OFFSEC

www.offsec.ir

Smart Cards & Security

Ebrahim Ghasemi
ebrahim@offsec.ir

[OFFSEC www.offsec.ir WhoAmI]

Ebrahim Ghasemi

Days of experience in **Smart Card programming**

Days of experience in **Network Traffic Analysis**

Days of experience in **Computer Security and Cryptography**

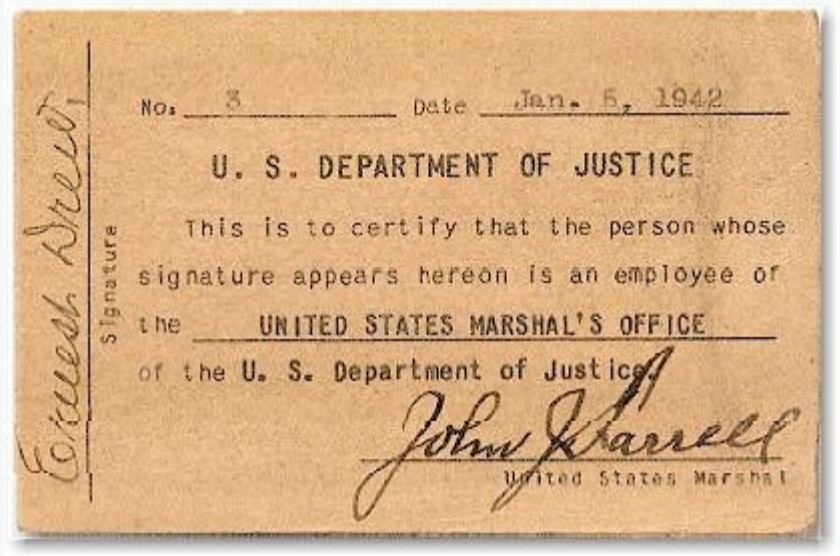
Contents

Smart Card Security

- Electronic Cards Evolution
- Smart Cards & Javacards
- Common Attacks

