PROVE & RUN

THREE VIEWS ON IOT SECURITY

And a few more

Eric VETILLARD

77, avenue Niel, 75017 Paris, France

eric.vetillard@provenrun.com

JAVA CARD Forum

ORACLE

Moving from Java Card research to Java Card products to Java Card Consultant/Evaluator to Java Card Product Manager at Oracle.
And then, switch to IoT Cloud Service, then IoT at Prove & Run

Pitching IoT Security

Cool !

SECURE?

Safe !

2015 Hack

Poor decision	Safety reasoning	Security reasoning
Using the same keys	Simple process No complex infrastructure	Keys need to be **diversified** A key needs to be broken on every car
No systematic encryption	Only critical messages are encrypted	A secure channel protects against **reverse engineering**
Configuration data no tamper-proof	Configuration data integrity is protected by a checksum	Configuration data **authenticity** is protected by a cryptographic checksum
The vehicle ID is in error messages	Simplify diagnosis by having the data	A **remote attacker** doesn't have the ID, so let's protect it
Using DES	Well-known, fast algorithm	**DES is broken**, let's mandate AES
No protection against replay attacks	Same message, same action	A **recorded message** cannot have the same effect when replayed

http://m.heise.de/ct/artikel/Beemer-Open-Thyself-Security-vulnerabilities-in-BMW-s-ConnectedDrive-2540957.html

"The probability of unauthorized access to the OneTouch Ping system is extremely low"

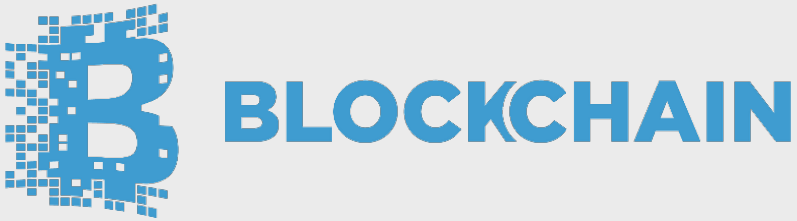
http://www.reuters.com/article/us-johnson-johnson-cyber-insulin-pumps-e-idUSKCN12411L

Consultants

Consultants **listen** to their customers
Consultants need a good **story line**
Consultants need palatable **results**

Checklists

Consultants are a **reflection** of their customers
Consultants **improve** their customer's thinking
But this process may go **out of hand** …



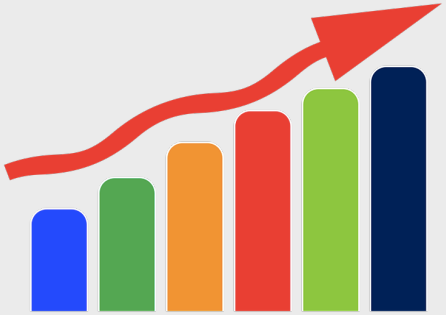
Quantum-safe Cryptography

Important in the future, but what priority **today?**

Secure Elements
Unbreakable, no mention of integration

Device Security
Solves your security problems instantly. Also unbreakable.

Analytics
Catch all attacks. No mention of configuration



Scary attacks
Imminent chaos

My Solution
Instant bliss

Some rationale
Serious stuff

Tough job for every decision maker

Clueless IoT developers
and safety enthusiasts

Check-list consultants
and science-fiction promoters

Enthusiastic security vendors
1000 perfect solutions

Clueless IoT developers
and safety enthusiasts

Check-list consultants
and science-fiction promoters

Enthusiastic security vendors
1000 perfect solutions

Government

Users

CHALLENGE

Mix this!

Invasive Government

Tired users

Clueless IoT developers

Check-list consultants

Enthusiastic security vendors

Cultural Shift

Functional Security

Security for makers?

Security is not cool

Mirai & friends …

Agile is bad for security

IoT Security Frameworks

industrial internet CONSORTIUM

OTA Online Trust Alliance

CSA cloud security alliance®

GSMA

Avoiding regulation through self-assessment

OWASP
The Open Web Application Security Project

AIOTI

Some Pressure Required

Users?

Not ready for pressure

May happen at any time

Government

Getting closer to regulation

What good will it do?

