Smartcards

Erik Poll
SoS group
University of Nijmegen
Overview

• What is a smartcard?
• Why use smartcards?
• What are the possibilities and limitations of smartcards?
• Attacks on smartcards
What is a smartcard?
What is a smartcard?

• Tamper-resistant computer, embedded in piece of plastic, with limited resources

• capable of securely
  - storing information
  - processing information

(This is what makes a smartcard smart; stupid cards can store but not process data)
Smartcard contacts

External power supply and external clock
Many modern smartcards are now contactless
Smartcard hardware

- **CPU** – 8 to 32 bits
- **memory**
  - RAM
  - ROM (for some program code)
  - EEPROM/Flash/... (“hard disk”, for code and data)

Modern cards may have 1K RAM, 16K ROM, 64K EEPROM

- **limited I/O: just a serial port**
- **possibly:** crypto co-processor, random number generator
Smartcard software

- Smartcard contains very simple operating system, capable of executing programs

- Programs can be written in
  - proprietary machine code language, or
  - higher level language, notably Java Card

Most new SIMs are now Java Cards.
Smartcard example uses

- bank card, chipknip
- GSM SIM in mobile phone
- pay TV
- public transport (OV smartcard)
- passport
- student/employee cards to control access buildings, computer networks, ...
Stupid cards

• Stupid cards only store information (securely or insecurely), and can’t process it.

• Examples of stupid cards:
  - magnetic-stripe cards
  - some old chipcards are not really smart, because the chip only provides a (passcode protected) file system. (Eg chipcard formerly used in pay phones)
Why use smartcards?
What are the possibilities and limitations of smartcards?
Example use of smartcards

• How does **electronic purse (chipknip)** work?
• How does **GSM SIM** work?

• Where and how do
  - confidentiality
  - integrity
  - authentication
  - non-repudiation
play a role in these applications?
CIA and smartcards

• Confidentiality:
  - of data (crypto keys) on card
• Integrity:
  - of data and program code
• Authentication:
  - because (data on) card cannot be copied
• Non-repudiation
  - because (data on) card cannot be copied
  - also logging on the smartcard (and integrity of this log)
Typical use of smartcard

- Private key $K$ never leaves the card
- Card issuer does not have to trust the network, the terminal, or card holder
Example: logging on over a network

- Send password unencrypted over net (eg. rlogin)  
  Trust network, terminal, user
- Send password encrypted over net (eg. slogin)  
  Trust terminal, user
- Idem, but user, not terminal, does encryption  
  Trust user
- Using smartcard  
  Trust no-one, except the smartcard  
  (NB smartcard is controlled by card issuer, not card holder!)
NB the problem with cryptography

Any use of crypto introduces problems:

1. key distribution
   • how do we generate & distribute keys?
2. key storage
   • where can we safely store keys?
3. en/decryption
   • who do we trust to perform en/decryption?

Smartcards can offer a solution
Smartcard vs mag-stripe cards

• Smartcard cannot easily be copied or altered, unlike a mag-stripe card
• Copying mag-stripe cards - **skimming** - is big criminal business, as copying cards, and observing PIN codes, is easy...
Skimming
Skimming
Skimming
Skimming
Example: checking PIN codes

• How can an ATM check PIN codes
  - for mag-stripe card?
  - for smartcard?
Example: internet banking

• Some internet banking systems use smartcard reader with display
• Why not use smartcard reader in my PC?
  - better to have simple reader rather than complicated PC not part of TCB
  - possibility of a malicious code on PC
  - typing PIN code on PC not acceptable
TCB and smartcards

• Smartcard typically part of the TCB (Trusted Computing Base), ie. the trusted part of the system

• NB “trusted” is a negative quality: it means “you have to trust it” not “you can trust it”

  - If any part of the TCB fails, security is broken
  - TCB should be as small and reliable as possible
Example: digital signatures

• (How) could smartcard be used to generate digital signatures?
• What are problems?

• Smartcard limitations (from perspective of card holder):
  - smartcard does not have a (trusted) display
  - smartcard does not have a (trusted) keyboard
  - no way to check if/what the smartcard signs
Attacks on smartcards
Smartcard are not 100% secure

• Growing range of **attacks** (and associated countermeasures) is known

• Crucial question: is the risk acceptable?
  - are the costs of an attack larger than the potential financial gain for the attacker?