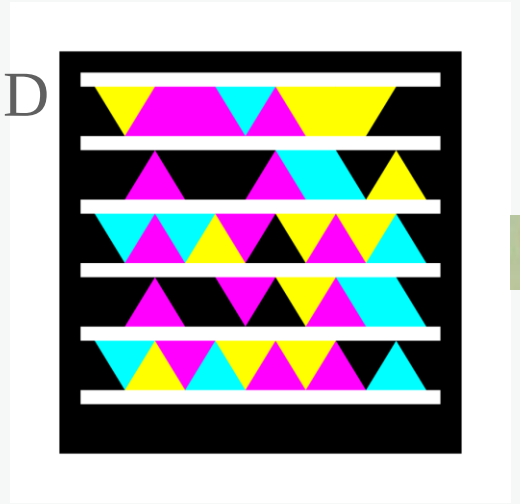
-[Electronic Cards Evolution]- Papers

- Literally Papers (Neanderthal!)
- Embossed Cards
- Holograms



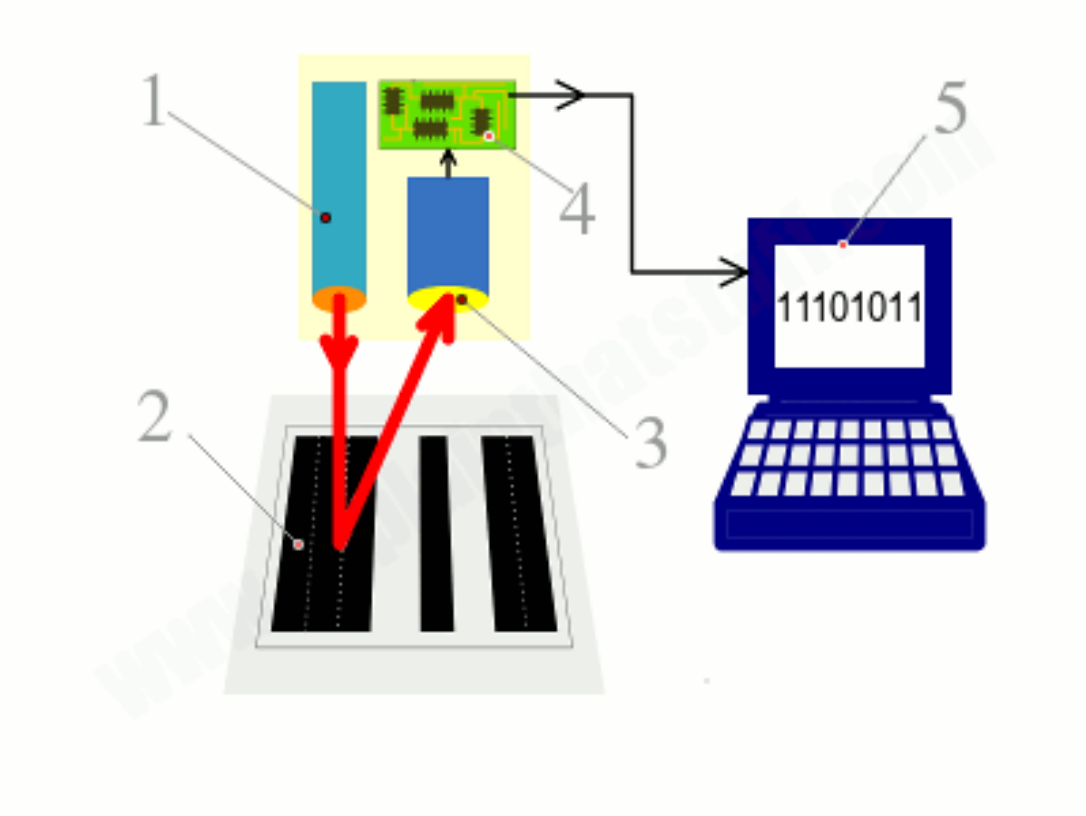
Barcode Cards – Introduction

- Types
  - Linear / 1D
  - Matrix or Square / 2D
- Standard Formats
  - EAN13 & EAN-8
  - UPC-A & UPC-E
  - Code128
  - ITF-14
  - …
- Pros: Cheap and Easy to User
- Cons: Low Data Density & Ease of Forgery



Barcode Cards – How Does it Works?

