TEE for 2FA

Trusted Execution Environments for Two-Factor Authentication: Comparing Approaches

Erik Poll

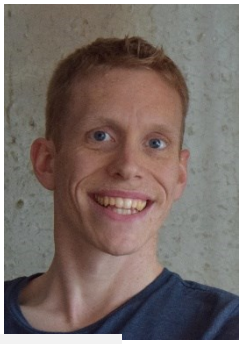
Digital Security group, Radboud University

based on joint work with

Roland van Rijswijk-Deij

Digital Security @ Radboud University

Some of our people & research topics

Joan Daemen
Bart Mennink
Cryptography


Peter Schwabe
Simona Samardjiska
Post-quantum crypto

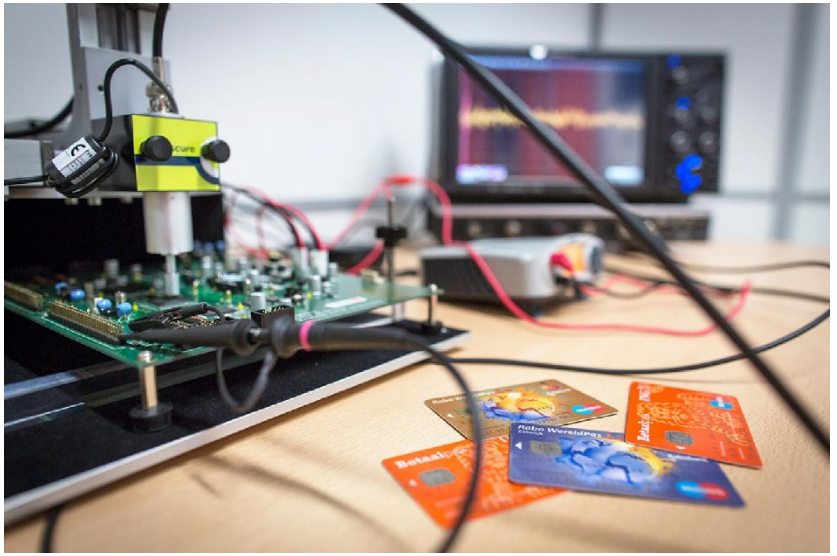

Lejla Batina
Ileana Buhan
Hardware Security


Bart Jacobs & Eric Verheul
Identity management & authentication

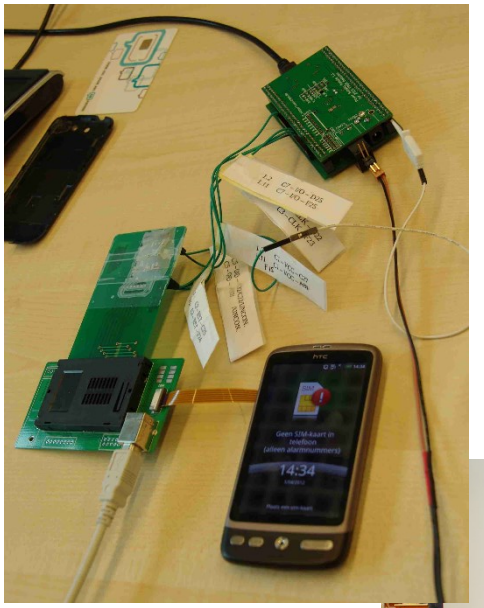


Jaap-Henk Hoepman
Mireille Hildebrandt
Frederik Zuiderwijn Borgesius
Privacy

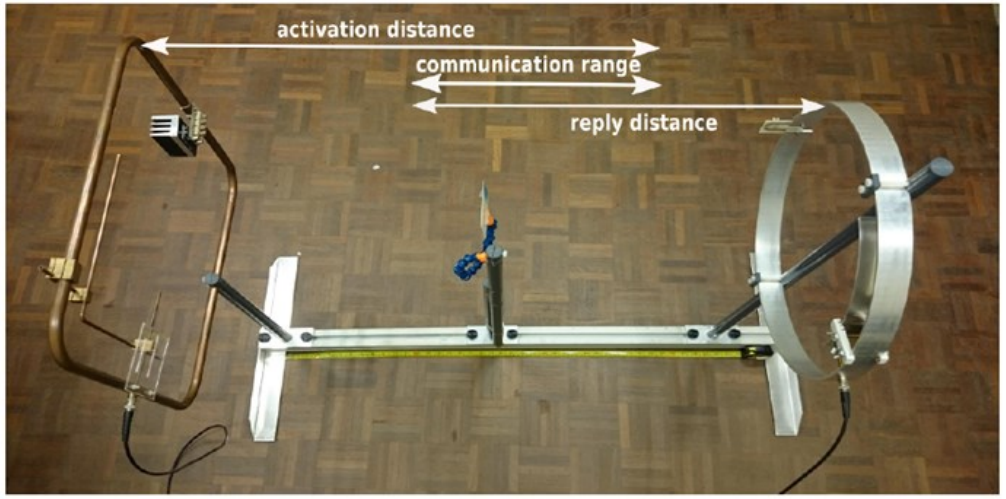
Some of our toys



side channel analysis



communication interception & modification



hi-power RFID antenna

‘Strong’ Authentication Solutions

Critical component for security, often hardware-based

1. *What are the security guarantees?*

2. *How strong are these guarantees?*

   - What are the trust assumptions? (aka the TCB)
   - How hard is it to break them?

Successful security solution

- Key storage
- Key usage