Investors

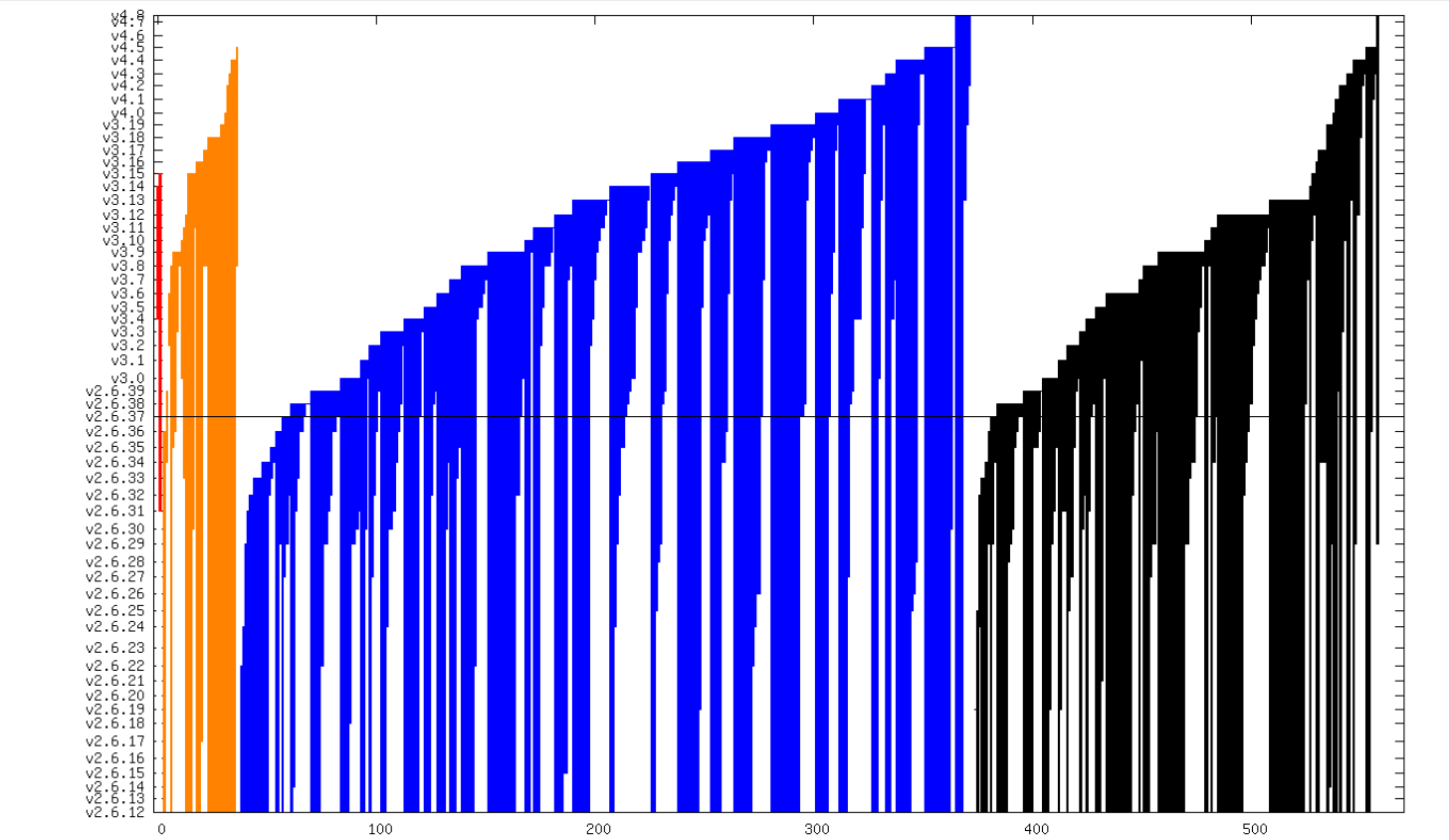
Great pressure on startups

If security is a liability

This is all about economics and incentives.

