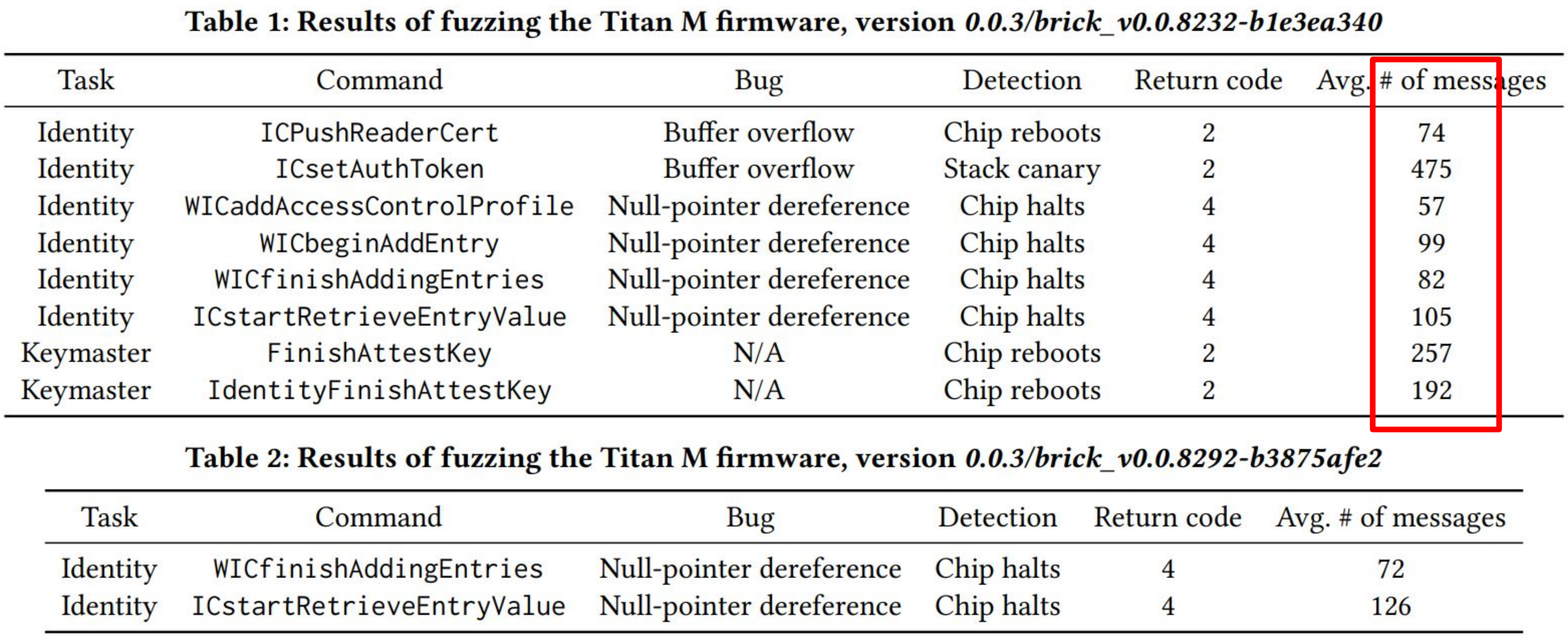


Table 1: Results of fuzzing the Titan M firmware, version *0.0.3/brick_v0.0.8232-b1e3ea340*