Successful security solution

- 'Trusted' I/O to the user

Getting rid of this special hardware?

TEEs (Trusted Execution Environments) try to offer similar security functionality in standard computers

- **Eg. TPM, Intel TXT, Intel IPT, Intel SGX, ARM TrustZone, Apple Secure Enclave, Android Secure Storage, Samsung KNOX, …**
- **NB a variety of solutions (HOW)**

  offering a variety of features (WHAT)

Pros & Cons of TEEs?

Pros

- No separate hardware needed ☺

Cons

- No separate piece of hardware ☹
  - ➢ more complexity, more heterogeinity
  - ➢ harder to understand the security guarantees

    for end users & for IT professionals
  - ➢ complicated business & licensing models

Pro *and* Con

- It is (nearly) always online

Security features of TEEs

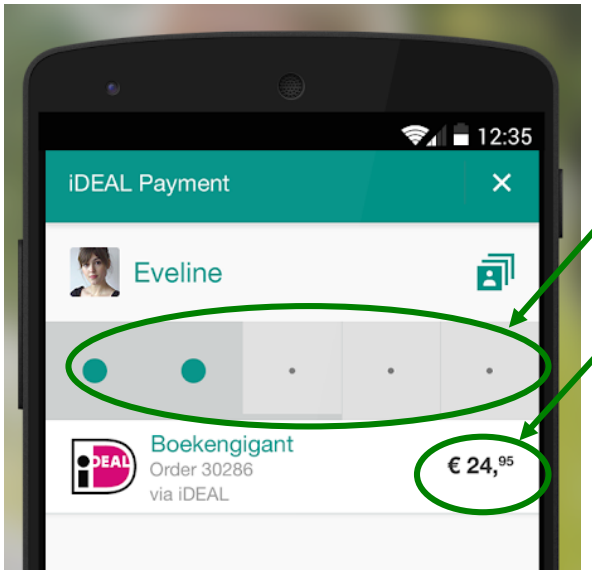
- Key storage 
- Key usage 

but:

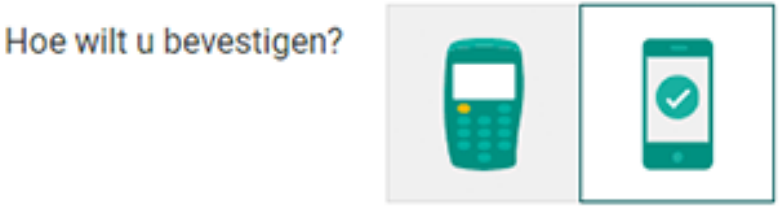
- **are keys bound to the device or to a specific app?**
- **are these TEE mechanisms too clumsy, too expensive, or more secure than necessary for NFC payment apps?**

Security features of TEEs?

- Trusted I/O?



In other words, can we get these equally secure?



