Tamper Resistant Smartcards - Attacks and Countermeasures (Slides)

Eric Auer

July 21, 2000

What we will look at

1. *Introduction:*
   - 1.1 Why to use smartcards
   - 1.2 What kinds of problems may arise

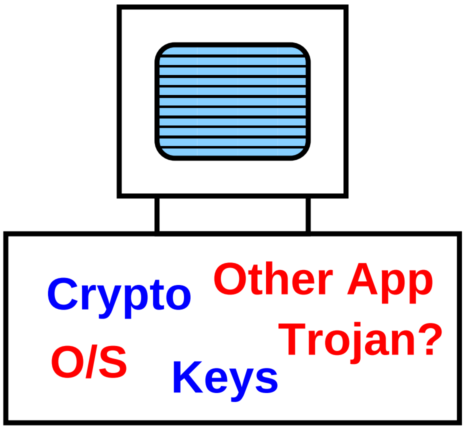
2. *Non-invasive attacks:*
   - 2.1 Eavesdropping with a DSP
   - 2.2 Fault generation, glitch attacks

3. *Invasive attacks:*
   - 3.1 Reverse engineering
   - 3.2 High-tech wiretapping
   - 3.3 Silicon modifications

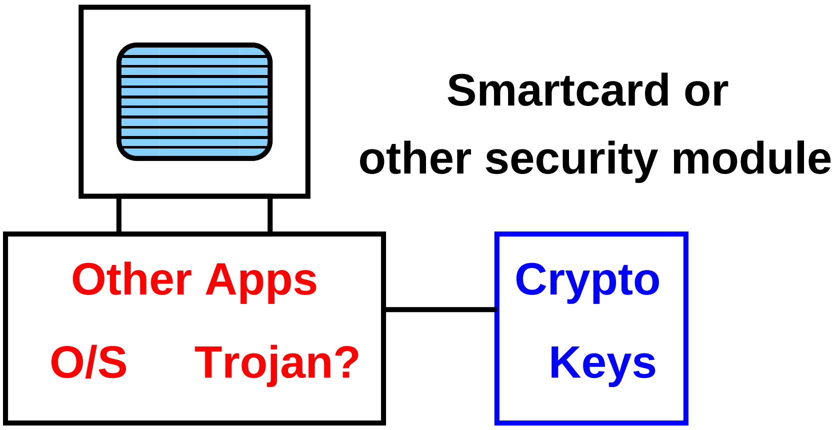
4. *Conclusion:*
   - Security can be improved at low effort but only up to a certain level

1.1 Why security modules are used

They make protection of keys easier:



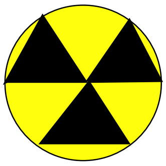
Keys stored inside complex system:
Many ways for spoofing and eavesdropping



Keys only accessed by module:
Secret data is isolated from complex system

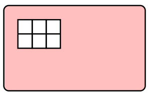
1.2 How to protect your device

Safelocks in nuclear weapons:



- Buried deep into the warhead
- Protected by armed guards
- Internal battery for tamper sensors
- Sensors trigger blast for self destruction

Smartcards:



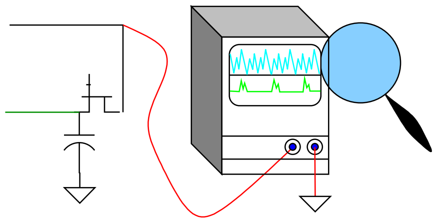
- Are easily stolen, no massive packaging
- Hostile hardware hidden in terminal?
- Cannot self destruct (card/keys) while off
- Removing power disables tamper sensors

But:

Some tamper resistance *is* possible for smartcards without much effort.