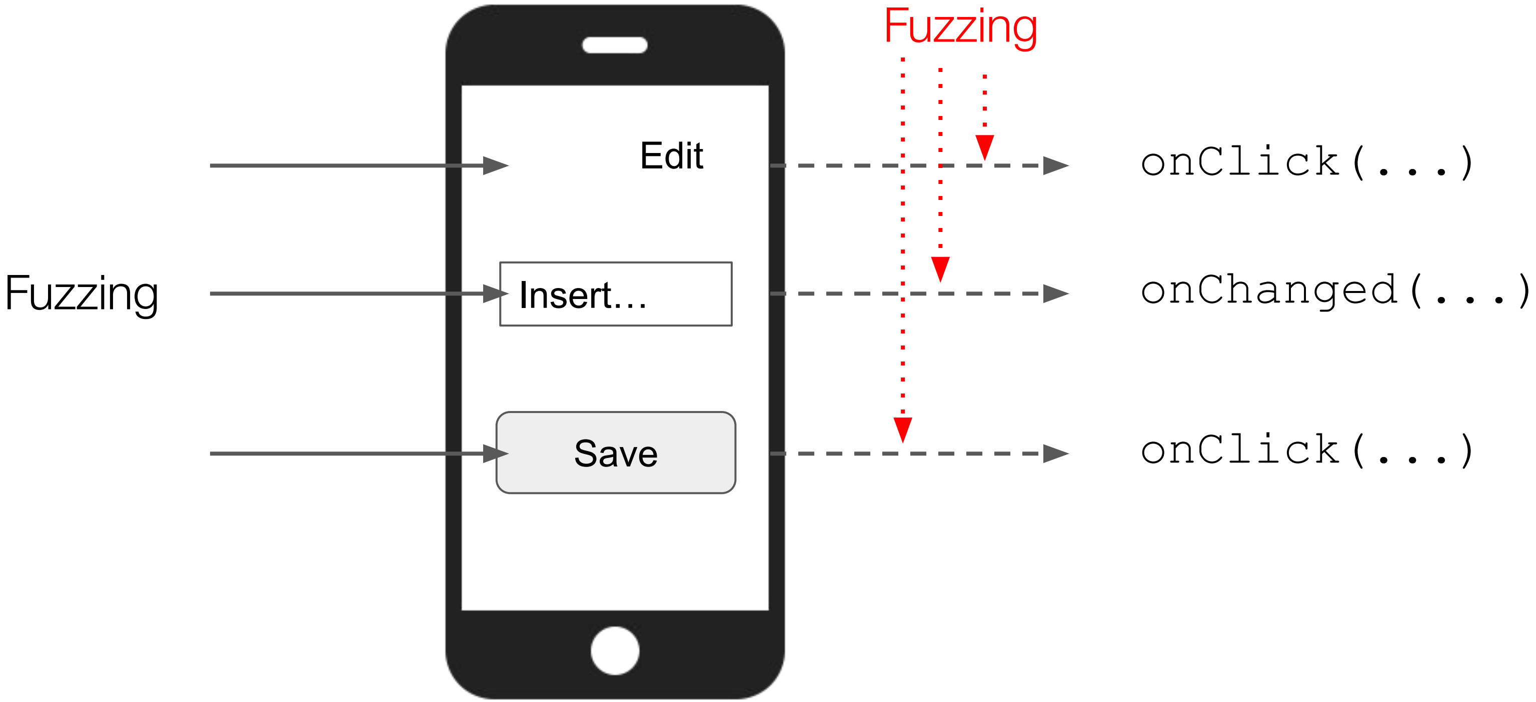
Task	Command	Bug	Detection	Return code	Avg. # of messages
Identity	`ICPushReaderCert`	Buffer overflow	Chip reboots	2	74
Identity	`ICsetAuthToken`	Buffer overflow	Stack canary	2	475
Identity	`WICaddAccessControlProfile`	Null-pointer dereference	Chip halts	4	57
Identity	`WICbeginAddEntry`	Null-pointer dereference	Chip halts	4	99
Identity	`WICfinishAddingEntries`	Null-pointer dereference	Chip halts	4	82
Identity	`ICstartRetrieveEntryValue`	Null-pointer dereference	Chip halts	4	105
Keymaster	`FinishAttestKey`	N/A	Chip reboots	2	257
Keymaster	`IdentityFinishAttestKey`	N/A	Chip reboots	2	192

Table 2: Results of fuzzing the Titan M firmware, version *0.0.3/brick_v0.0.8292-b3875afe2*

