



Security of IoT and embedded devices: Insights from an academic perspective

Andrei Costin, PhD
firmware.re

firmware · re

Who are we?



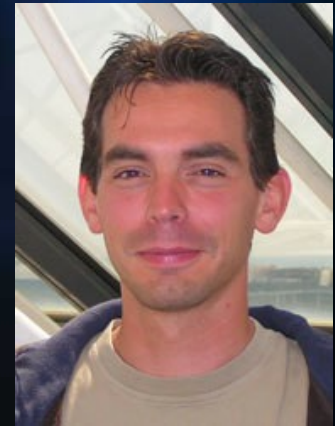
Andrei Costin



Jonas Zaddach



Aurélien Francillon
EURECOM



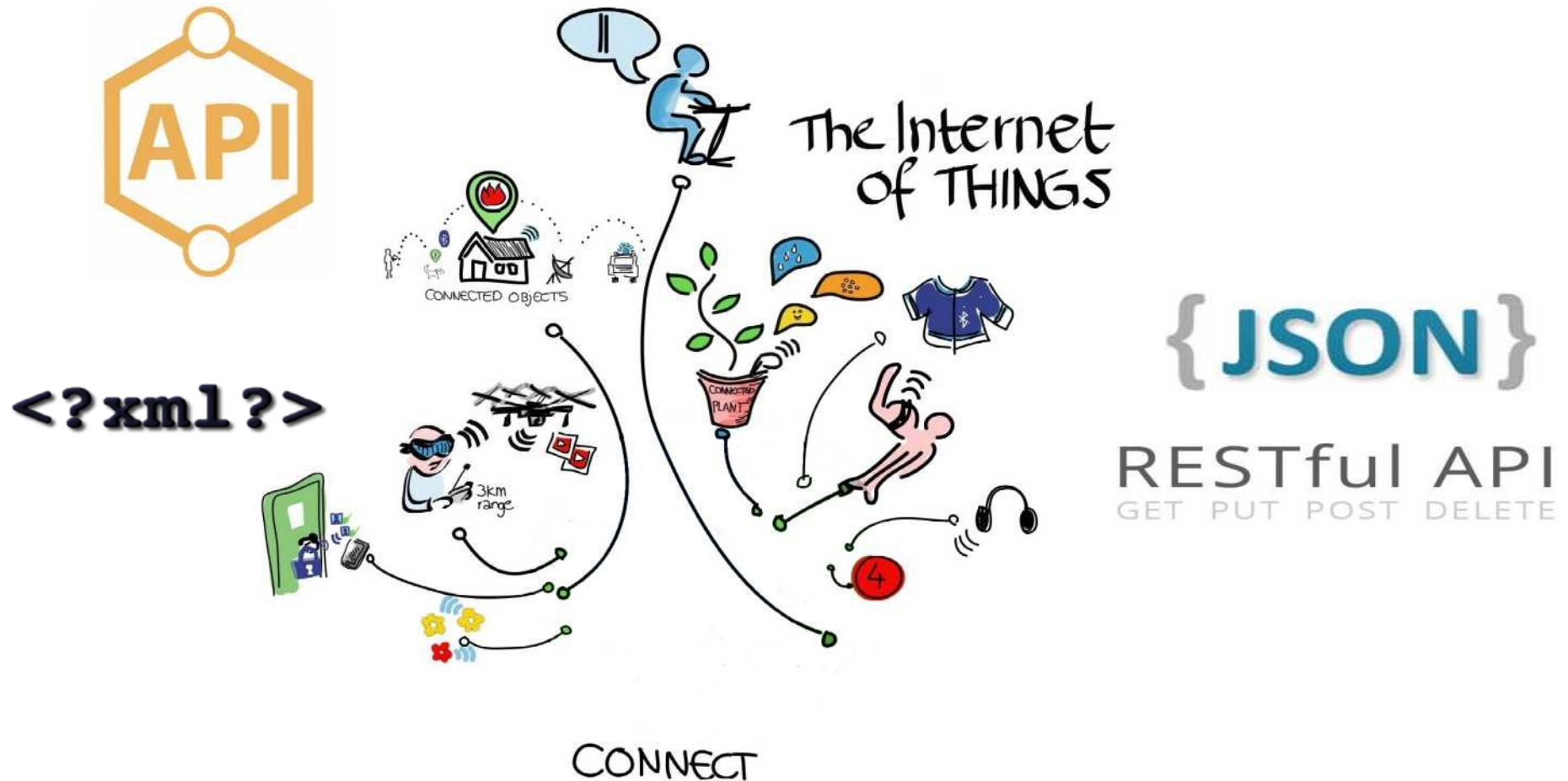
Davide Balzarotti
EURECOM

Embedded Devices Are Everywhere



by Wilgebroed on Flickr [CC-BY-2.0]

Embedded Devices Smarter and More Complex



by Wilgenbroed on Flickr [CC-BY-2.0]

Embedded Devices More Interconnected



<?xml?>



CONNECT
THE WORLD
by Wilgenbroed on Flickr [CC-BY-2.0]

{JSON}

RESTful API
GET PUT POST DELETE



Software is Everywhere

- Embedded devices are **diverse** – but all of them run **software**, commonly referred to as **firmware**



Importance of Embedded Systems' Security

- Embedded devices are **ubiquitous**
 - Even invisible, they are essential to our lives
- Can operate for many years
 - **Legacy systems**, no (security) updates
- Have a **large attack surface**
 - Web interfaces
 - Networking services
 - Forgotten debug interfaces
 - ...


Many Examples of Insecure Embedded Systems

- Routers

Firefox Reverse Engineering a D-Link B...
www.devttys0.com/2013/10/reverse-engineering-a-d-link-backdoor/

Based on the source code of the HTML pages and some Shodan [search results](#) D-Link devices are likely affected:

- DIR-100
- DIR-120
- DI-624S
- DI-524UP
- DI-604S
- DI-604UP
- DI-604+
- TM-G5240



Additionally, several Planex routers also appear to use the same firmware:

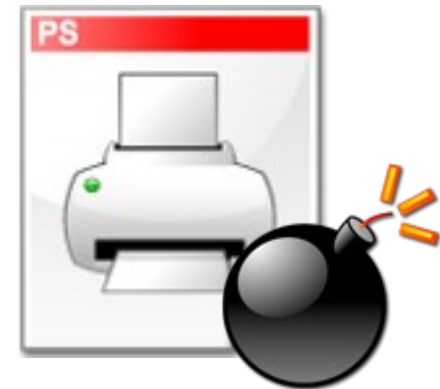
- BRL-04R
- BRL-04UR
- BRL-04CW

You stay classy, D-Link.

Many Examples of Insecure Embedded Systems

- Routers
- Printers

Networked printers at risk
(30/12/2011, McAfee Labs)



Many Examples of Insecure Embedded Systems

- Routers
- Printers
- VoIP

Cisco VoIP Phones Affected By On Hook Security Vulnerability (12/06/2012, Forbes)



Many Examples of Insecure Embedded Systems

- Routers
- Printers
- VoIP
- Cars

Hackers Reveal Nasty New Car Attacks – With Me Behind The Wheel (12/08/2013, Forbes)



Many Examples of Insecure Embedded Systems

- Routers
- Printers
- VoIP
- Cars
- Drones

Hacker Releases Software to Hijack Commercial Drones

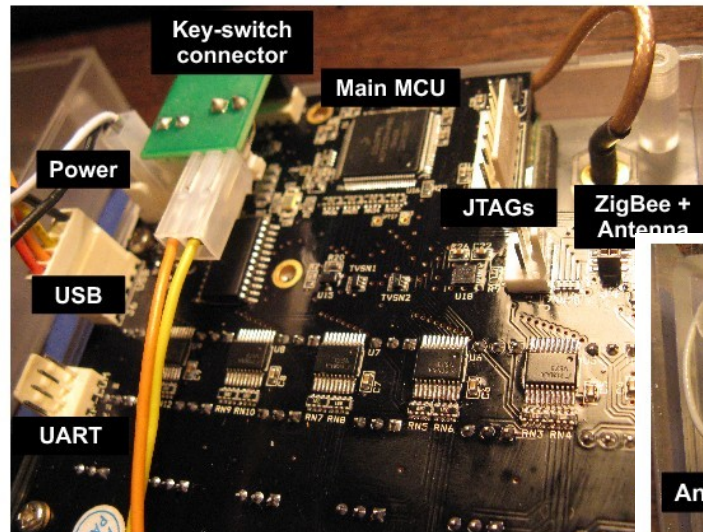
by BRYANT JORDAN on DECEMBER 9, 2013

 Like 489 people like this. Be the first of your friends.