• Threats depend on application
  - eg. cloning more interesting for PayTV than GSM SIMs
Smartcard attacks

- attack **confidentiality**: eg. get access to keys stored on card, to clone cards
- attack **integrity**: change data stored on card or change behaviour of card

- Confidentiality and integrity not just important for crypto keys or PIN codes, but also for software on the card and logic implemented in the hardware of the card
Logical attacks

• Find and exploit software bug, using the normal communication channel, eg.
  - hidden commands (eg for initialisation)
  - buffer overflows, eg to read past end of file
  - try to abuse file access privileges
  - exploit weakness in crypto-protocol
  - malicious applet on multi-application smartcard

• No equipment needed, but change of success low
Countermeasures against logical attacks

• Write quality software
• Testing
• Formal verification
• Perform code reviews to spot software problems
• Improve OS, APIs, programming languages to make software bugs less likely
• Open research area!!
Side-channel attacks

- Side-channel = any other channel than the normal I/O channel that may be observed
- Possible side-channels:
  - power consumption
  - timing
  - electro magnetic radiation
  - ....
- A side-channel might leak information, or be manipulated...
Power consumption of a smartcard
This is probably a DES encryption!
Differential Power Analysis (DPA)

Deduce information from power consumption

Countermeasures against DPA

• in software
  - careful coding of crypto-algorithms
  - redundancy in data representation

• in hardware
  - add clock jitter or other noise
  - dual rail logic
Power glitching

• precisely timed dip in power supply to induce fault, eg
  - prevent an EEPROM write
    • eg to PIN counter
  - read memory contents as zero
    • eg of crypto-key
  - Some crypto-algorithms may be attacked using such fault injections (DFA-Differential Fault Analysis)
Active side-channel attacks

• **Other side channels:**
  - clock frequency
  - temperature/heat
  - light or X-rays
  - EM radiation

• **Countermeasures:**
  - **hardware:** sensors to detect changes in voltage, etc.
  - **software:** double-checking results of computations
Physical (or invasive) attacks

- reverse engineer and tamper with the physical chip

- first step: getting access to chip’s surface
  - remove chip from the smartcard
  - use chemical to remove expoxy resin and the top metal/silicon layers of the chip
Removing chip from smartcard
Etched smartcard with chip exposed
Tools for physical attacks

• **Microscope**
  - optical or scanning electron microscope (SEM)

• **Focused Ion Beam (FIB)**
  - not only observe, but also make changes: removing or adding wires, insulators,...

• **Probe station**
  - to probe wires on the chip
Probing

Sub micron probe station
Easy to localise Bloc

Crypto

Logic

EEPROM

RAM

ROM
Probing

Probing with eight needles
Probing

• **Observe data on the chip in operation**
• **Typically:** tap **data on bus**
  - by putting needle on bus wires
• **Probing can be done using**
  - physical needles (>0.35 micron) or
  - electron beam
Using Focused Ion Beam in probing

Fibbing can be used to
• add probe pads for lines too thin or fragile for needles
• surface buried lines
Countermeasures against probing

• use smaller circuitry
• protective layers or sensors on chip surface
• multiple layers on chip
  – with sensitive data on deeply buried wires
• scramble or encrypt bus
  – attacker then has to reverse engineer the scrambling logic
• use glue logic instead of (easy to spot) bus
Multiple layers on chip

The same gate before and after etching to remove top layer
Using Focused Ion Beam (fibbing)

- All chips contain circuitry to check chip after production
- After testing, test logic is disabled by blowing a fuse
- FIB can restore test logic
ROM memory content extraction

Staining of ion implant ROM array
ROM memory content extraction

- ROM contents can be observed
- usually no crypto material in ROM, but knowledge about code stored in ROM can help with other attacks

- Countermeasure: encrypt ROM
  - attacker now has to reverse engineer the encryption logic
RAM voltage contrast SEM
RAM memory content extraction

• Scanning electron microscope can be used to observe RAM contents

• Countermeasure: scramble or encrypt RAM

• Content of EEPROM or Flash is harder to extract
Smartcards attacks - future

• Ongoing arms race between smartcard manufacturers and attackers
• Physical attacks becoming harder, due to improved countermeasures and smaller circuitry
• But increasing complexity of software on smartcard may introduce new logical attacks
Smartcard attacks - conclusions

• Smartcards is not tamper-proof, as witnessed by
  - logical attacks
  - side-channel attacks: DPA, glitching
  - physical attacks

• Smartcards are tamper-resistant and tamper-evident, to a degree