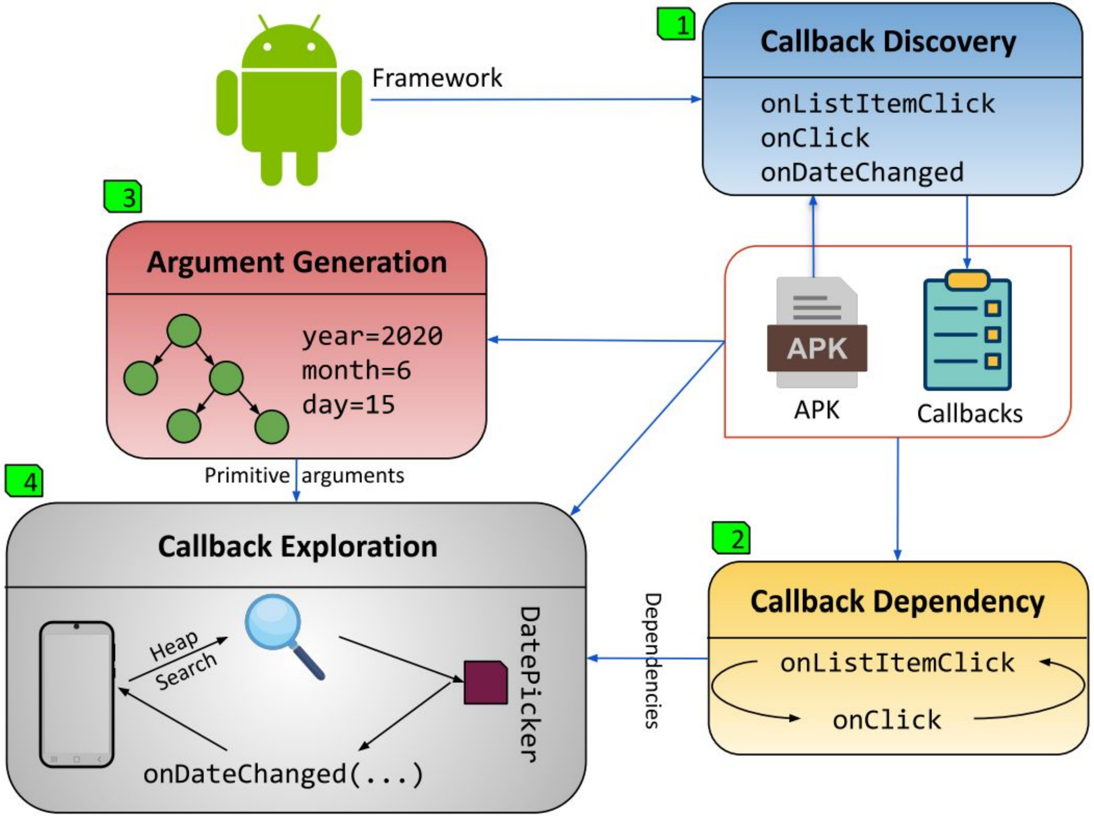
Task	Command	Bug	Detection	Return code	Avg. # of messages
Identity	`WICfinishAddingEntries`	Null-pointer dereference	Chip halts	4	72
Identity	`ICstartRetrieveEntryValue`	Null-pointer dereference	Chip halts	4	126

Reversing and Fuzzing the Google Titan M Chip
In Procs. of the Reversing and Offensive-oriented Trends Symposium (ROOTS), 2021

Fuzzing Android Apps



Columbus: Fuzzing Android Apps



Results & Outcomes

Columbus has **5% - 31%** more in **average coverage** than existing tools

Discovers **1.23 - 5.48** times more **crashes**

Columbus found **70 crashes** in 54 popular apps

COLUMBUS: Android App Testing Through Systematic Callback Exploration
Procs. of the International Conference on Software Engineering (ICSE), 2023.