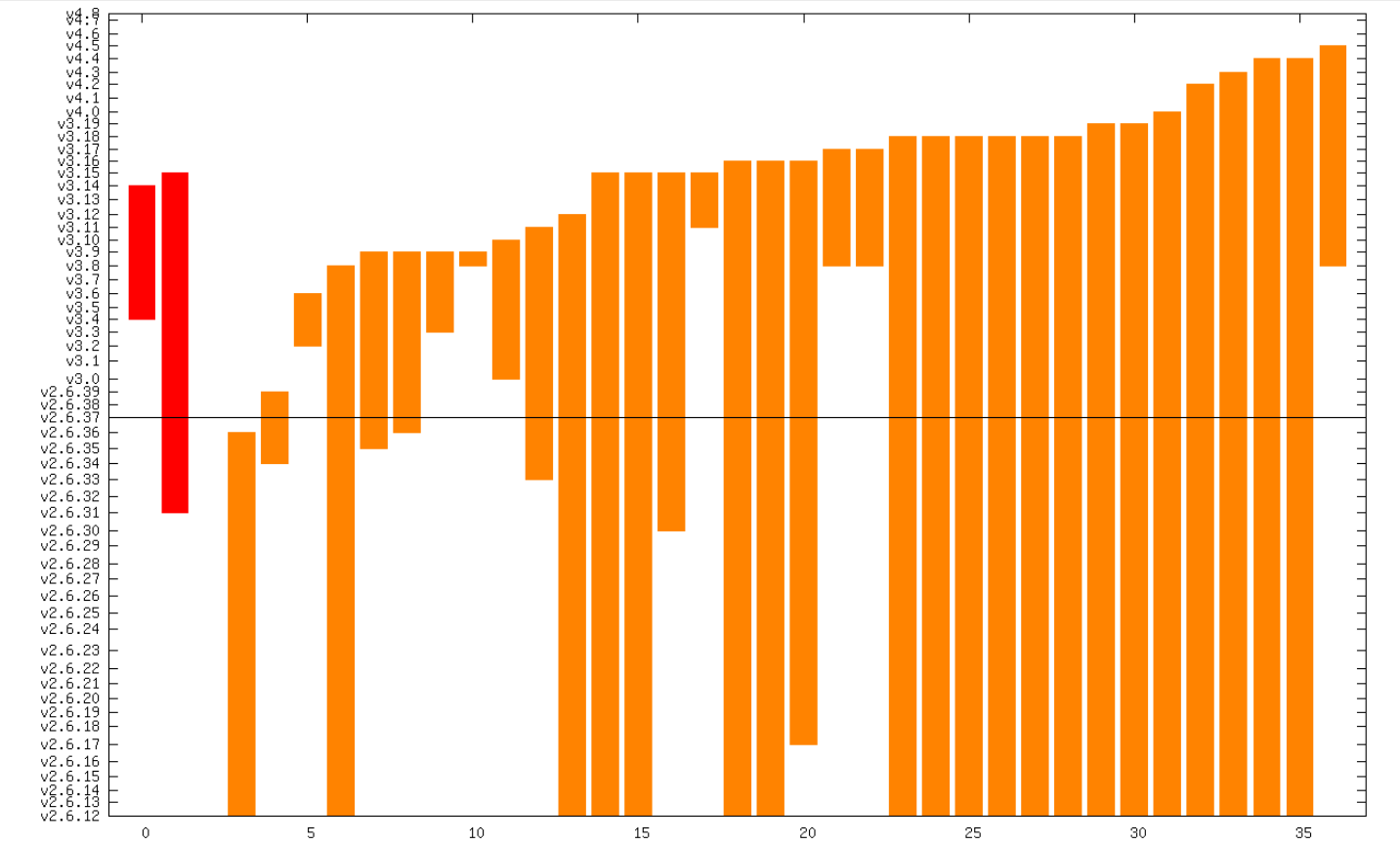
Resilience

Vulnerabilities are here to stay



Average time between identification and fix: 5 years

https://outflux.net/blog/



Average fix time is even 6.4 years for “High” level

https://outflux.net/blog/

A race condition was found in the way the Linux kernel's memory subsystem handled the copy-on-write (COW) breakage of private read-only memory mappings. An unprivileged local user could use this flaw to gain write access to otherwise read-only memory mappings and thus increase their privileges on the system.

https://bugzilla.redhat.com/show_bug.cgi?id=1384344#

9 year old low-priority bug

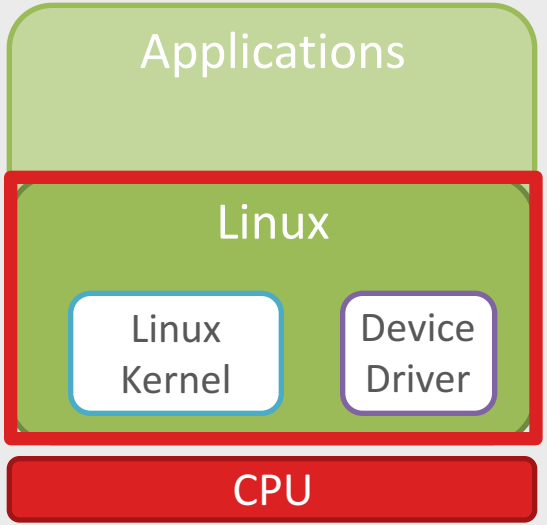
Exploited last month

https://dirtycow.ninja/

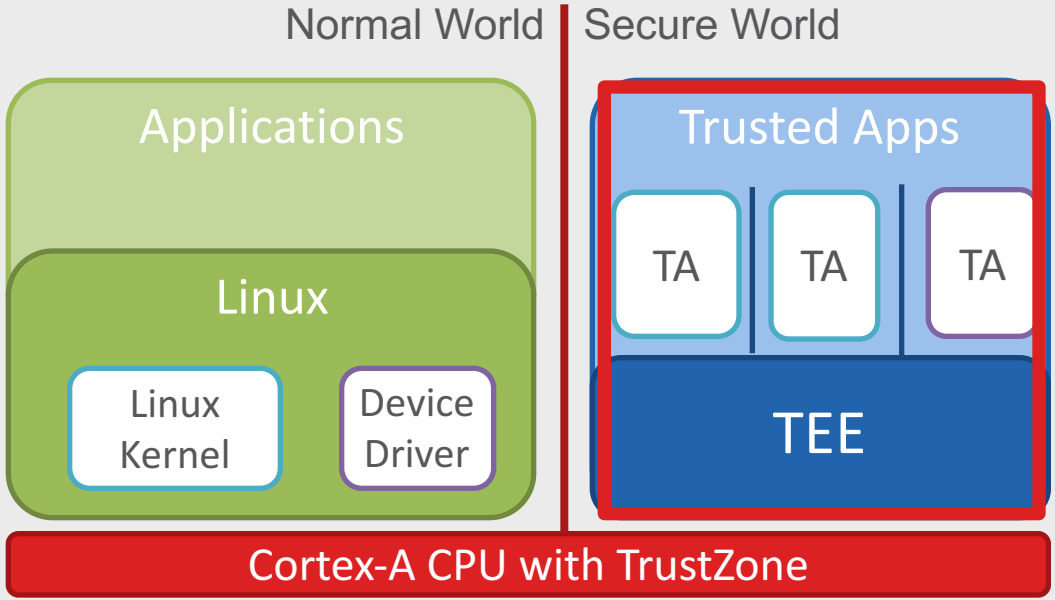
Linux Kernel: Unsafe at any clock speed

- **Linux is everywhere, and 0-days are far too common**
  - **All commits are watched by the best hacker teams**
  - **Most kernel issues are in third-party drivers**
  - **Patching is hard at best, impossible in many cases**