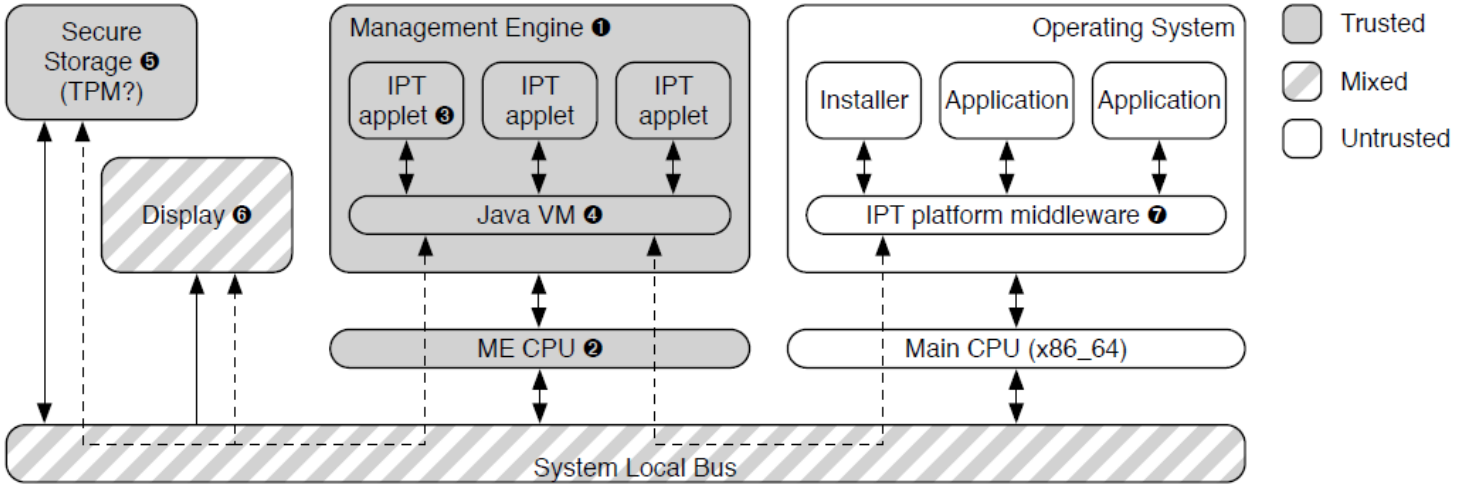
Intel IPT (RIP) & ARM TrustZone

- Two very different TEE solutions
  - IPT uses a separate chip,
    ARM Trustzone a special mode on the same chip
- Some security functionality in common
  - Isolated Execution
  - Secure Storage
  - Remote Attestation
  - Secure Provisioning

[Roland van Rijswijk-Deij and Erik Poll, *Using Trusted Execution Environments in Two-Factor Authentication: comparing approaches*, Open Identity Summit 2013]

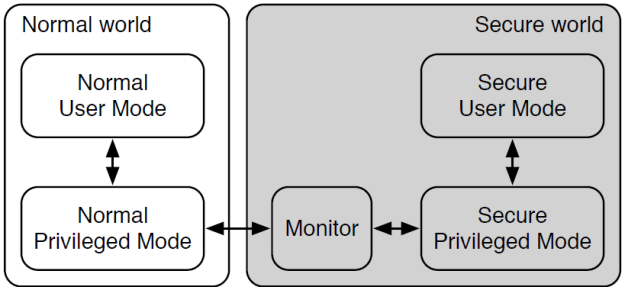
Trusted I/O ?

- For IPT, can the separate (trusted) chip control input & output?



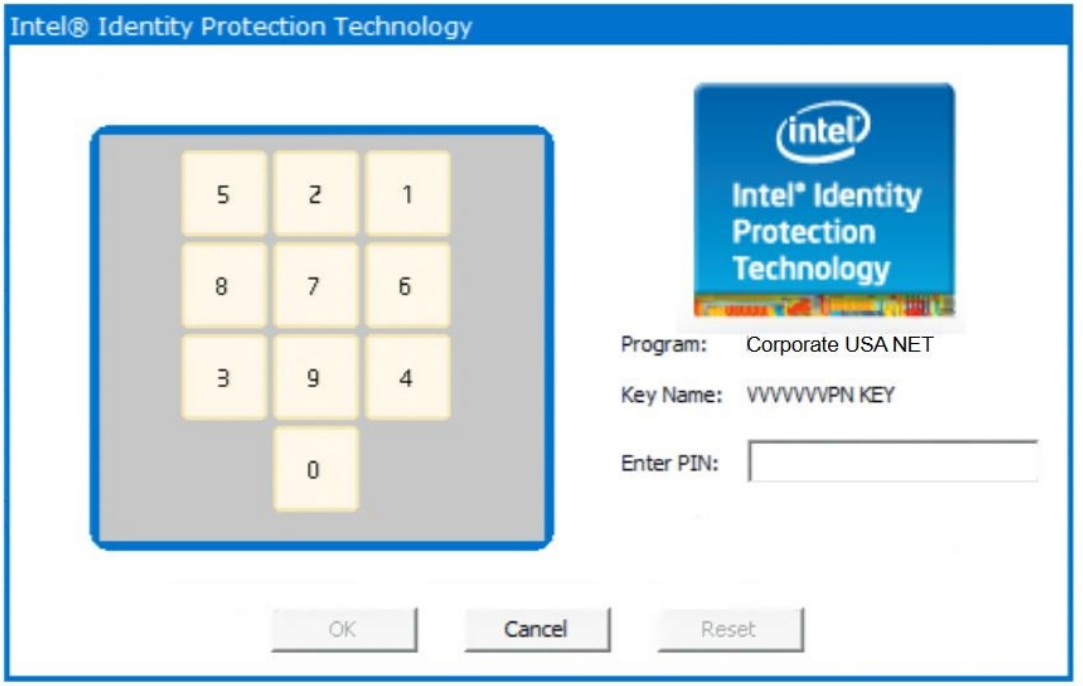
  - Only secure output, input can be intercepted