AVR Lifecycle

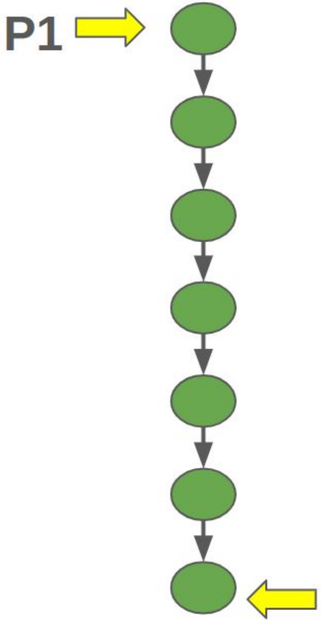
Detect

Analyze

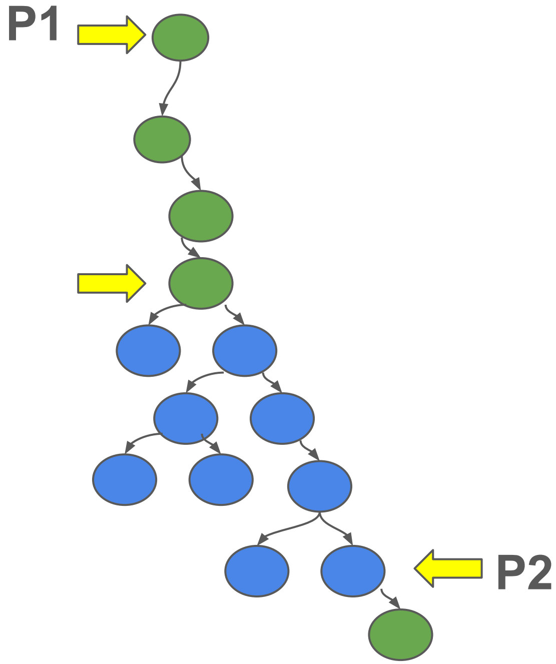
Patch

Prevent

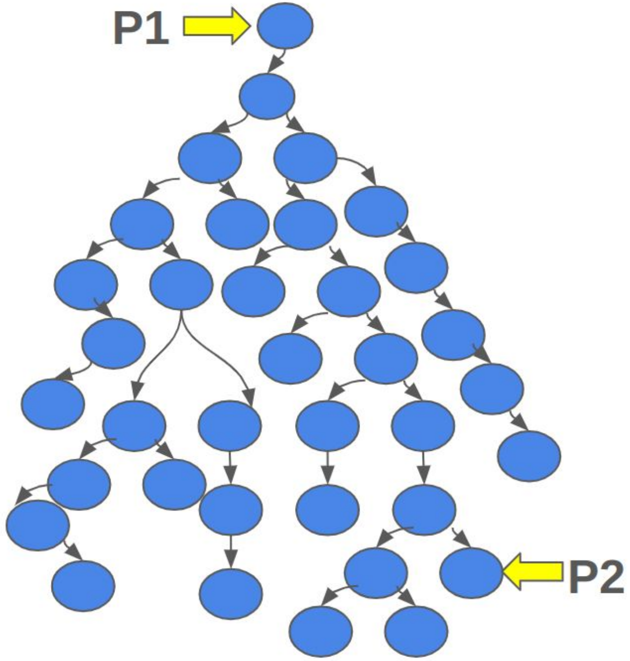
Interleaved Symbolic Execution



concrete execution



Interleaved symbolic execution

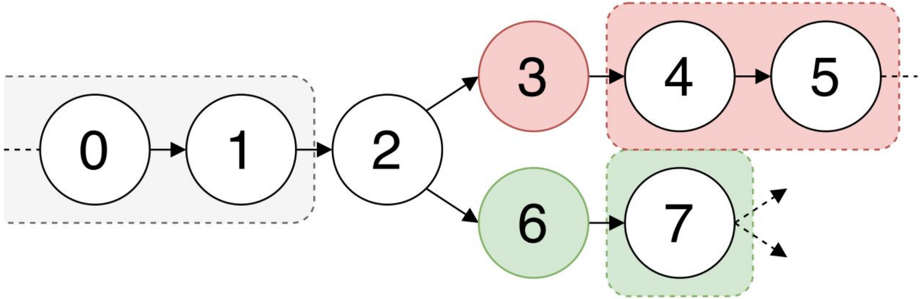


under-constrained symbolic exe

SYMBION: Interleaving Symbolic with Concrete Execution
Procs. of the IEEE Conference on Communications and Network Security (CNS), 2020.