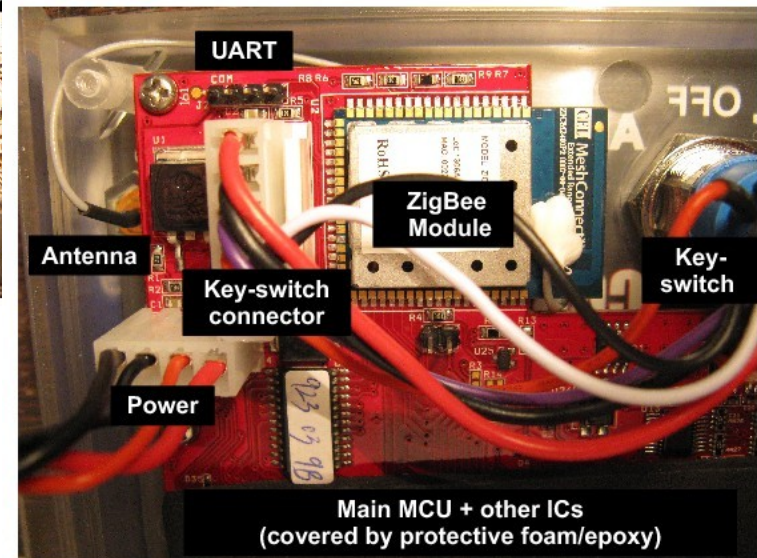
Many Examples of Insecure Embedded Systems

- Routers
- Printers
- VoIP
- Cars
- Drones
- Fireworks



Firing Module

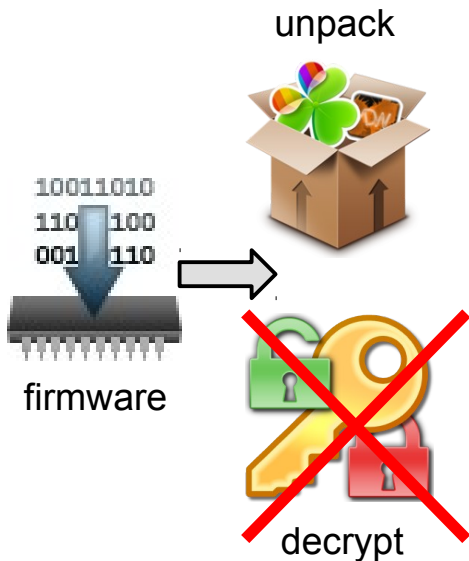
Remote Control



Manual Analysis Process



Manual Analysis Process

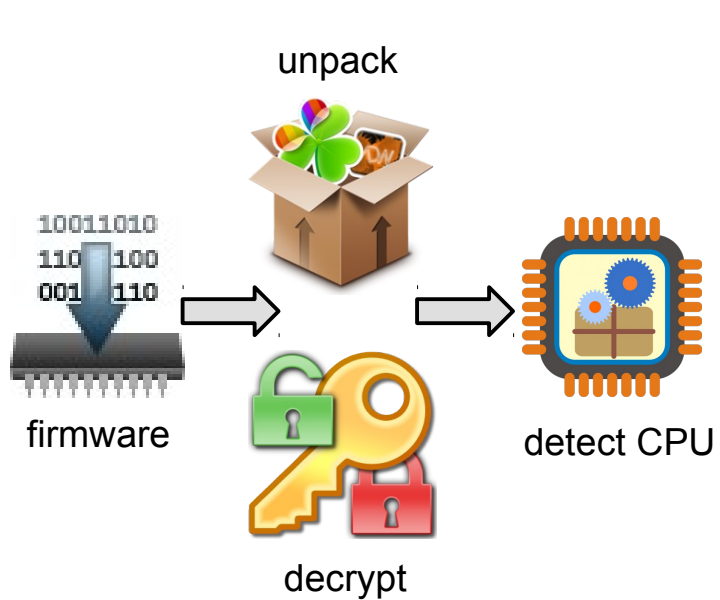


IHEX format

```
:10000000C942A000C9434000C9434000C943400AA
:100010000C9434000C9434000C9434000C94340090
:100020000C9434000C9434000C9434000C94340080
:100030000C9434000C9434000C9434000C94340070
:100040000C9434000C9434000C9434000C94340060
:100050000C94340011241FBECFE5D8E0DEBFCDBF25
:100060000E9436000C9445000C9400008FEF87BB73
:100070002CE231E088B3809588BB80E197E2F901FA
:0E0080003197F1F70197D9F7F5CFF894FFCF3C
:00000001FF
```