- For TrustZone, can the secure (trusted) world control input & output?



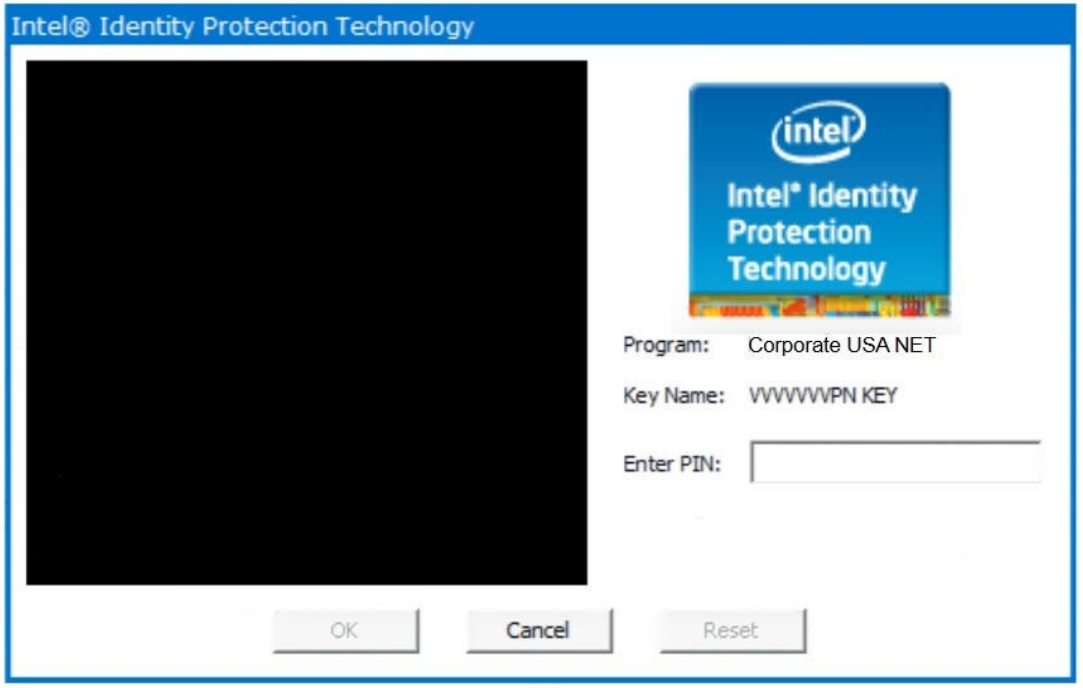
  - Yes, but only for *some* configurations & software stacks

Secure PIN entry on IPT



Note: Keyboard must be scrambled to prevent interception of the PIN

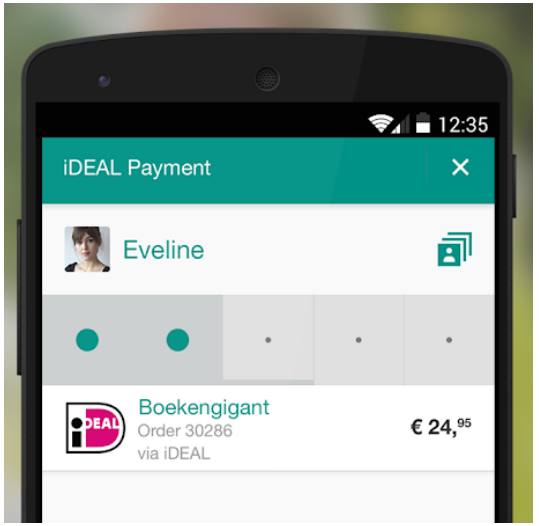
Secure PIN entry on IPT



What malware or a compromised OS on the same device would see

Remaining problem

- ***How can the user know who they are talking to?***



  - **Eg: am I giving my PIN code to ABNAmro or some malware?**

- **Fundamental limitation of both IPT and TrustZone.**
  And any TEE-based solution?