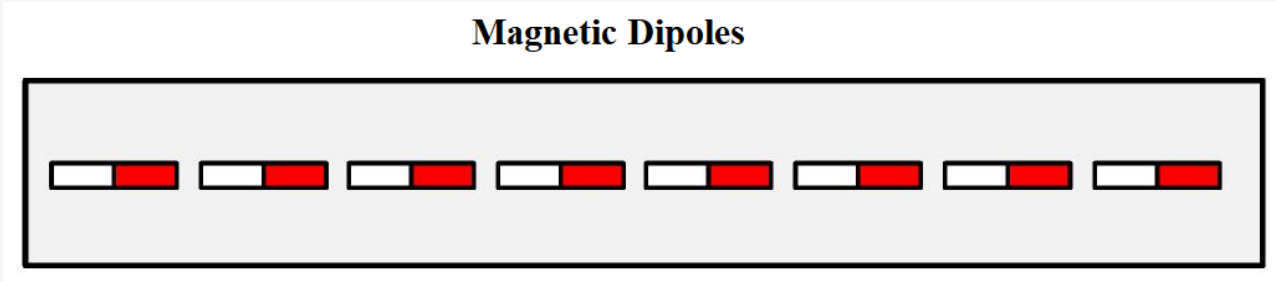
Past :: Light Reflection



Now :: Image Processing

Magnetic Cards – Introduction

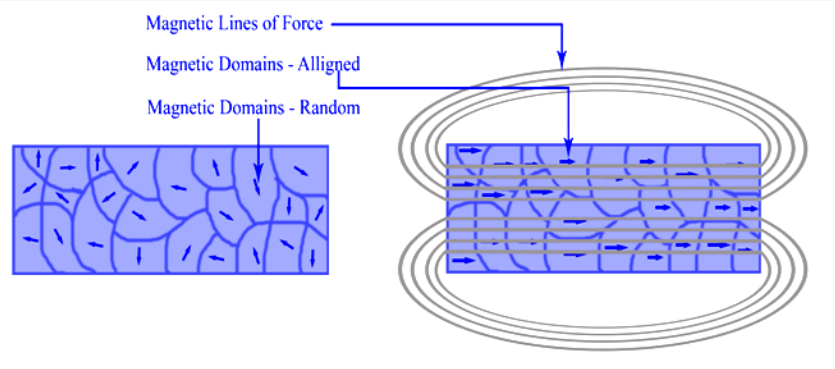
- AKA
  - Magstripe
  - Swipe Card
- Pros: Cheap and Easy to User
- Cons: Low Data Density & Ease of Forgery

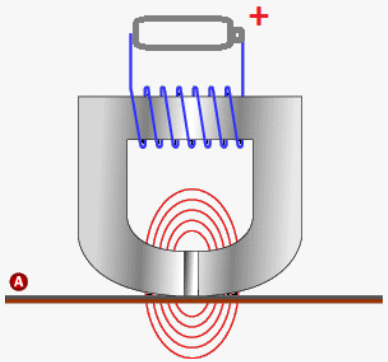


Magnetic Cards – How Does it Works?

:: Write ::

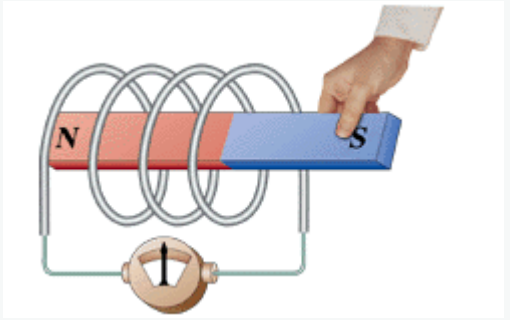
Magnetic Dipoles & Electromagnetic Fields

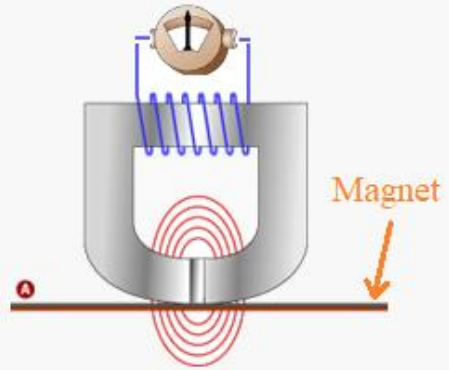




:: Read ::

Electromagnetic Induction

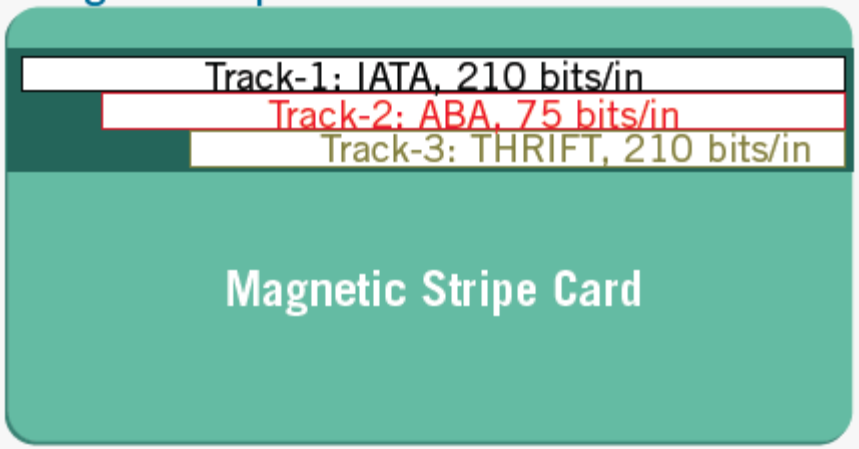




Magnetic Cards – Usage

- Hardware
  - Encoder → Writer
  - Decoder → Reader
- Low Data Density → Multiple Tracks (Rows)
- ISO/IEC 7813
  - 3 Tracks of Data
    - International Air Transportation Association (IATA)
    - American Bankers Association (ABA)
  - Banking Cards Use Track #2 and #1 (Sometimes)