plain text firmware

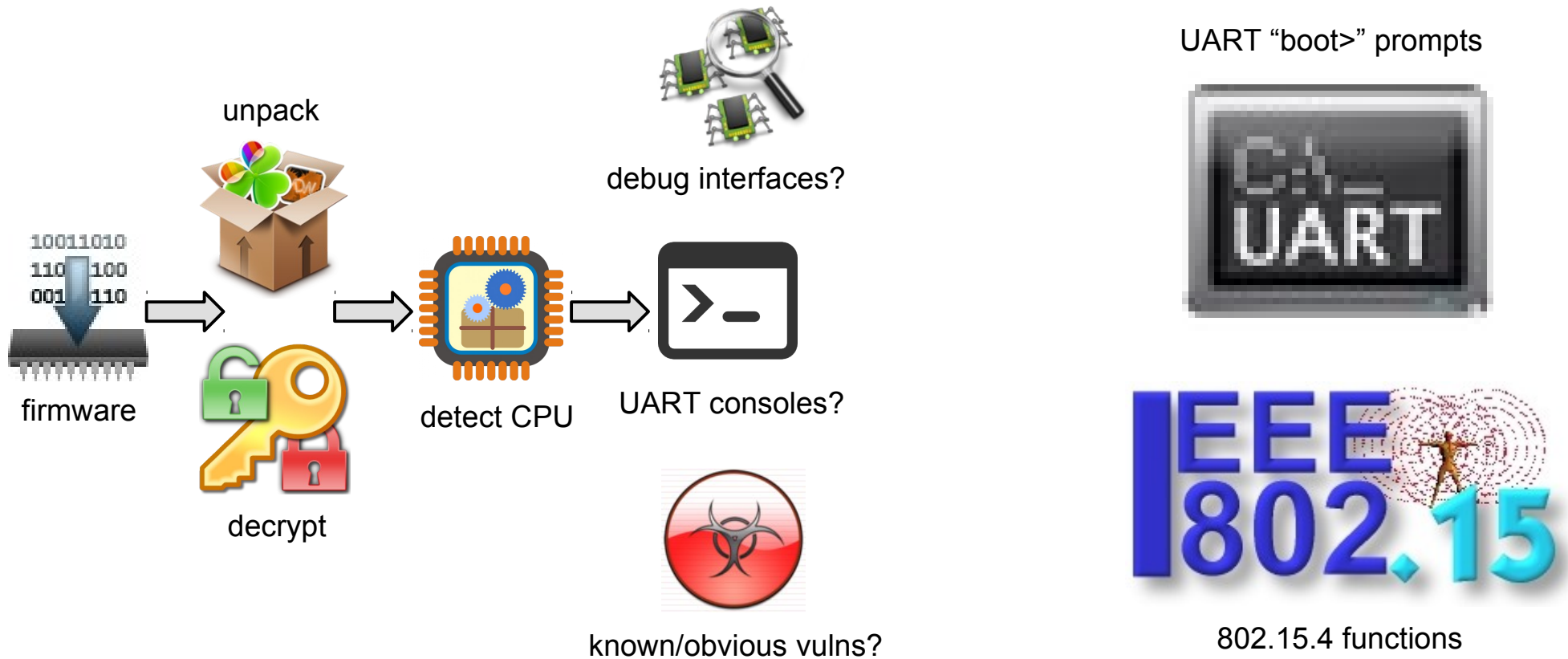
Manual Analysis Process



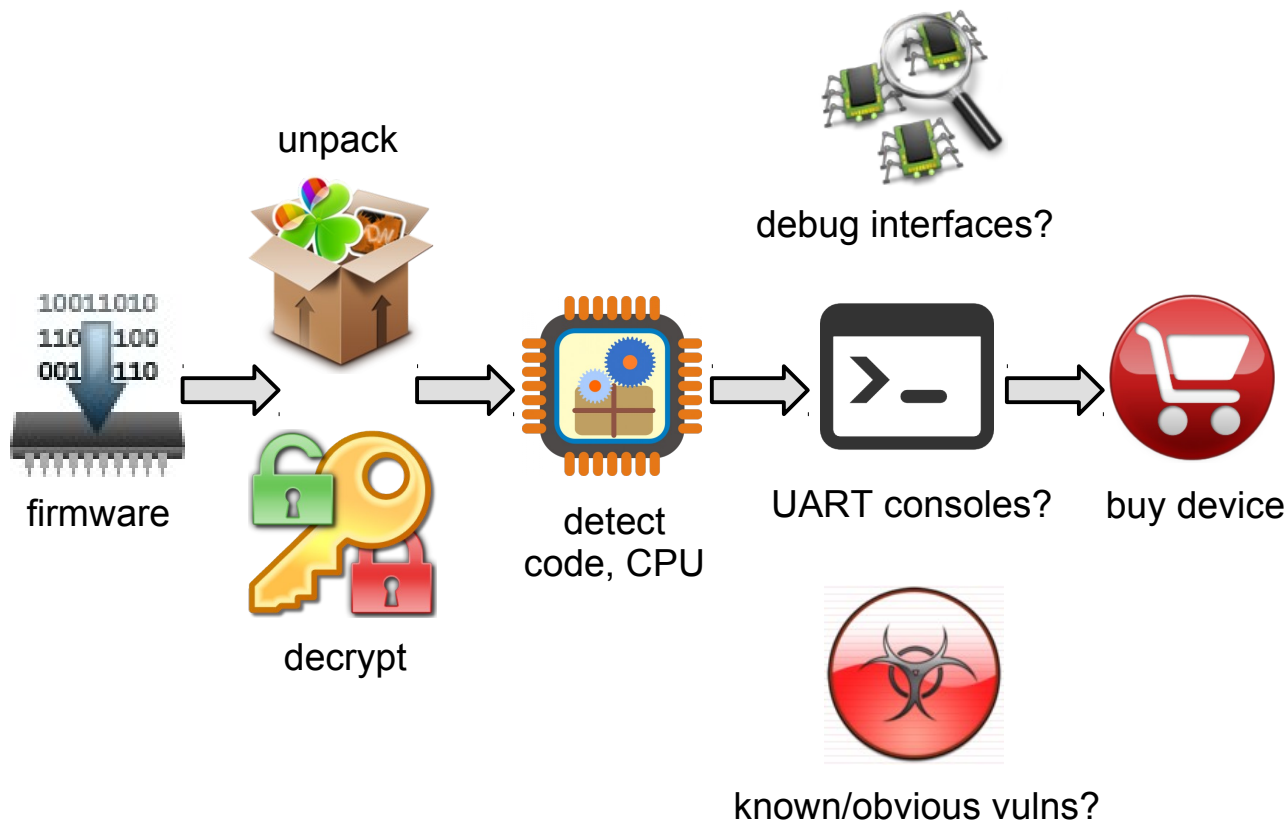
Motorola m68k-based CPU



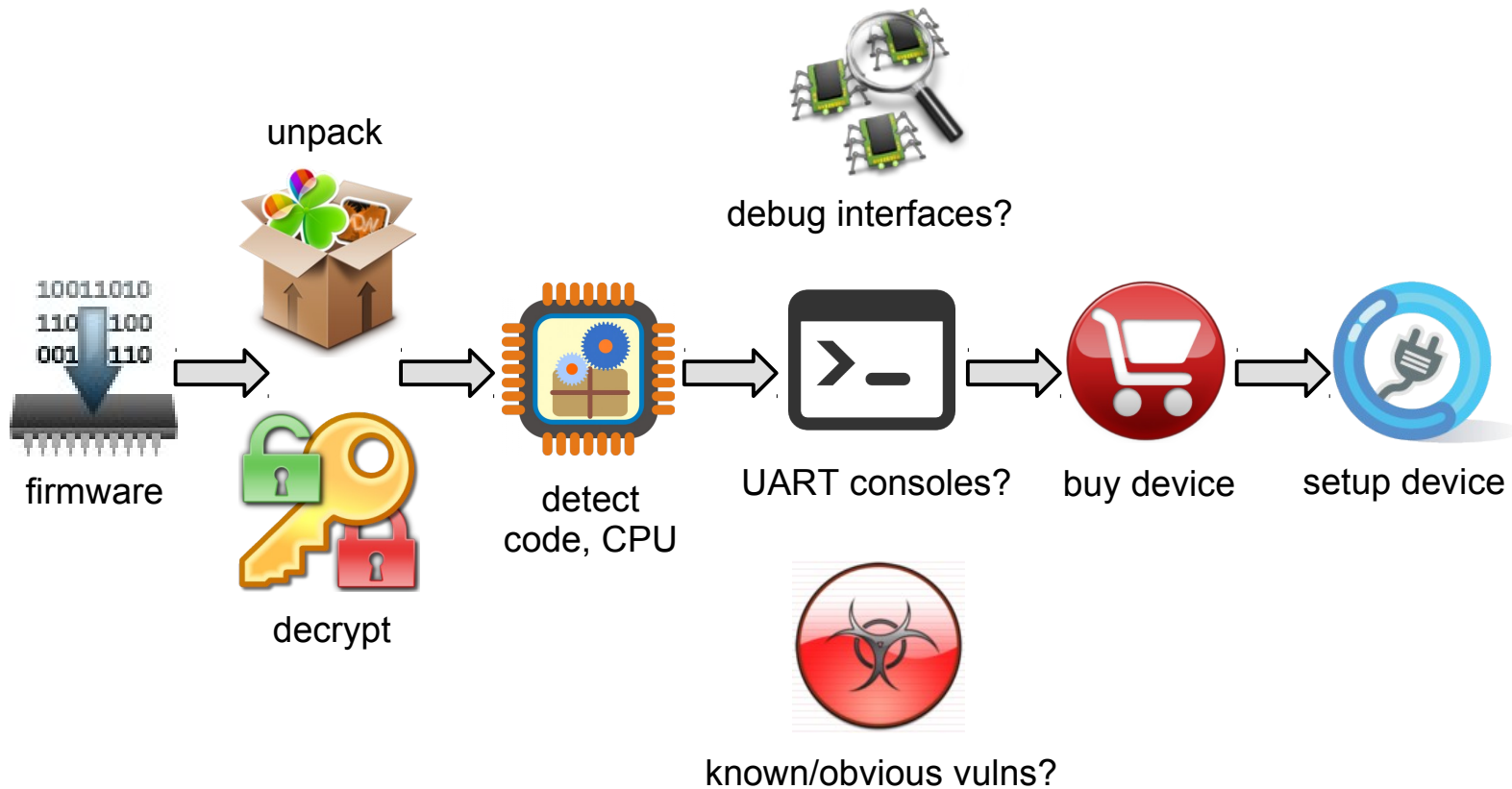
Manual Analysis Process



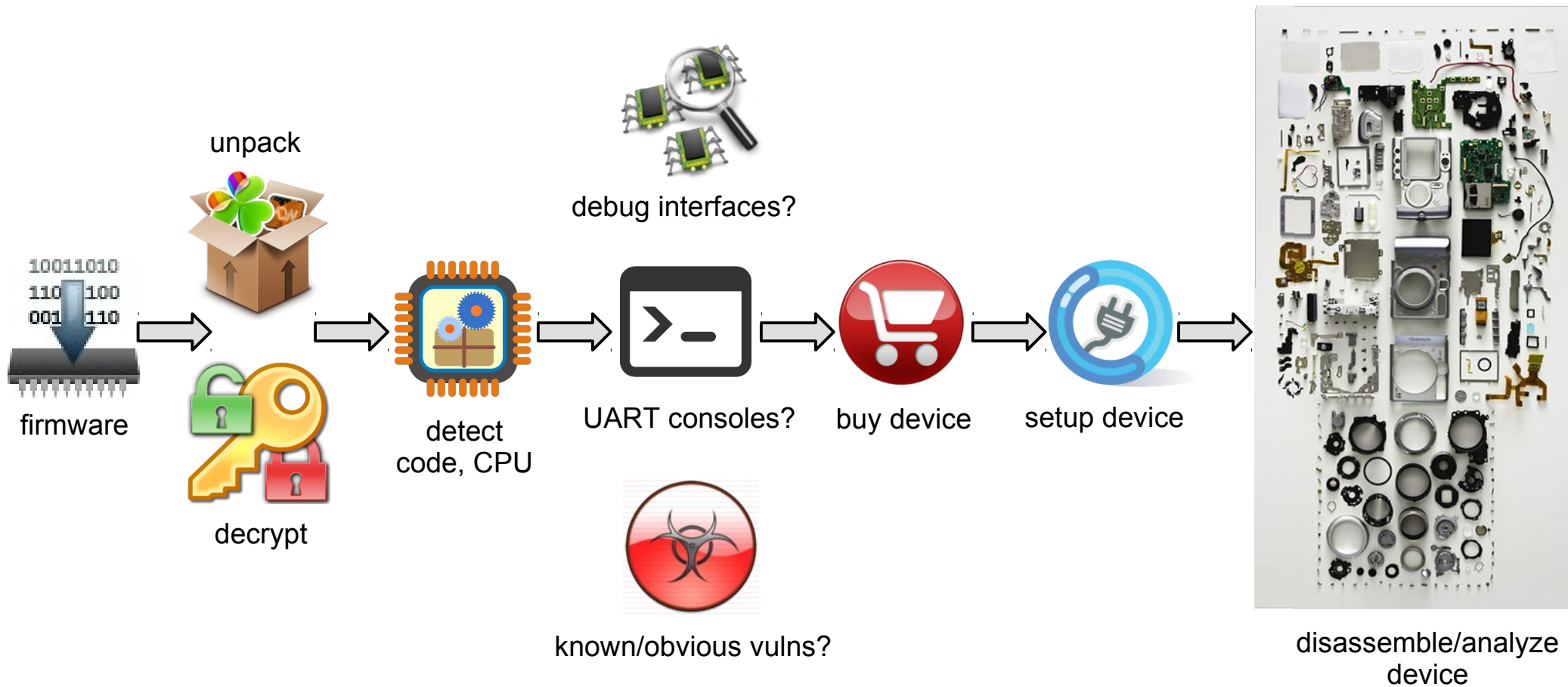
Manual Analysis Process



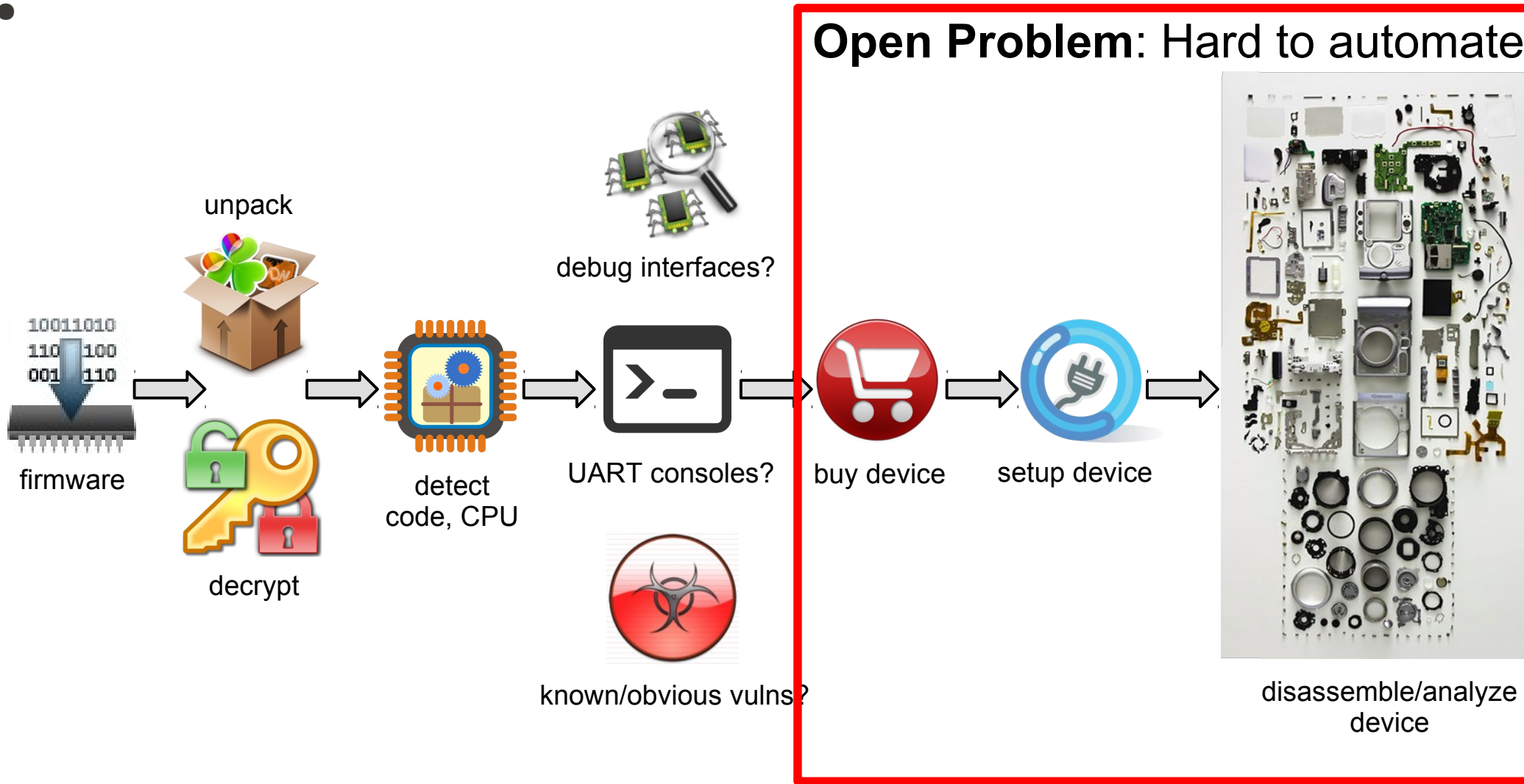
Manual Analysis Process



Manual Analysis Process

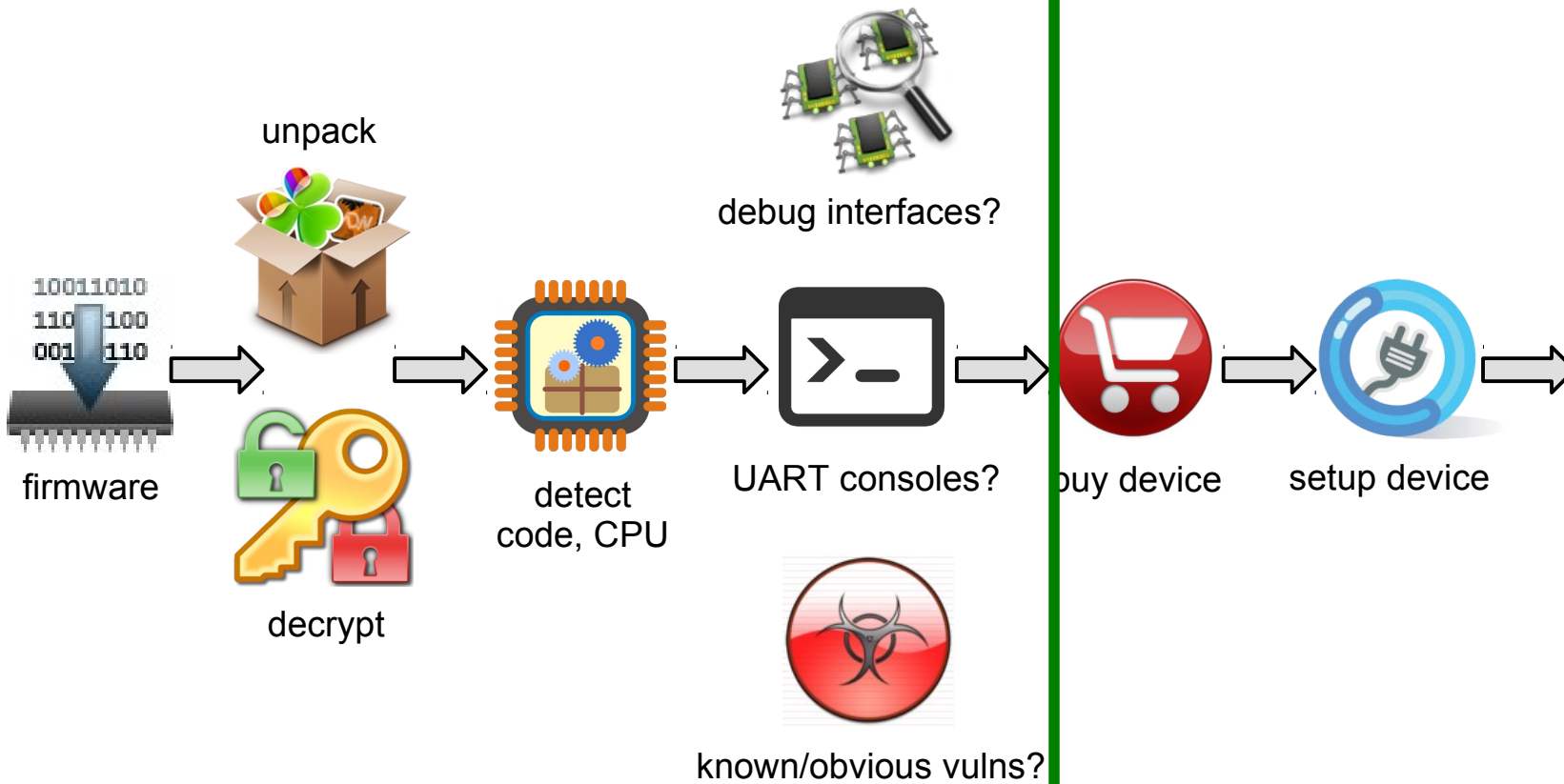


Manual Analysis Process



Manual Analysis Process

Goal: Automate these steps



disassemble/analyze device

Research Goal

Perform a **large scale analysis** to better **understand**, **analyze**, and **classify** the firmware images



Challenges

- Large number of devices → Analysis **without devices**
- Large number of firmware files → **Scalable** architectures
- Highly heterogeneous systems → **Generic** techniques
- Increasingly “smart”, “connected” → Focus on **web interfaces & APIs**
- Highly unstructured firmware data → **Large** dataset **classification**
- Vulnerable devices exposed → **Technology-independent** device **fingerprinting**