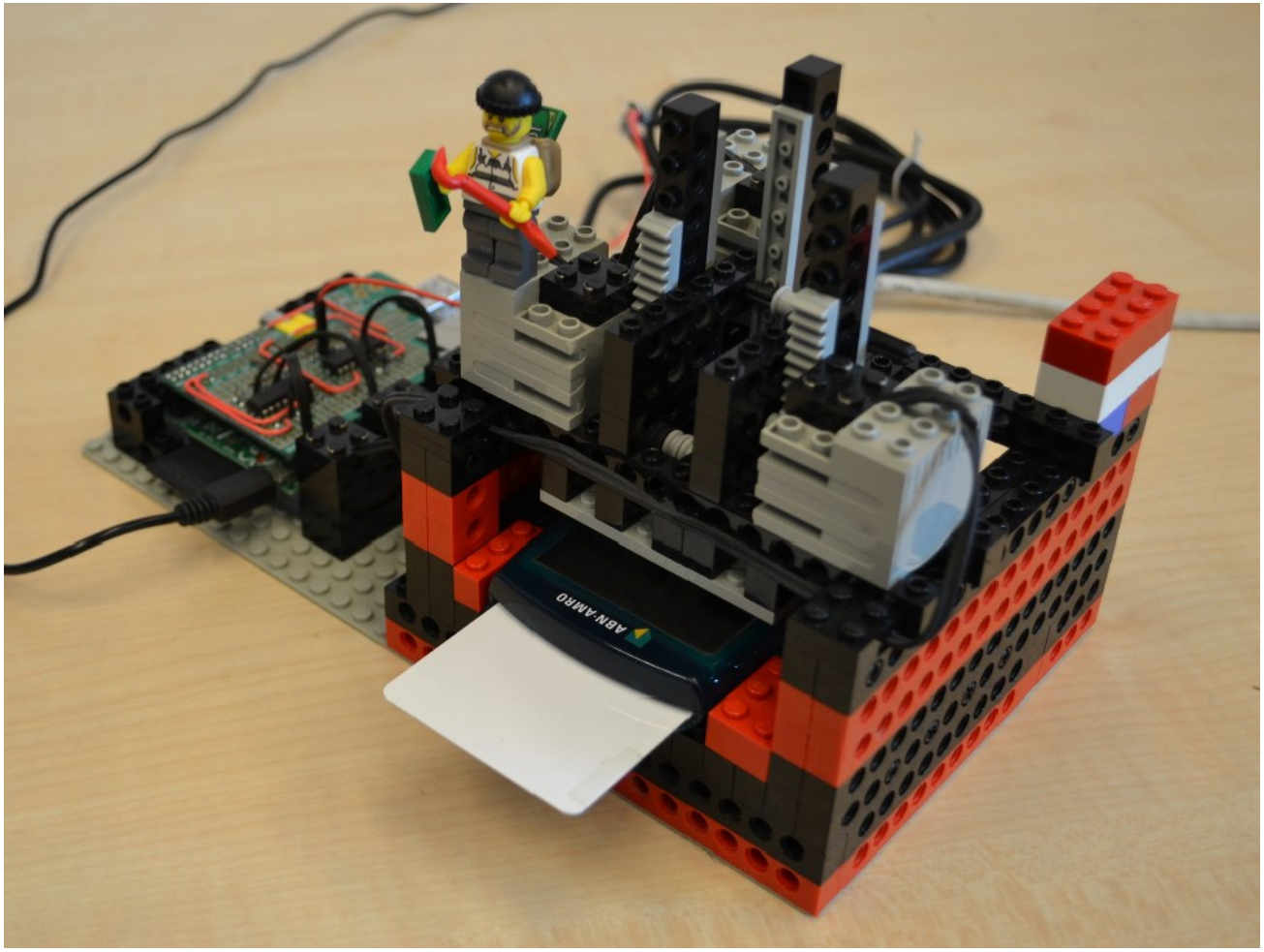
Conclusions

- TEEs are interesting but complex
- Some TEEs can provide trusted I/O
  - but: how does user know who they are interacting with?
- Is this still two factor?
  - Mapping TEE solutions to eIDAS levels is not so obvious

- Beware: claims or terminology involving the word 'trust' or 'trusted' are often bullshit or misleading, so be very suspicious!

[Roland van Rijswijk-Deij and Erik Poll, *Using Trusted Execution Environments in Two-Factor Authentication: comparing approaches*, Open Identity Summit 2013]

Thanks for your attention!



https://tinyurl.com/legolearning

[Automated Reverse Engineering using LEGO, WOOT 2014]