ML-guided Symbolic Execution

Train a classifier to select the branch path more likely to lead to vulnerabilities

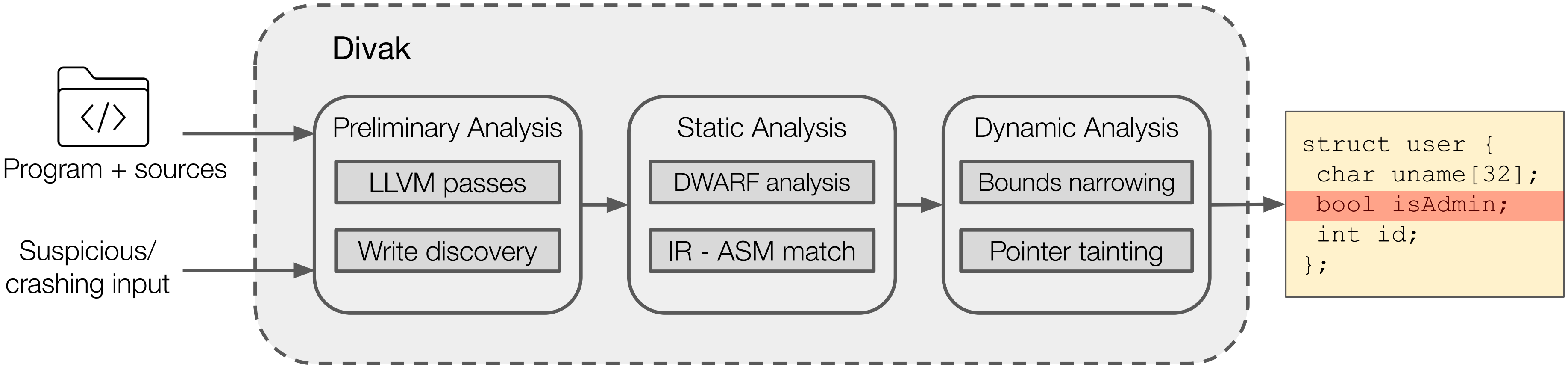


SyML reaches both **more and different** vulnerabilities on CGC dataset

Successful on 3 real-world Linux CVEs, **knowledge transfer**

SyML: Guiding Symbolic Execution Toward Vulnerable States Through Pattern Learning
Procs. of the International Symposium on Research in Attacks, Intrusions and Defenses (RAID), 2021.

Divak: Characterizing OOB writes



```
struct user {
 char uname[32];
 bool isAdmin;
 int id;
};
```

Non-invasive approach && detect intra-object OOBs

Divak: Non-invasive Characterization of Out-Of-Bounds Write Vulnerabilities
Conference on Detection of Intrusions and Malware and Vulnerability Assessment (DIMVA), 2023.