2.1 Non-invasive: Eavesdropping

How it is done:



- State changes in gates, SRAM cells and bus drivers cause current peaks
- Assign findings to internal state – timing analysis is most useful
- Eavesdrop CPU flags to reveal key bits, test results or even algorithms
- Knowing the chip design eases attacks

Countermeasures:

- Make reverse engineering harder
- Use proprietary design
- Modify design to avoid current peaks
- Use autonomous internal clock
- Insert random idle cycles

2.2 Non-invasive: Fault generation

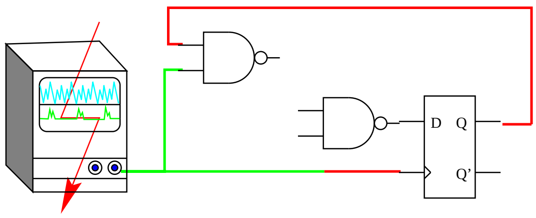
Simple attacks (e.g. reduce V_{CC}):

- Reduce entropy in random numbers
- Make simple tamper detectors fail
- Keep older phonecards and pay-TV keys from updating account data

Basics of glitch attacks:

- Flipflops compare their input to V_{CC} during a short time after the clock edge
- Target few flipflops with a glitch at the right time (exploit delays)
- Shorter clock cycles keep part of the logic from updating its state

2.2b Non-invasive: Glitch attacks



Goals:

– Have output loops give away more data

– Skip or weaken encryption loops

– Skip conditional jumps in security checks

Countermeasures:

– Use segmented pointers to limit damage

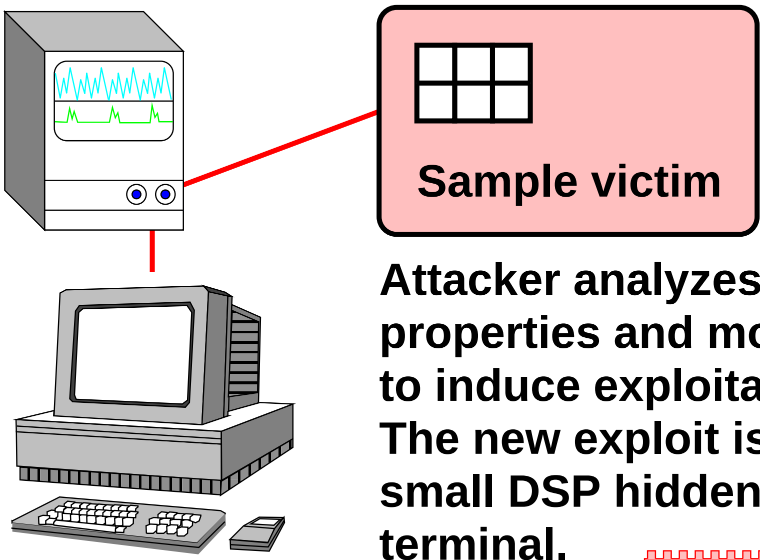
– Trigger self destruction on wraparound

– Handle tamper alarms at hardwired

– Double-check data to cope with faults

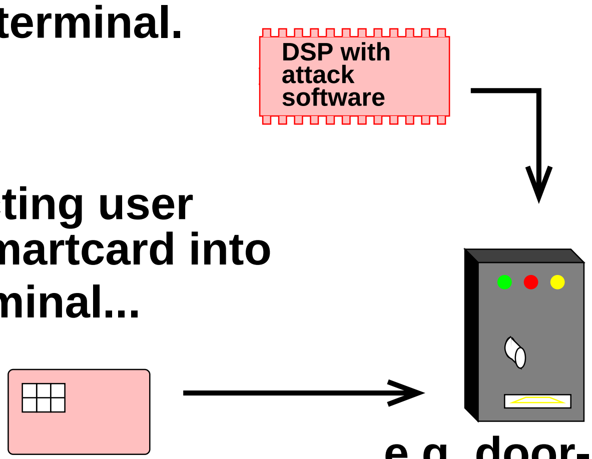
2. Non-invasive attacks

Hard to develop but easy to clone



PC with DSP board

Attacker analyzes exact signal properties and modifies signals to induce exploitable errors. The new exploit is uploaded to a small DSP hidden in a regular terminal.