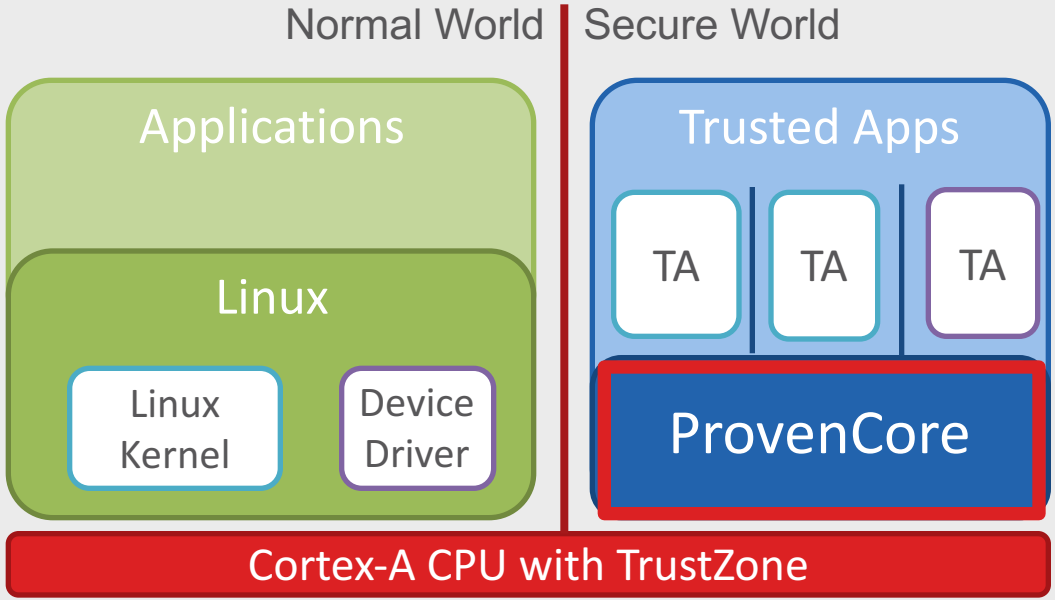
- **An initiative has been started**
  - **Public speech from Google's Kees Cook this summer**
  - **Rethinking Linux kernel security from the ground up**
    - Kernel self-protection, improved update
  - **Will take years**



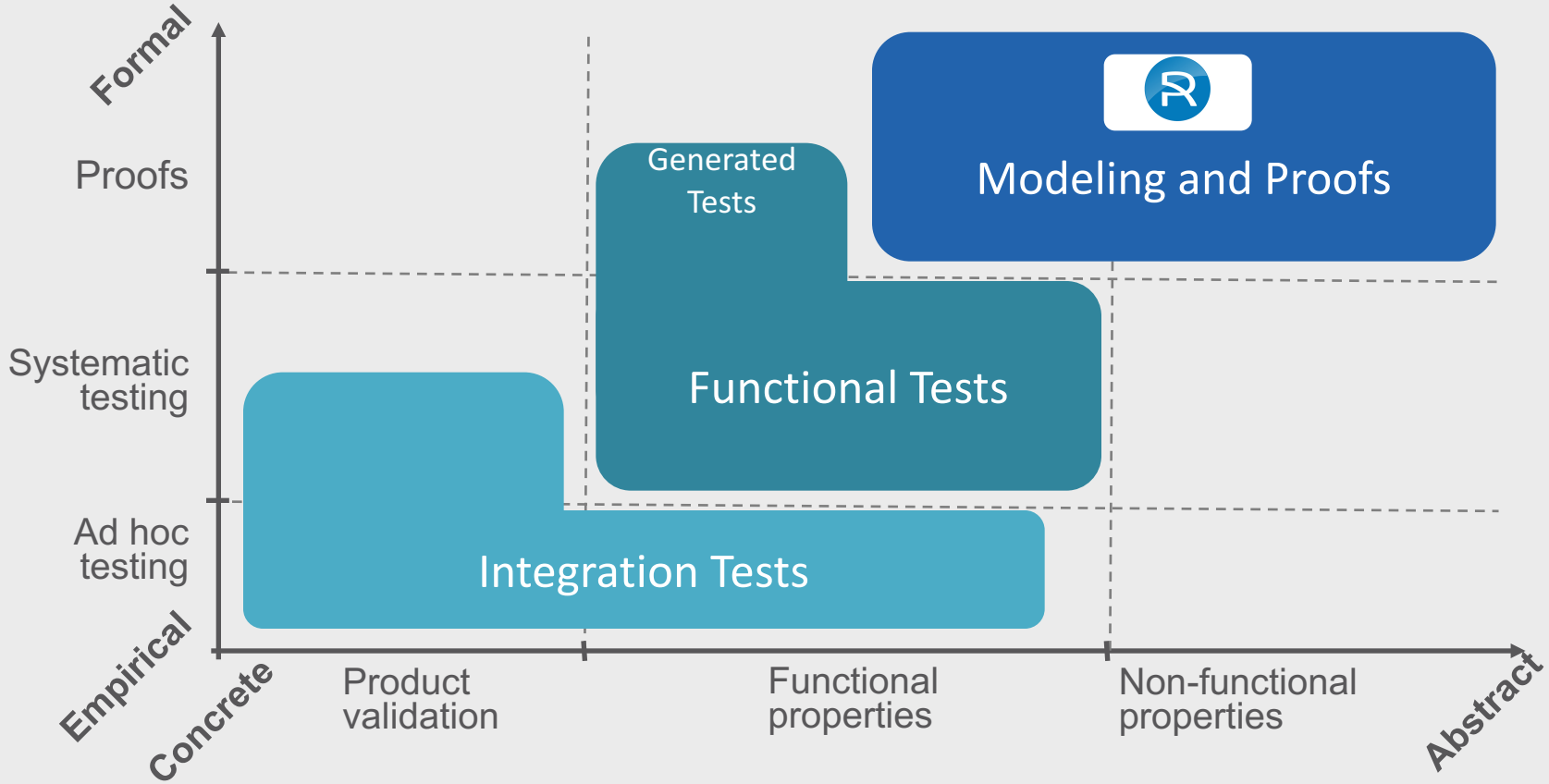
The TCB is too large, leaving too much room for attackers to get kernel privileges, *e.g.*, through third-party drivers