Our Project: First of its kind

Upload Files

Project Info

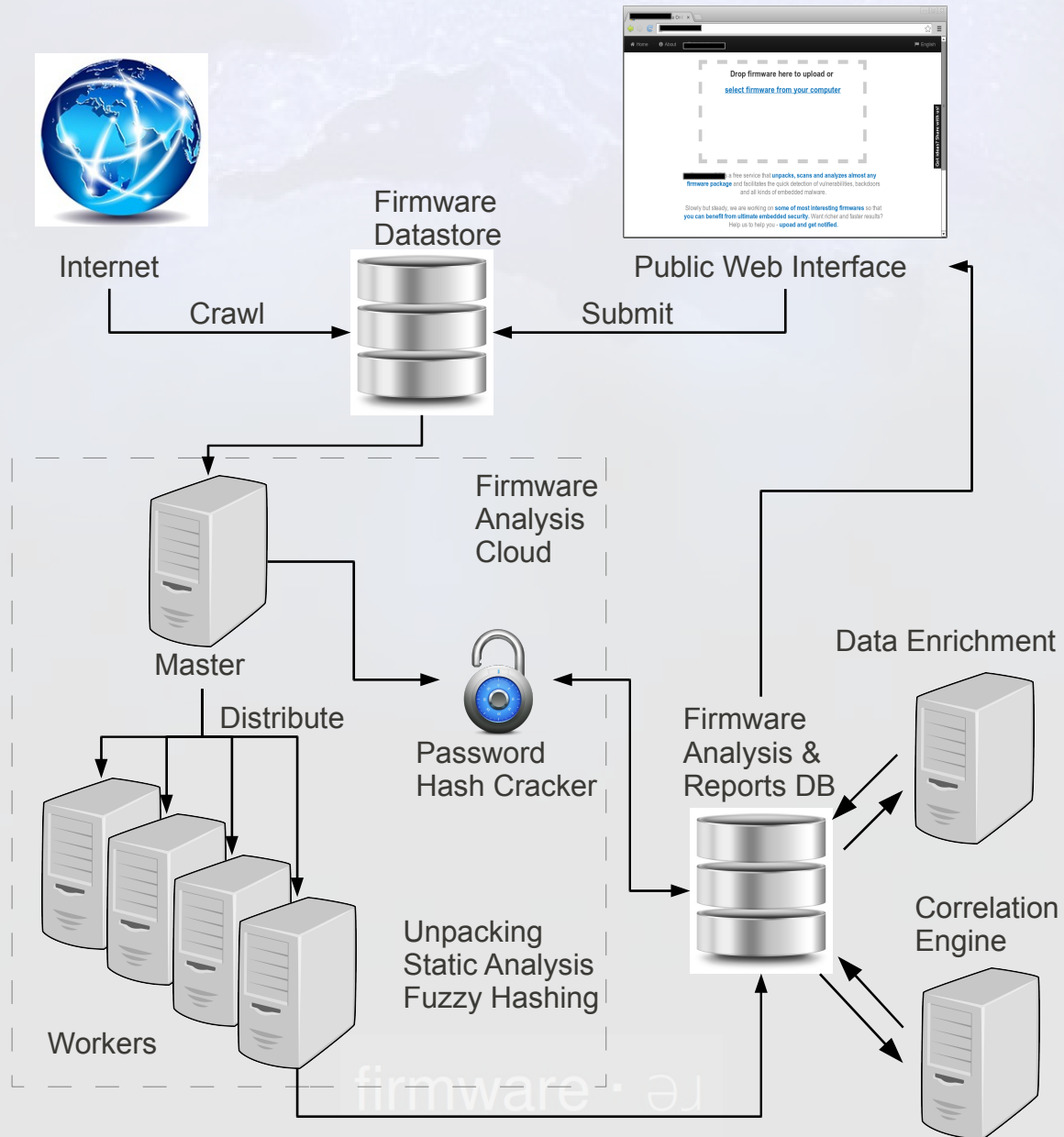
Some Samples

To start, drag-n-drop firmware here or
[select firmware from your computer](#)

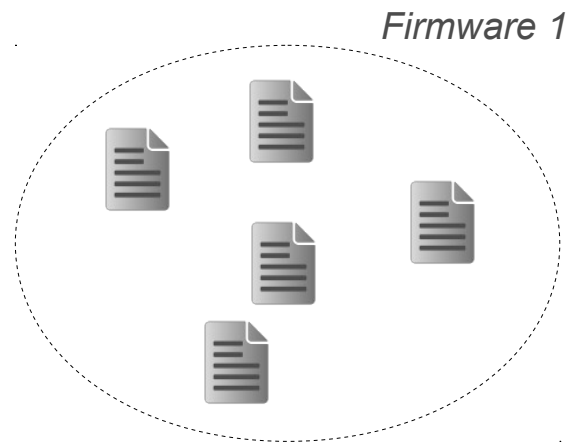
Got ideas? Share with us!



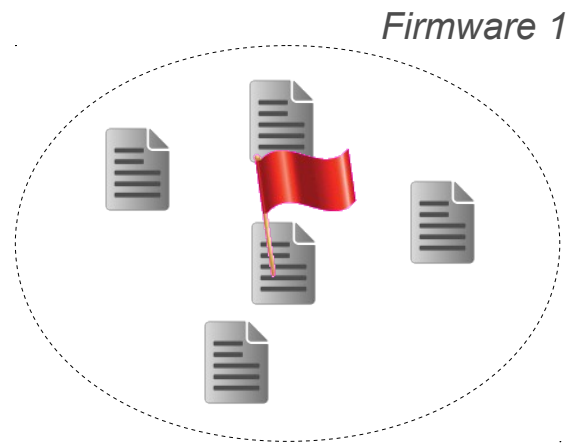
Our Architecture: First to be publicly presented