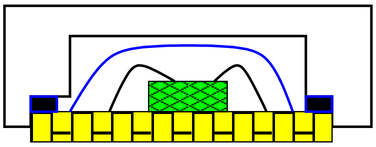
Now an unsuspecting user inserts another smartcard into the subverted terminal...

e.g. door-lock, EC terminal, ...

Another possible attack:

Crack a batch of pay-TV keycards right away using the test equipment (no hiding needed).

3. Invasive attacks: First steps

Getting access: 

– Cut out the chip module or melt the glue

– Dissolve epoxy resin with nitric acid (>98% HNO_3)

– Scratch holes in remaining Si oxide passivation layer or use a laser cutter ($ 100.000) or hydrogen flouride dry etching

– Microprobing needles held by micropositioners establish connection (MP workstation: $ 10.000 - $ 100.000)

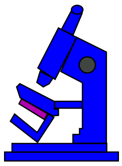
– Expensive equipment can be rented

Countermeasures:

– Special coatings that cannot be removed without damaging the chip

– Tamper sensor meshes as top layer (more useful with internal battery)

3.1 Invasive: Reverse engineering

How it works: 

– Image processing reconstructs gate network or spots standard cells

– Using a confocal microscope gives height information and reduces blurs

– Professional attackers make the implant mask visible or remove top layers (e.g. to analyze ROM)

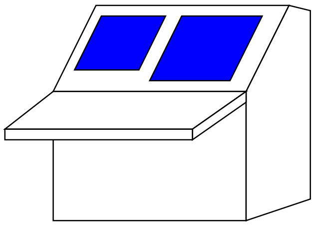
Countermeasures:

– Fusible link ROM is hard to analyze

– Use fusible links to allow design updates

– Crackers may be tricked by faked features

3.2 Invasive: More eavesdropping

Known methods: 

– Use microprobing needles and a DSP

– Lithium niobate crystals uncover signals

– Electron Beam Testers can do the same

– Low clock rate simplifies eavesdropping

Countermeasures:

– Protective meshes, polished top layers and “conformeal glues” strain attackers

– Use on chip tamper resistant clock source

– Add hidden idle cycles on a random basis

– Require operational low clock frequency sensor for boot up sequence

3.3 Focused Ion Beam editing

FIB editing:

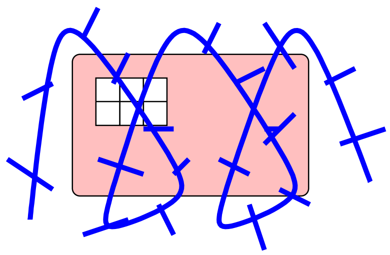
- A 5nm beam is used to drill holes
- Deposit platinum or insulators
- Laser interferometers allow $\pm$ 150nm positioning
- tap or modify circuits
- A FIB ws. is $ 500.000, rental is easy

Countermeasures:

- Destroy test circuits radically
- Use meshes and polished top layers as obstacles
- Avoid single points of failure
- Use heavy segmented memory model

4. Conclusion: Topics

Important topics:



- Tamper evidence (lets you block keys after evident attacks)
- Tamper resistance stops some attacks
- Cope with a few cracked cards
- Personalization (e.g. of software) helps to verify the identity of a card
- Public key or challenge-response protocols avoid giving away secrets early

4b Conclusion: Suggestions

Keeping most hackers out is affordable

- Polished top layer and protective meshes protect against low cost tools
- Destroy test circuity after use
- Filter out compromising signals
- Use a randomized on chip clock generator
- Filter out unusual / out of range signals
- Using a proprietary design and special sealing is worth while for some cards

But do not trust smartcards too much

- Active sensors need own power supply
- Self destruction is almost impossible