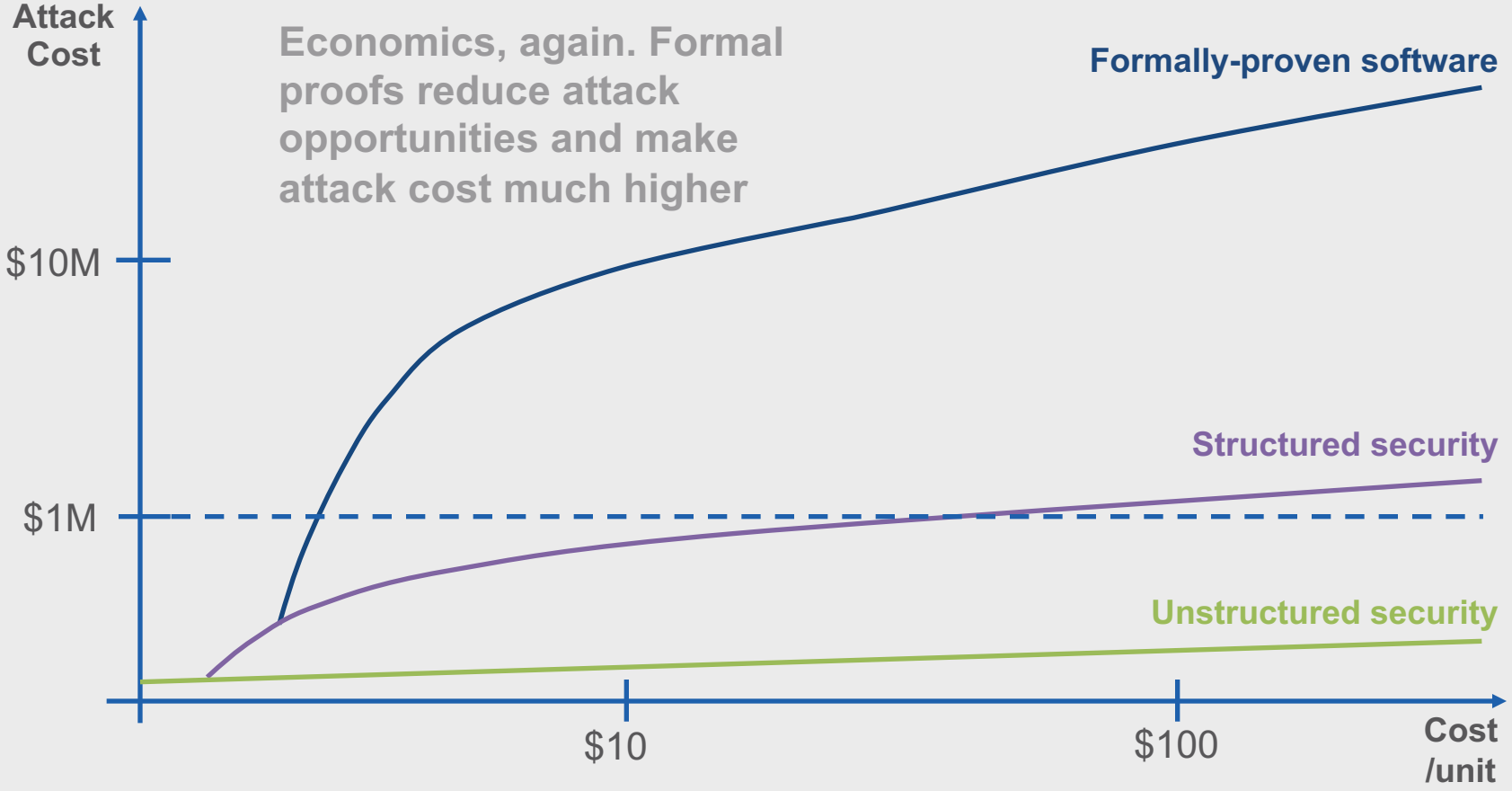


Critical assets are isolated, the TCB is smaller, but it is still too complex, and attackers manage to get in



ProvenCore pushes to the extreme. Secure Apps are not trusted by the kernel and they are isolated. The kernel is small enough to prove integrity and confidentiality properties.