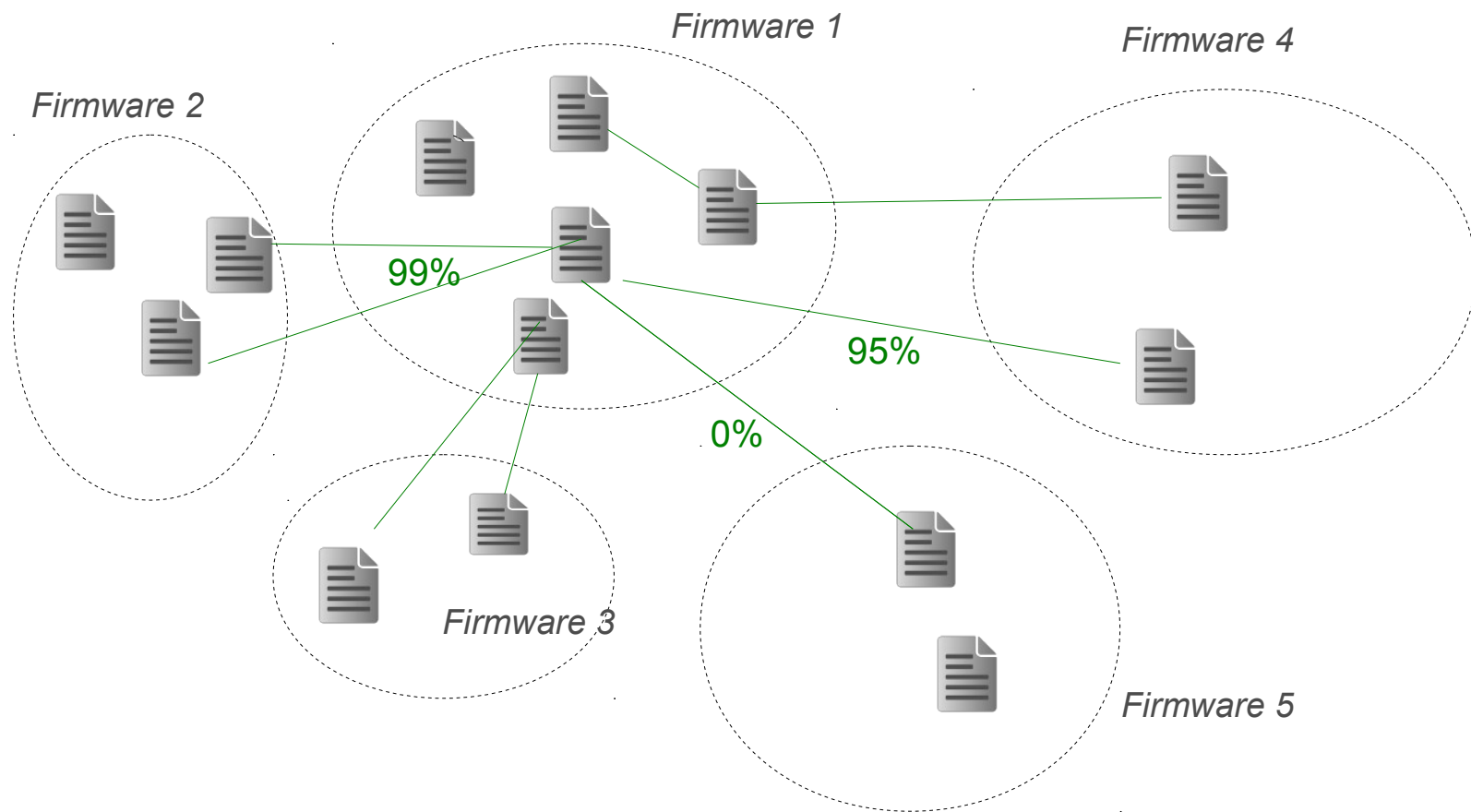
Example: Firmware content correlation



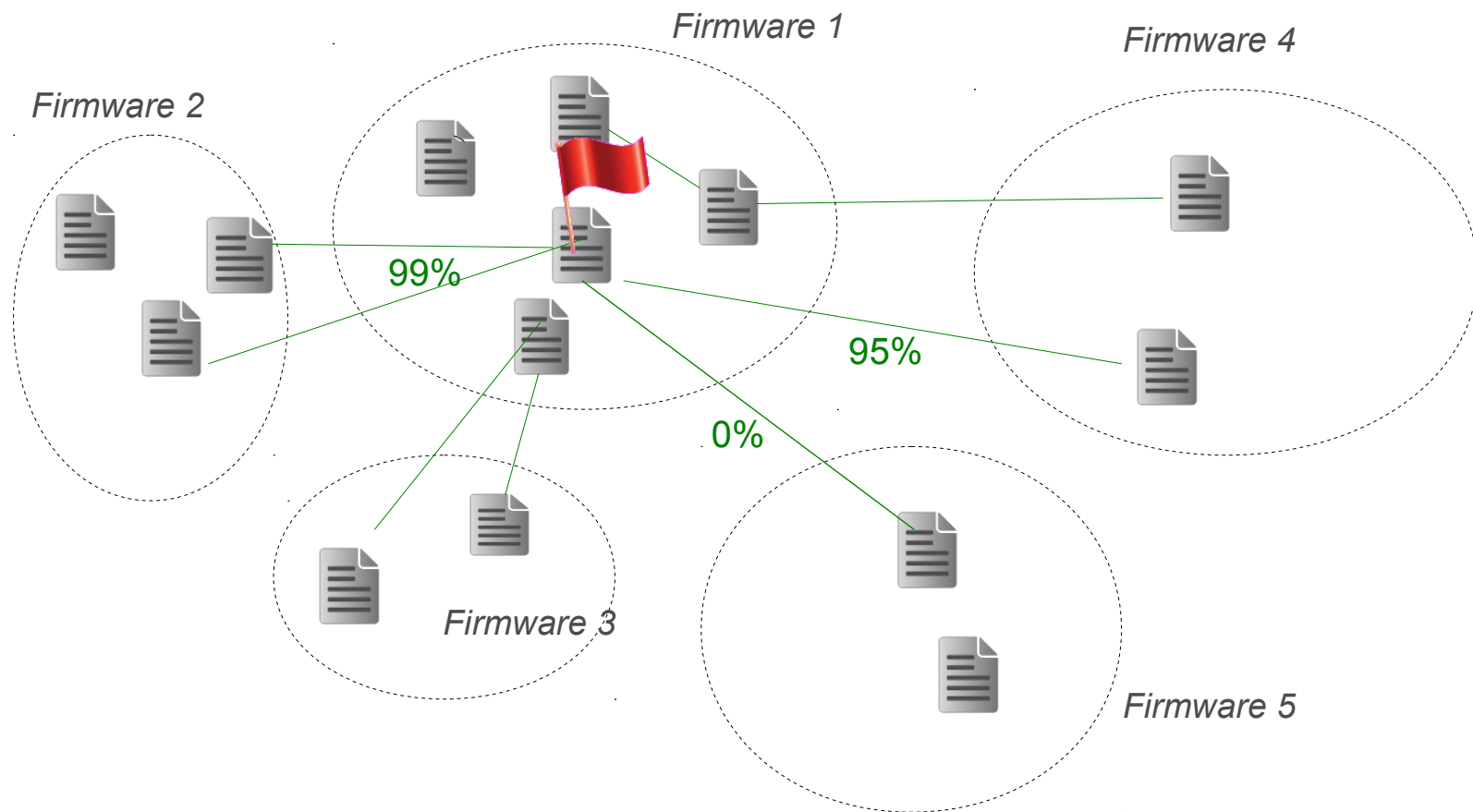
Example: Firmware content correlation



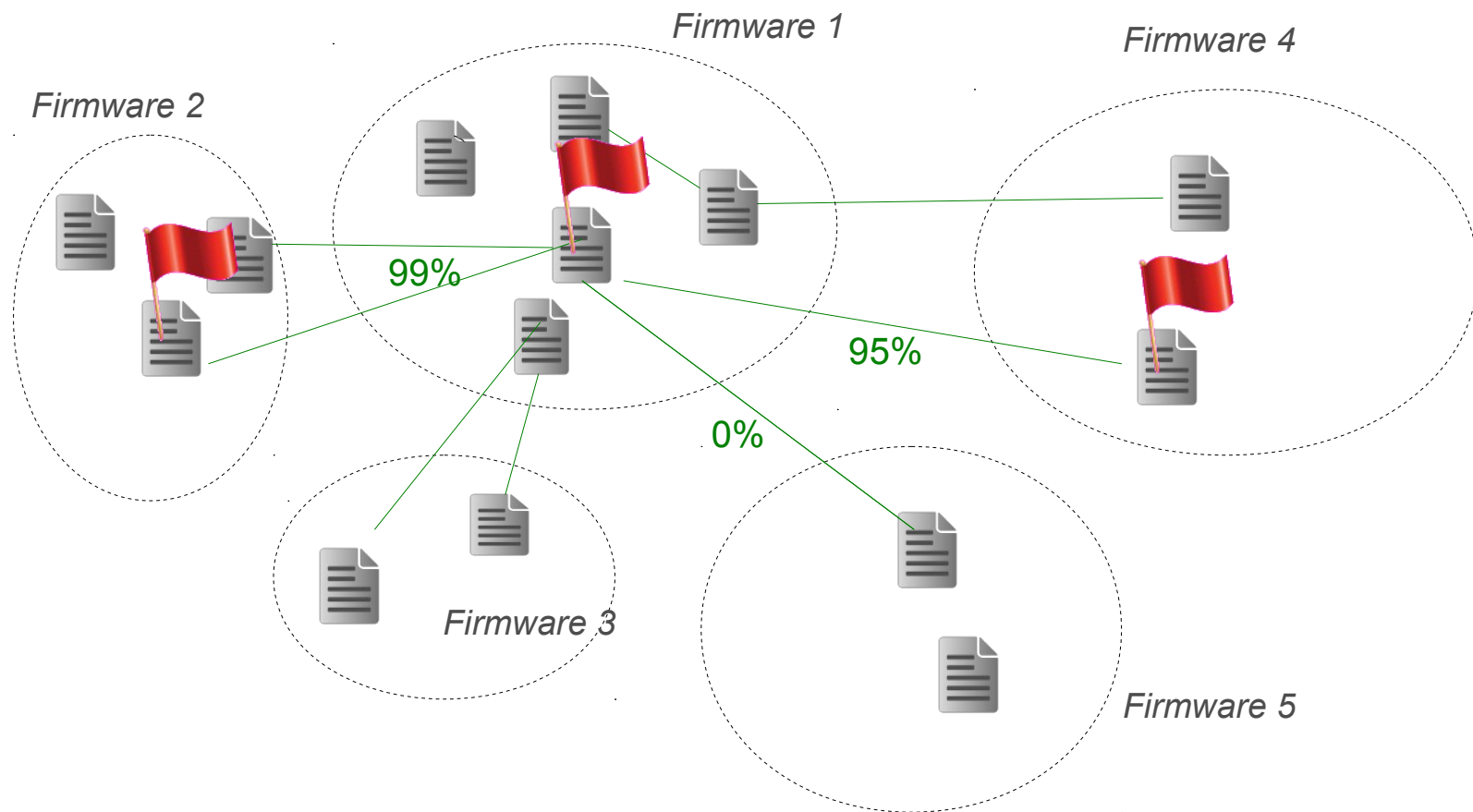
Example: Firmware content correlation



Example: Firmware content correlation



Example: Firmware content correlation



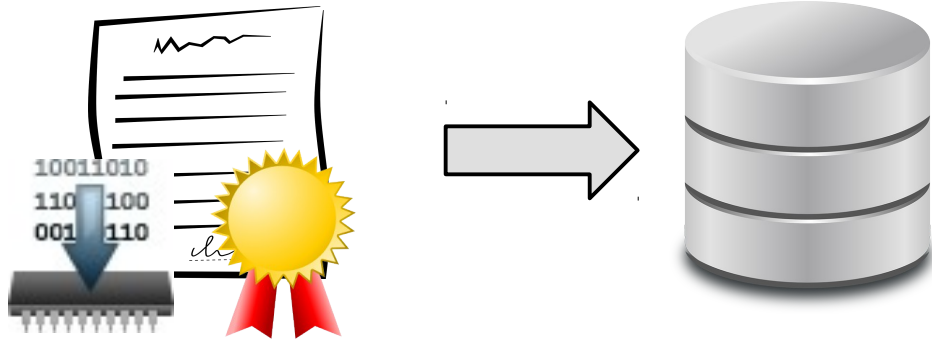
Example: Firmware HTTPS keys correlation