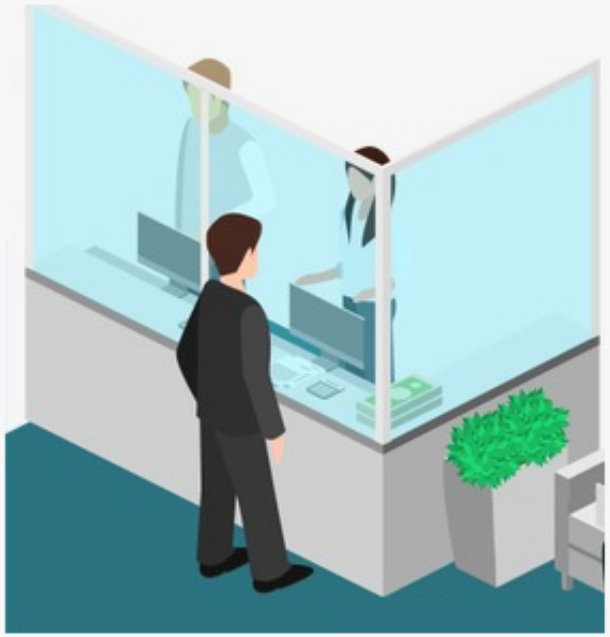
A magnetic stripe card.



Magnetic Cards – Banking Cards

Banking Cards Database

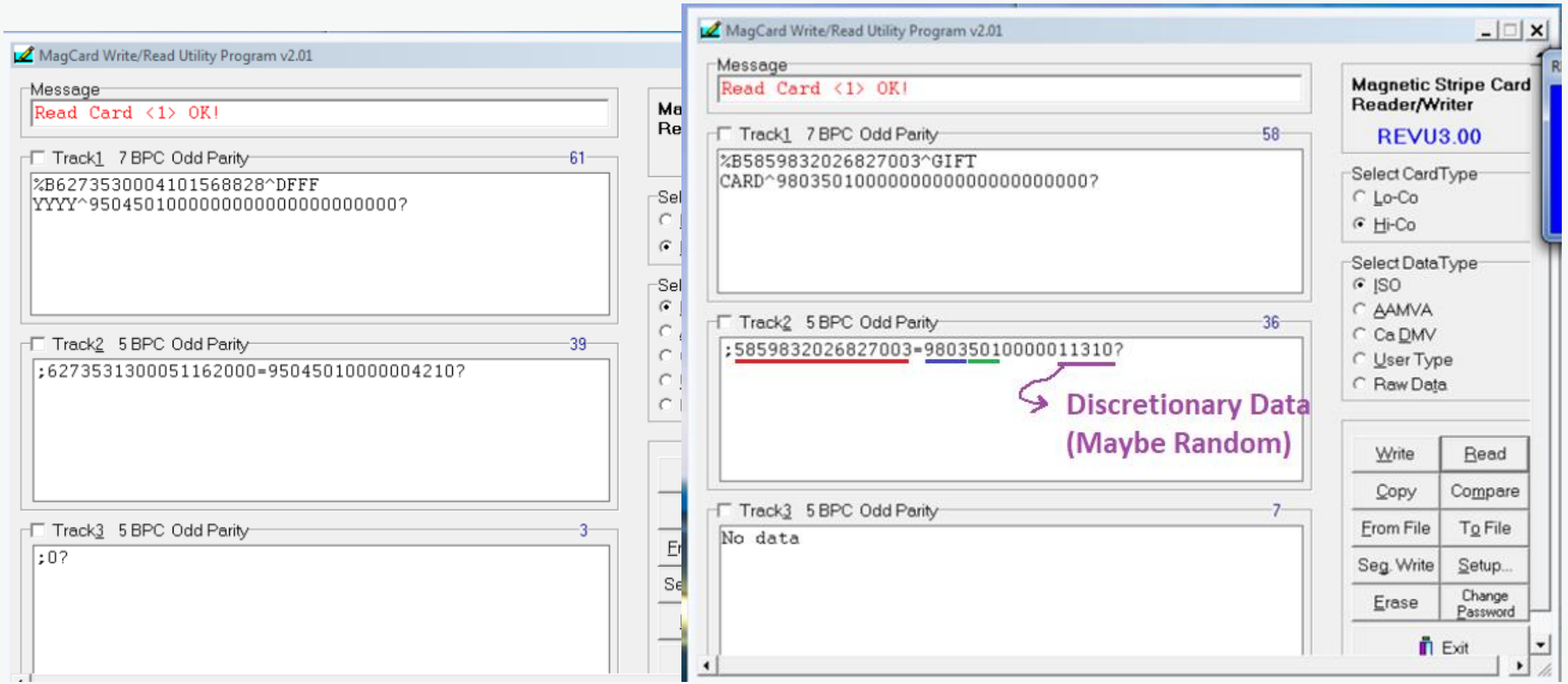
Name	Card Number	CVV2	ExpirationData	Random*	PIN
ebrahim	61043377.....	7823	139909	23464	1111