AVR Lifecycle

Detect

Analyze

Patch

Prevent

Patching Monolithic Firmware

Creating a Patch

What’s the input? No standard sources of input, numerous hardware peripherals

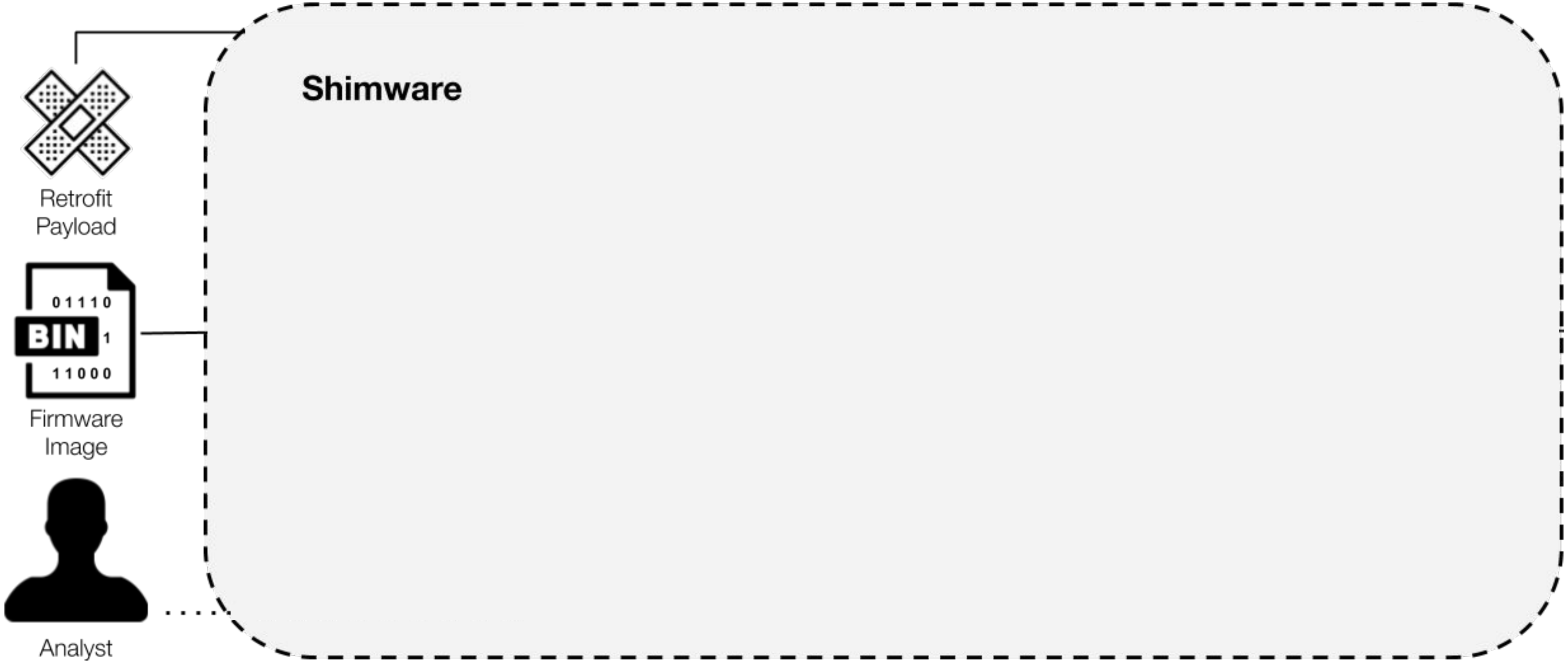
Inserting a Patch

Where? We cannot simply inject & shift && we have space issues

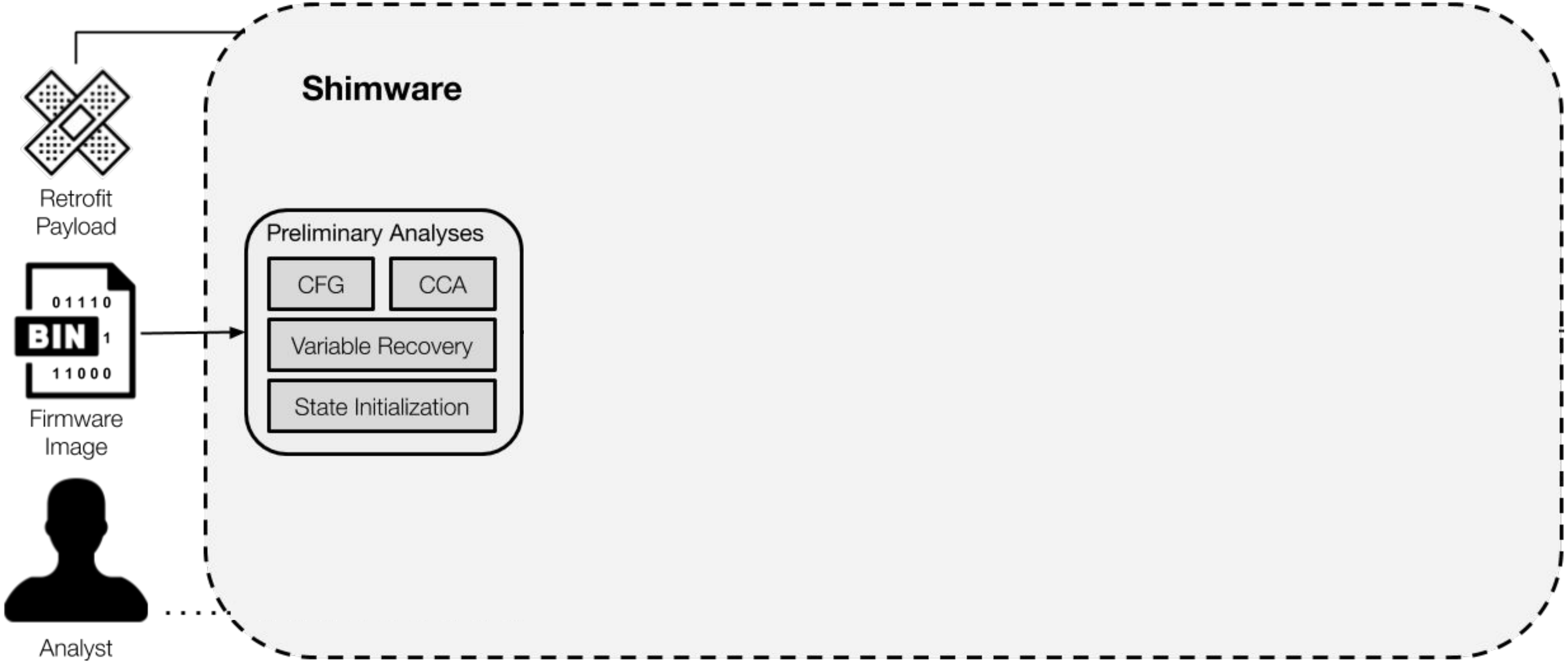
Deploying a Patch

How? Verification mechanism to preserve integrity

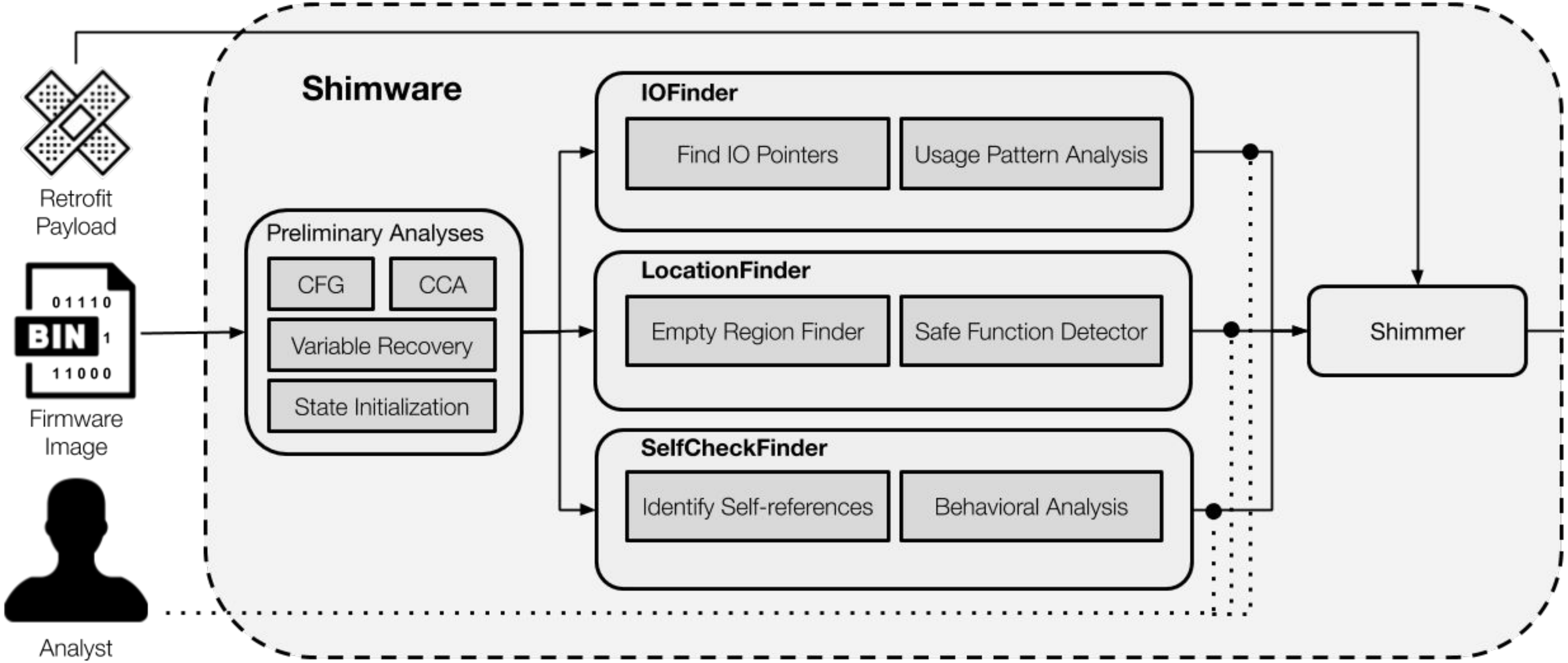