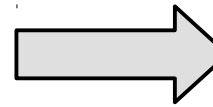
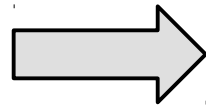
Example: Firmware HTTPS keys correlation



Example: Firmware HTTPS keys correlation

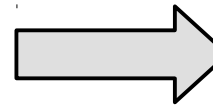
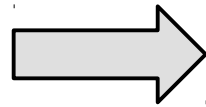


Example: Firmware HTTPS keys correlation



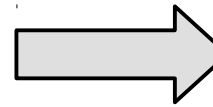
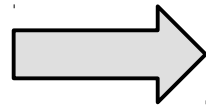
Vendor A

Example: Firmware HTTPS keys correlation



Vendor A

Example: Firmware HTTPS keys correlation



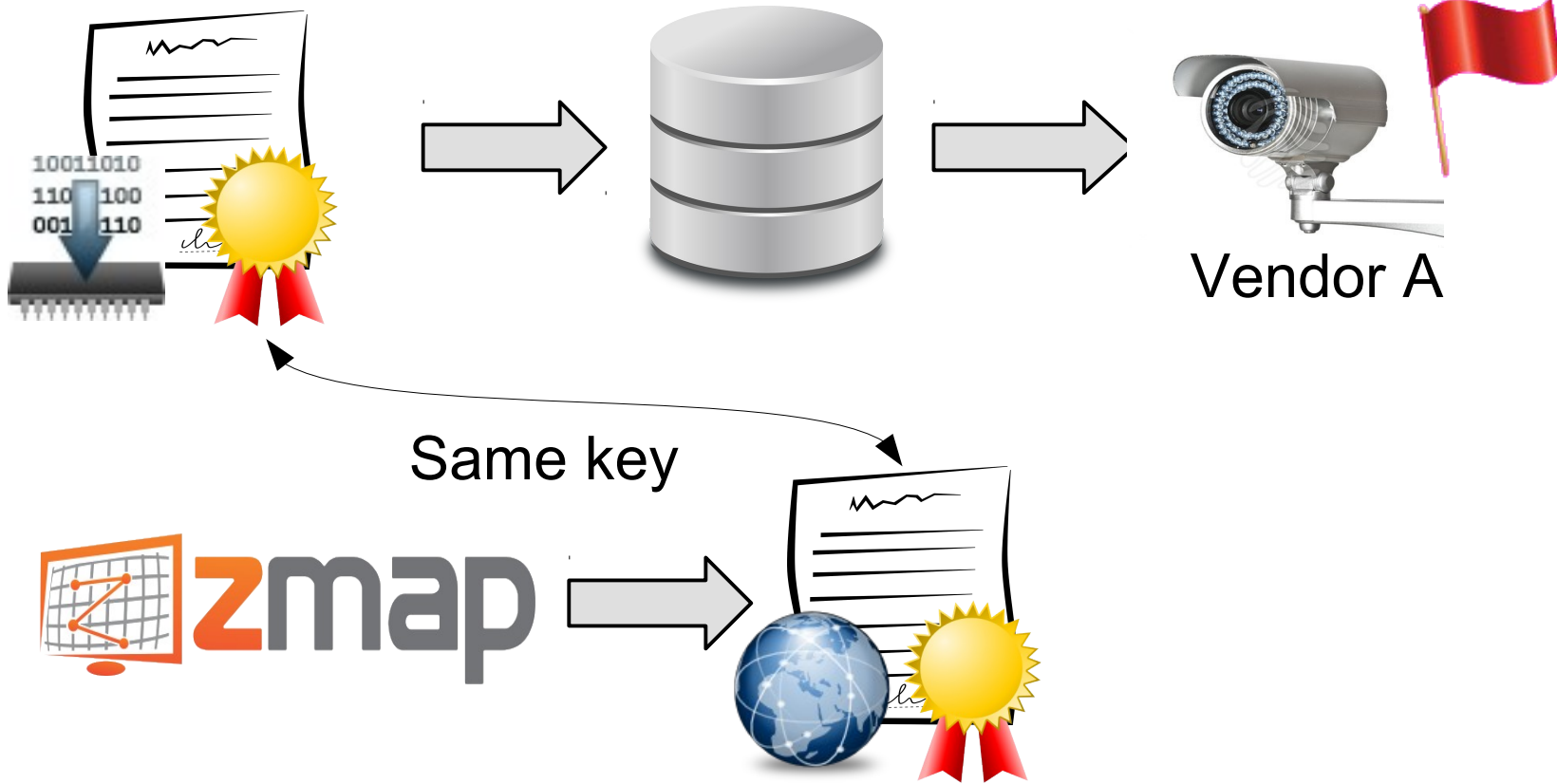
Vendor A



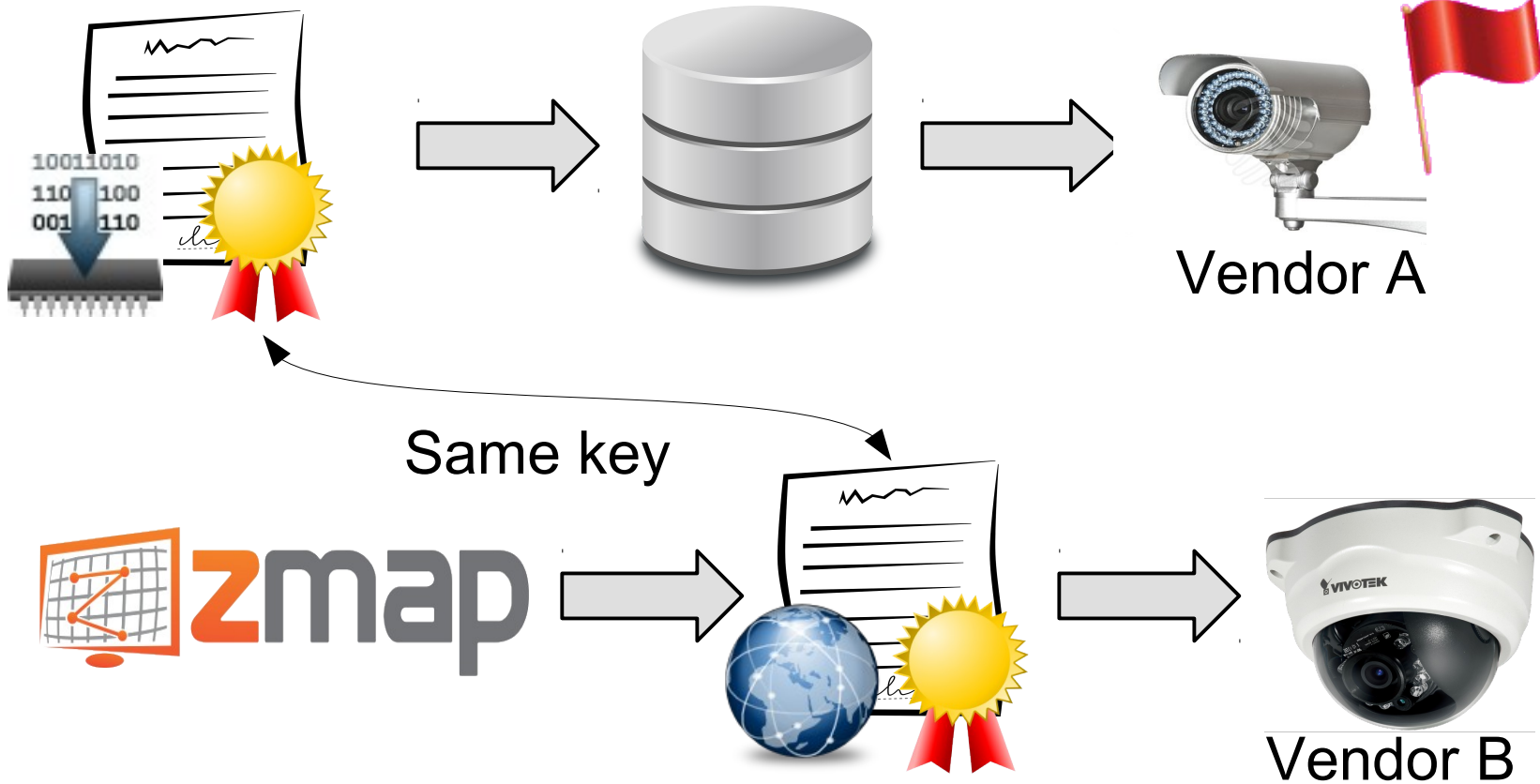
Example: Firmware HTTPS keys correlation