Challenge #1: Better resilience

Make it **Strong**

Make it **Massive**

Make it **Real**

Vulnerabilities are here to stay

Bad devices are here to stay

Already deployed: Millions of devices
Device lifetime: Up to 20 years
Support duration: 2-5 years, 10 years max

Firmware update: No, or not secure
Startup lifetime: Under 5 years

Hardware security lifetime: Under 10 years

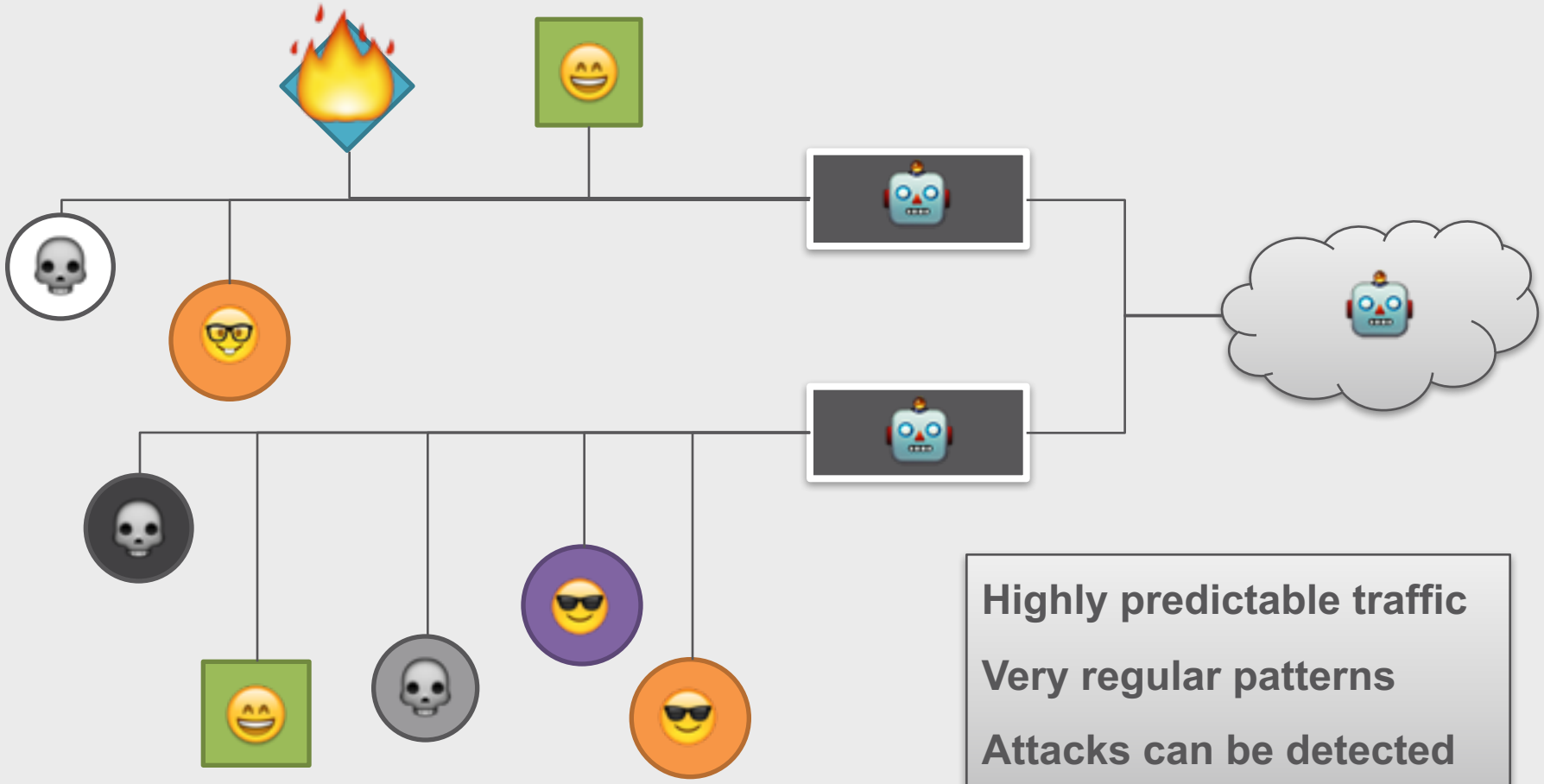
Cool > Secure

Challenge #2: Help IT get old

Engineering is bad at this

Reduce needed resources

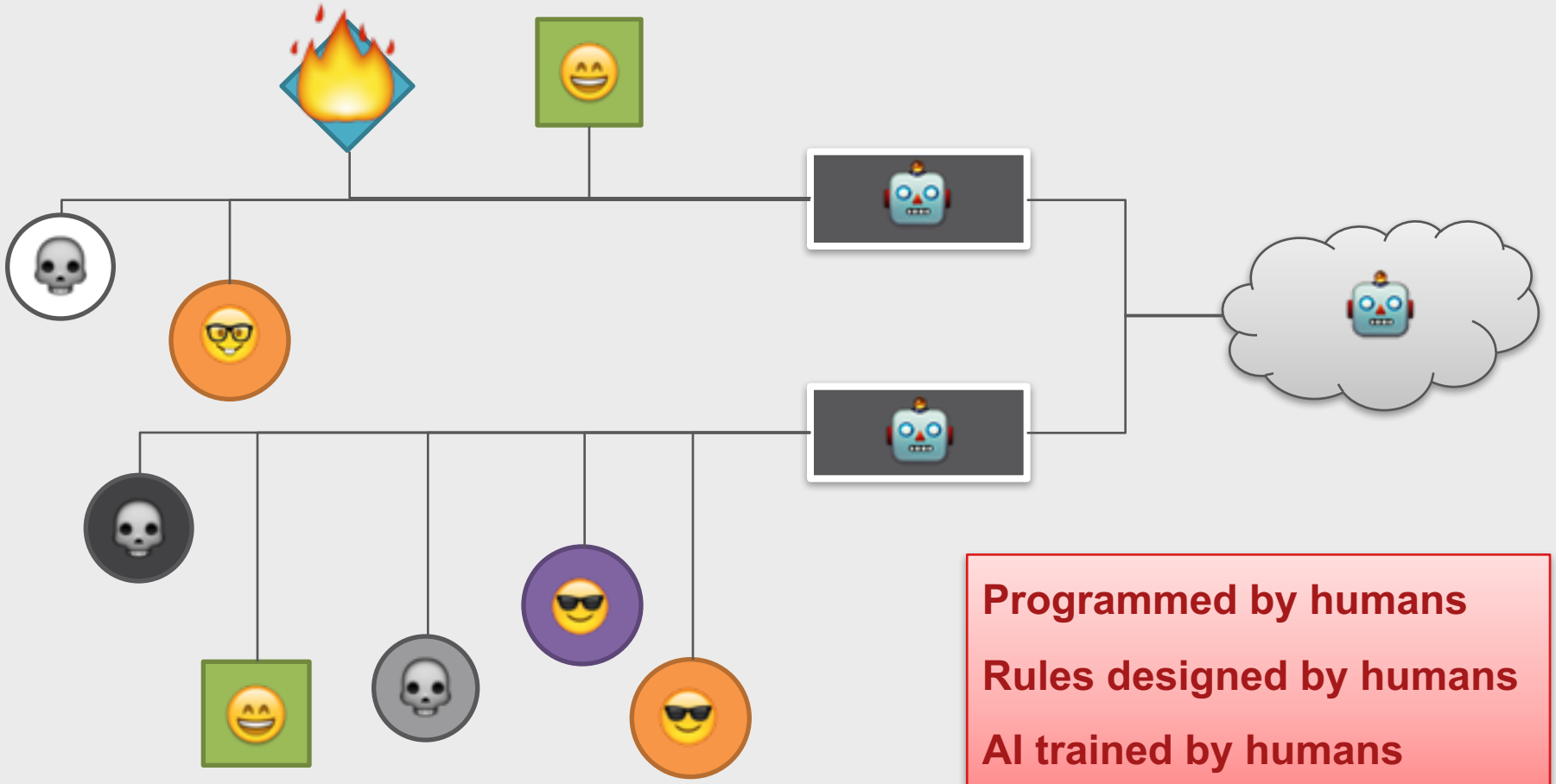
Security re-engineering



Highly predictable traffic

Very regular patterns

Attacks can be detected



Programmed by humans

Rules designed by humans

AI trained by humans

Challenge #3: Make AI fail

Avoid detection by AI

Make it catch real alerts

Help make it better

People

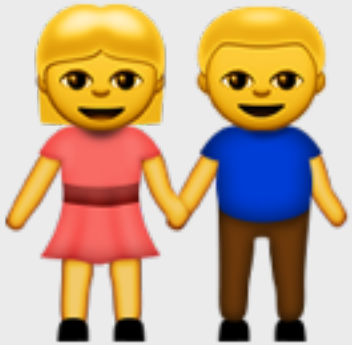
Vulnerabilities are here to stay

Bad devices are here to stay

Attacks will become really sophisticated

~~Highly sophisticated attack~~

~~Highly coordinated attack, probably nation-state~~



Current Attacks Are *NOT* Sophisticated

- The 2015 Jeep attack
  - 2 person.years of investigation
  - Mostly one-time reverse engineering
  - All attacks exploited very basic vulnerabilities

- The 2016 MIRAI attacks
  - Exploiting a magic combination
  - Extremely basic bugs in devices
  - Directly accessible targets, thanks to UPnP
  - Unsupervised devices, unlikely to be fixed

But Sophisticated Attacks Exist

- Powerful 0-day attacks hoarded by hacker groups
  - Can most likely take down any Linux implementation
  - Combined with attacks on maintenance servers
- Hardware-based attacks
  - Fault induction on secure boot
  - Even combined attacks, like Rowhammer attacks

Challenge #4: Attack harder

How do card attacks work?