Retrofitting Monolithic Firmware



Retrofitting Monolithic Firmware



Retrofitting Monolithic Firmware



Research paper currently under submission

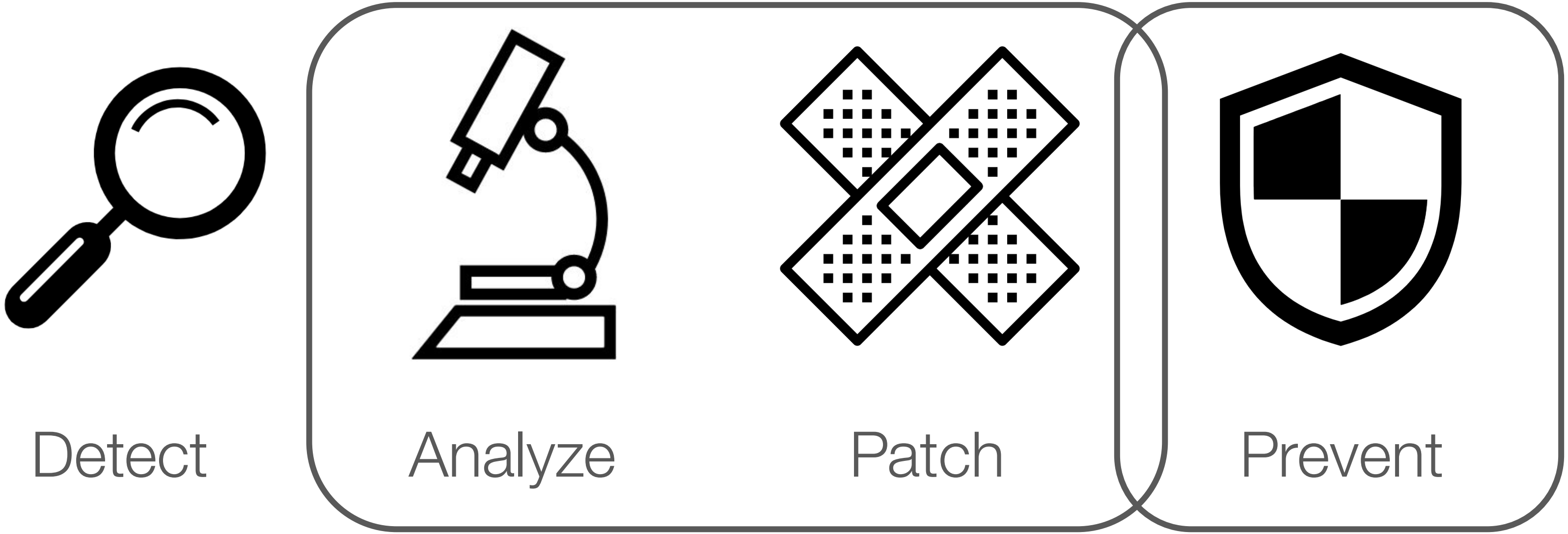
Coordinated Vulnerability Disclosure

We established a university-wide policy on coordinated vulnerability disclosure

- Clear to researchers & students how to behave (+ guidelines)
- Leverage in demanding that researchers follow these procedures
- Provides researchers with assurance that they will be protected
- Clear to recipients of disclosure notices how we handle the process

Operationalizing Cybersecurity Research Ethics Review: From Principles and Guidelines to Practice
Procs. of the International Workshop on Ethics in Computer Security (EthiCS), 2023.

What’s next?



Detect

Analyze

Patch

Prevent

Thanks!
Questions?

Andrea Continella
<a.continella@utwente.nl>
https://conand.me
🐦 @_conand