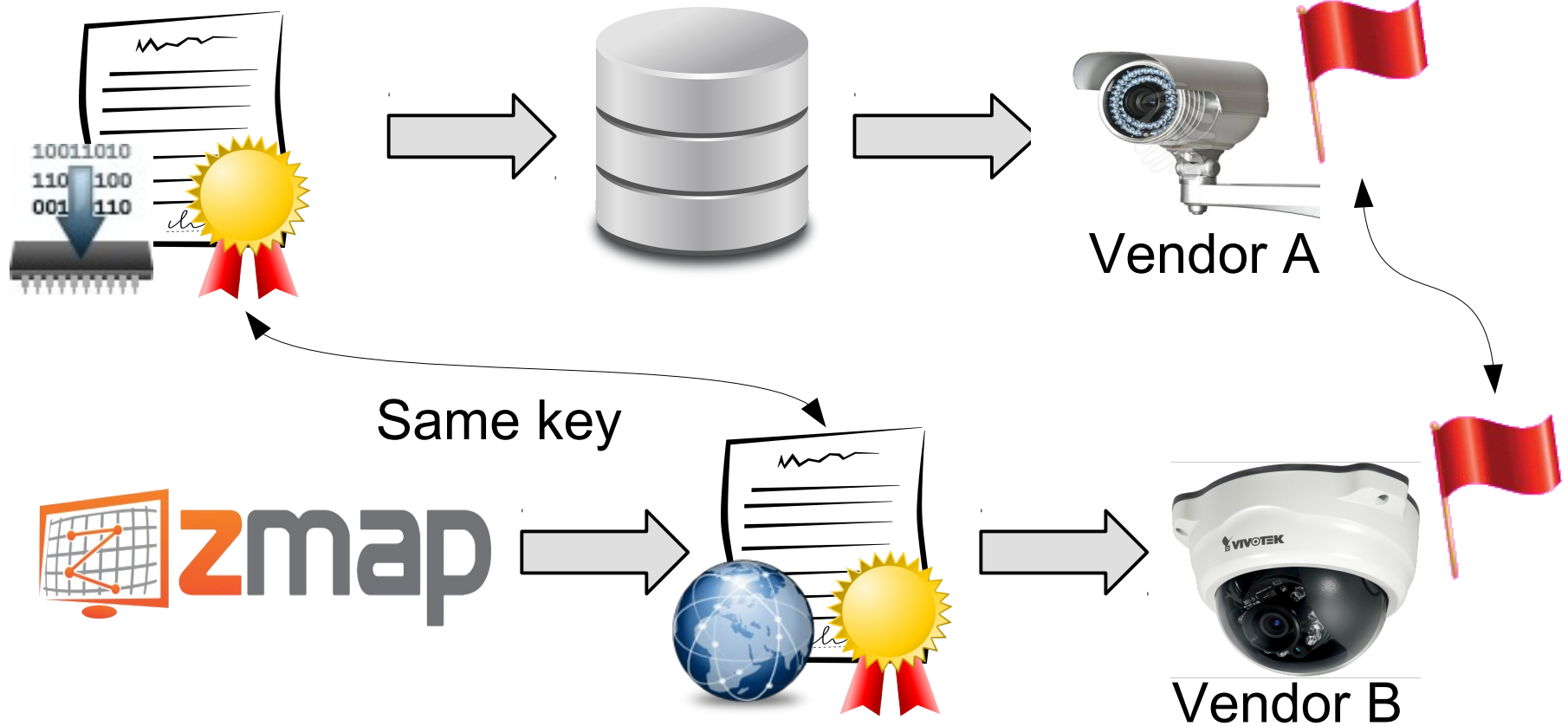
Example: Firmware HTTPS keys correlation



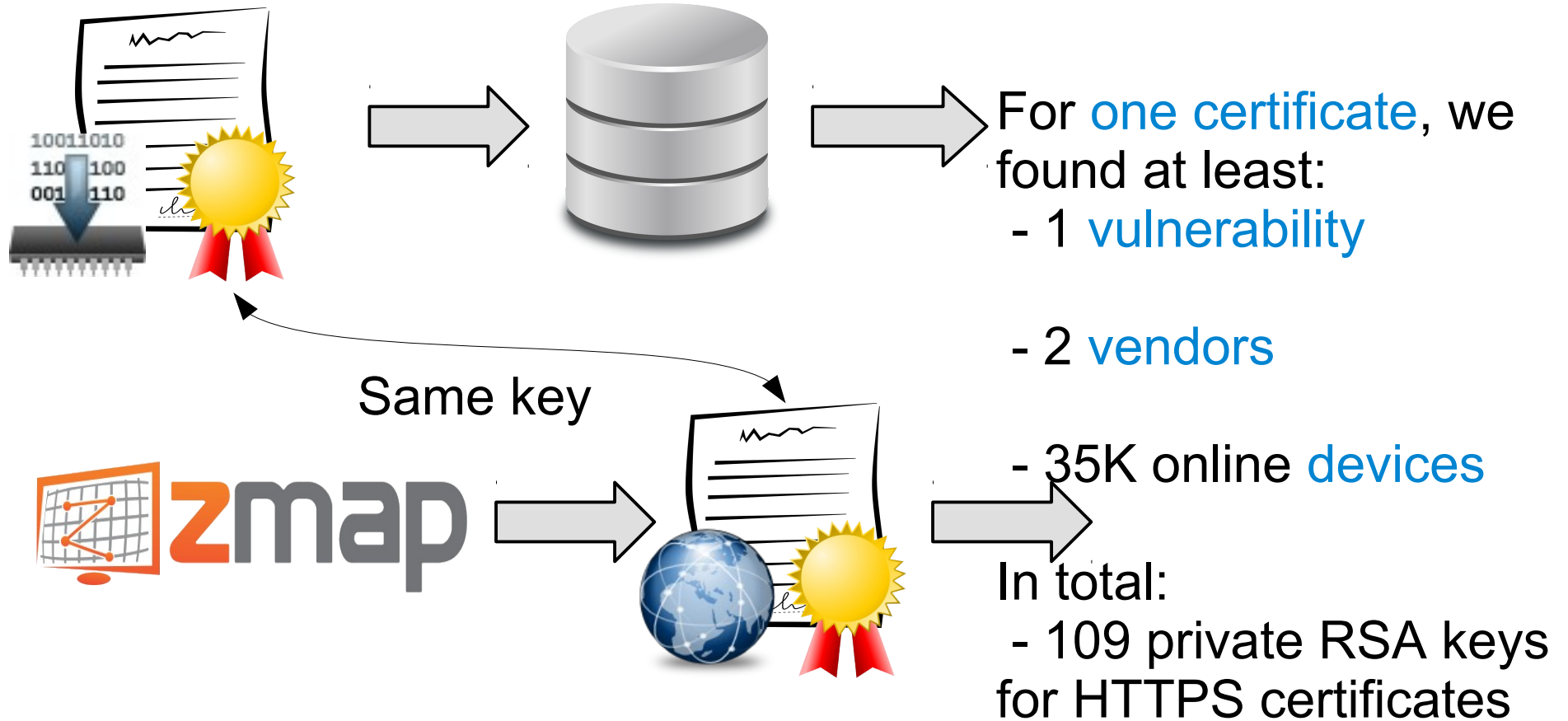
Example: Firmware HTTPS keys correlation



Example: Firmware HTTPS keys correlation



Example: Firmware HTTPS keys correlation



Results

Static Analysis

- 38 new **vulnerabilities**
- 693 **firmware images** with at least one vulnerability
- 140K **online devices** correlated to some vulnerabilities

Results

Dynamic Analysis

- High-severity **vulnerability impact**
 - Command injection, XSS, CSRF
 - **Automated+scalable** static and dynamic analysis
 - **185 firmware** images (~10% of original)
 - **13 vendors** (~25% of original)
- Total **alerts** from the tools
 - 6068 dynamic analysis alerts on 58 firmware images
 - 9046 static analysis alerts on 145 firmware images
 - Manual triage and confirmation is challenging

Results

Impact on Industry Actors

- **1 big player** in SCADA/ICS/embedded
 - In "**Top 100**" of "Fortune Global 500" (2015)
- **3 years** R&D contract (from 2015)
- **Using our frameworks**
 - Firmware collection, unpacking, analysis
 - Dynamic analysis and symbolic execution

Conclusions

- Plenty of **latent vulnerabilities** in embedded firmware
- Firmware security analysis is **absolutely necessary**
- Involves many **untrivial steps and challenges**
- A **broader view** on firmwares is not just beneficial, but necessary

Conclusions

- Security
 - Tradeoff with both cost and time-to-market
 - Clearly not a priority for **some vendors**

Conclusions

- *"Bug bounty programs can also provide an incentive to third-party researchers. Known vulnerabilities must be patched."*
- However, we faced:
 - Legal threats
 - Do not correctly fix the issue
 - Require endless follow-ups
 - Keep delaying the fix and disclosure

Conclusions

- *”A policy for vulnerability handling and disclosure awareness should be defined”*
 - Responsibly disclosing vulnerabilities is hard
 - Too few vendors have security@vendor.com
 - Does not scale for many reports (hundreds, thousands)

Summary

- We build-up research expertise and implement our expertise in working prototypes
- **First framework for automated large scale security analysis** and classification of firmwares and embedded devices
 - Simple and **advanced analysis** using dynamic and static
 - Quick identification of **(un)known vulnerabilities**
 - **Automated** classification and fingerprinting

References

- www.firmware.re
- www.s3.eurecom.fr/~costin/



The End

Thank You!
Questions?

{name.surname}@eurecom.fr

{name}@firmware.re