More Rowhammer/Cache

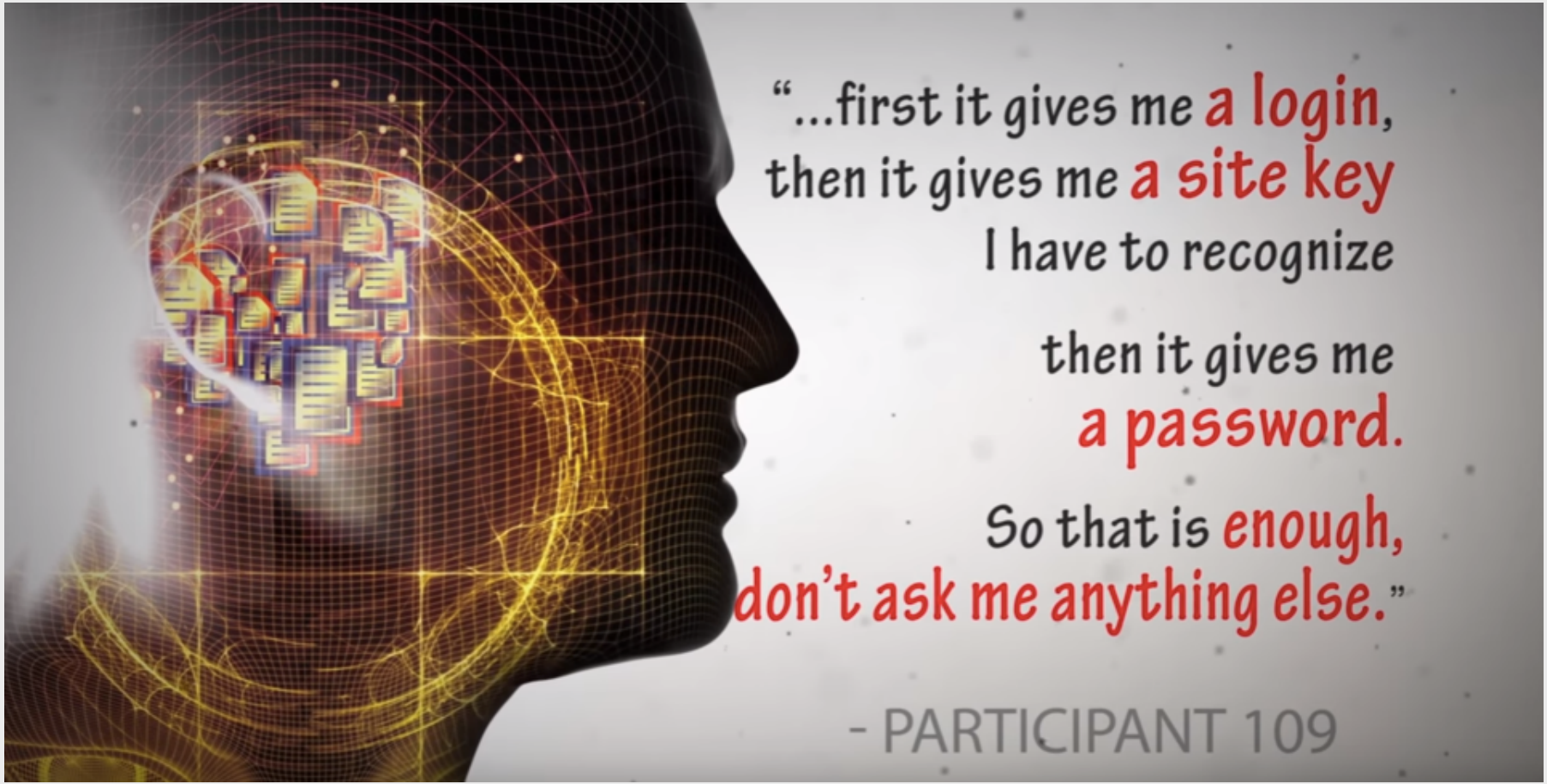
Attack the clouds

Vulnerabilities are here to stay

Bad devices are here to stay

Attacks will become really sophisticated

Users are clueless and will remain clueless



https://www.nist.gov/news-events/news/2016/10/security-fatigue-can-cause-computer-users-feel-hopeless-and-act-recklessly

ACCENTURE SURVEY

75% of security execs think they are good

70% state that cybersecurity is board-level

33% of attacks are missed by security teams

42% lack budget for hiring in security teams

https://www.accenture.com/t20161027T195446__w__/us-en/_acnmedia/PDF-35/Accenture-Building-Confidence-Facing-Cybersecurity-Conundrum-Transcript.pdf

MIRAI

Default root passwords can’t be changed

Default passwords are wrong

The very idea of root password is wrong

Selfie Pay sounds like a gadget at first, but ends up bringing real security advantages.

eGo sounds good, but it opens a whole new range of social engineering attacks

Security *vs.* Usability

Mobile applications are commonly used

And come with their own vulnerabilities

A way into home networks

A treasure for reverse engineers

https://www.virusbulletin.com/uploads/pdf/conference_slides/2016/Apvrille-vb-2016-mobileiot.pdf

Security *vs.* Usability

With IoT, no trade-off is possible

No unreasonable expectations from users
Security in all components of the IoT solution

Usability is an essential part of security

Usability should be certified

Challenge #5: Usable security

New models for devices

Radically new UI?

Certifiable usability criteria

Privacy?

Great research work

For instance, work on *Differential Privacy*

Not ready for prime time

AI / Data science are threats

Thank you!

Have a Great Conference

eric.vetillard@provenrun.com

javacard.vetilles.com