Magnet Stripe

Print on the Card Body

Magnetic Cards – Banking Cards



Magnetic Cards – Security

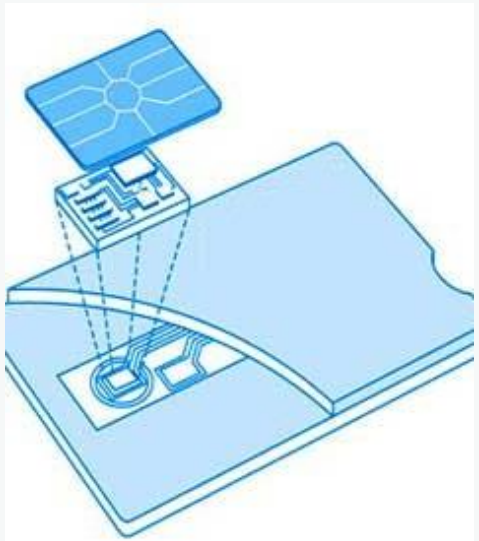
Chip Cards – Chip Type

- Dummy
  - Simple Memory (RFID Tags)
  - Memory + Access Management
  - Memory + Access Management + Secure Communication
- Smart
  - Programmable Microcontrollers

Smart

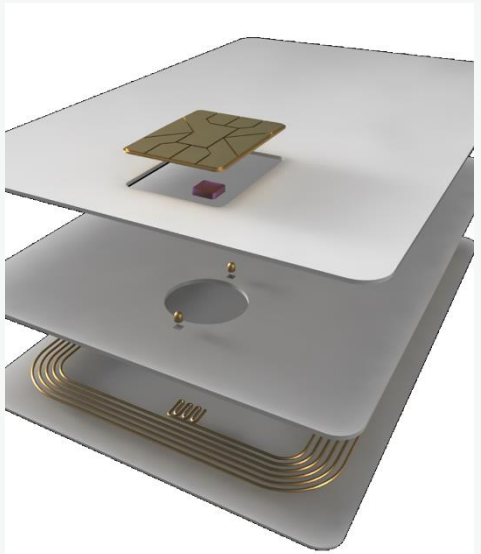
Dummy

Chip Cards – Interface Type

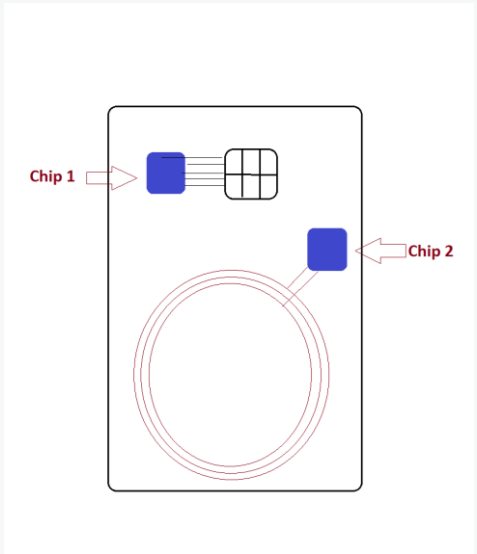


Contact

Contactless



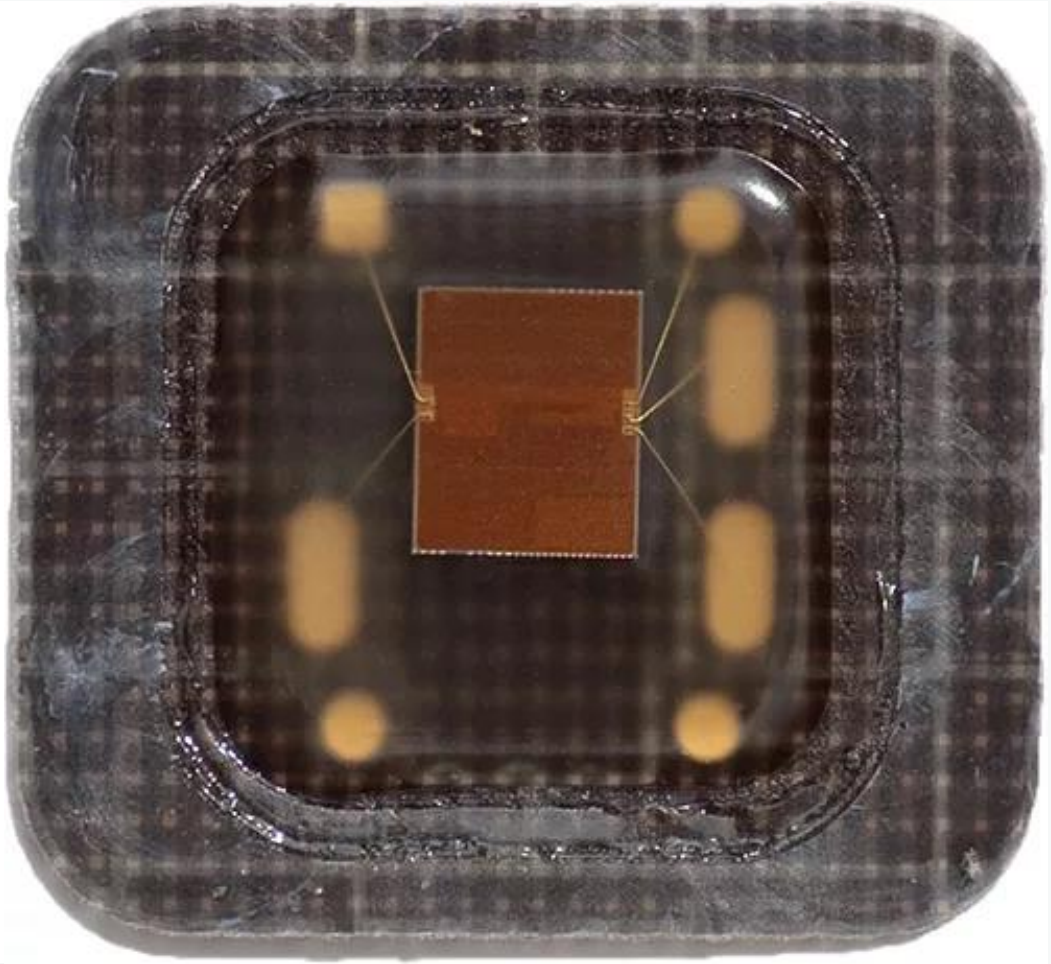
Dual Interface Combi



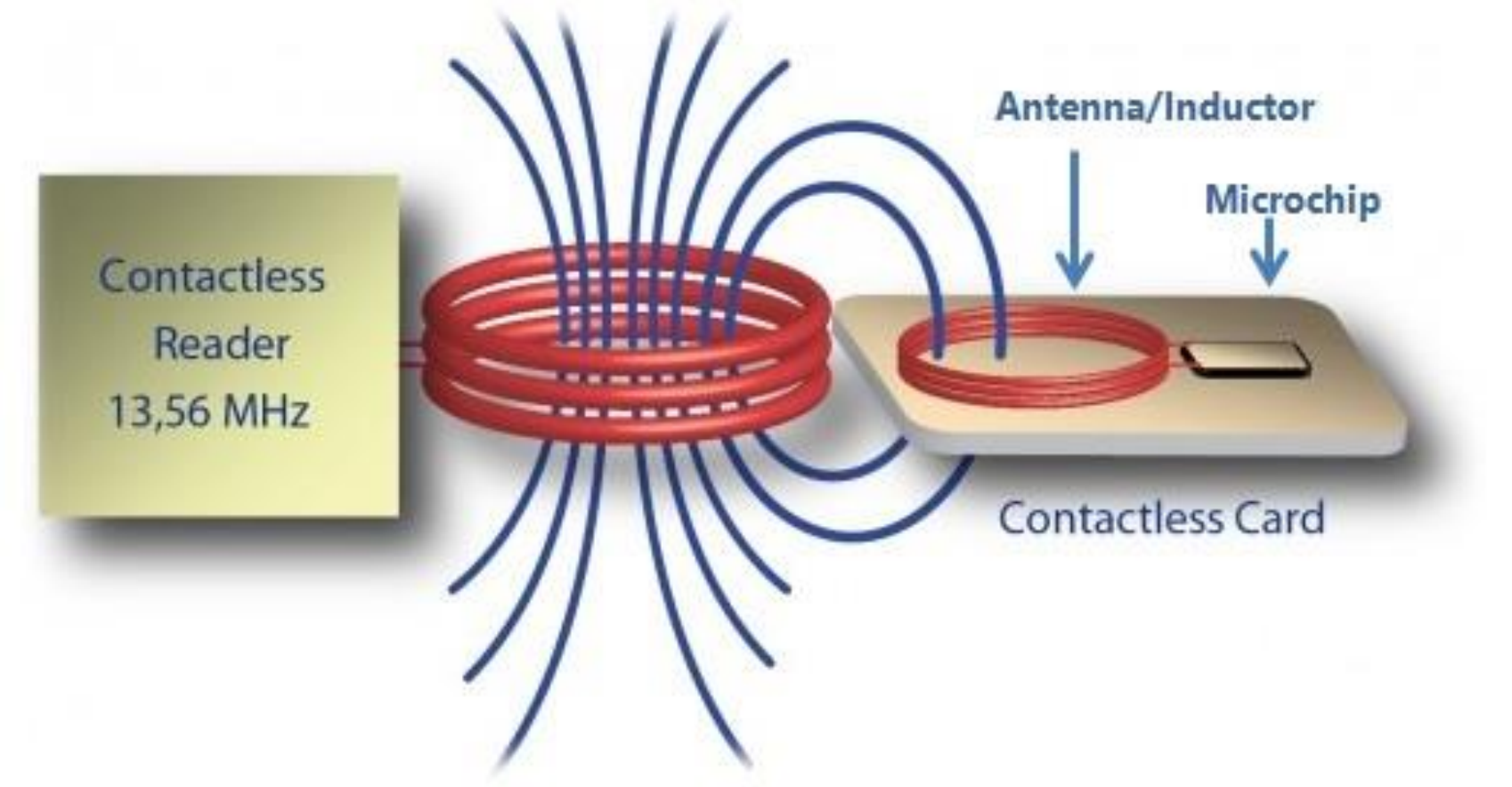
Dual Interface Hybrid

Chip Cards – Contact

- VCC: power supply
- RST: reset signal, used to reset the card's communications
- CLK: provides the card with a clock signal
- GND: ground (reference voltage)
- VPP: designated this as a programming voltage
- I/O: serial input and output (half-duplex).
- C4, c8: the two remaining contacts are used for usb interfaces and other uses



Chip Cards – Contactless



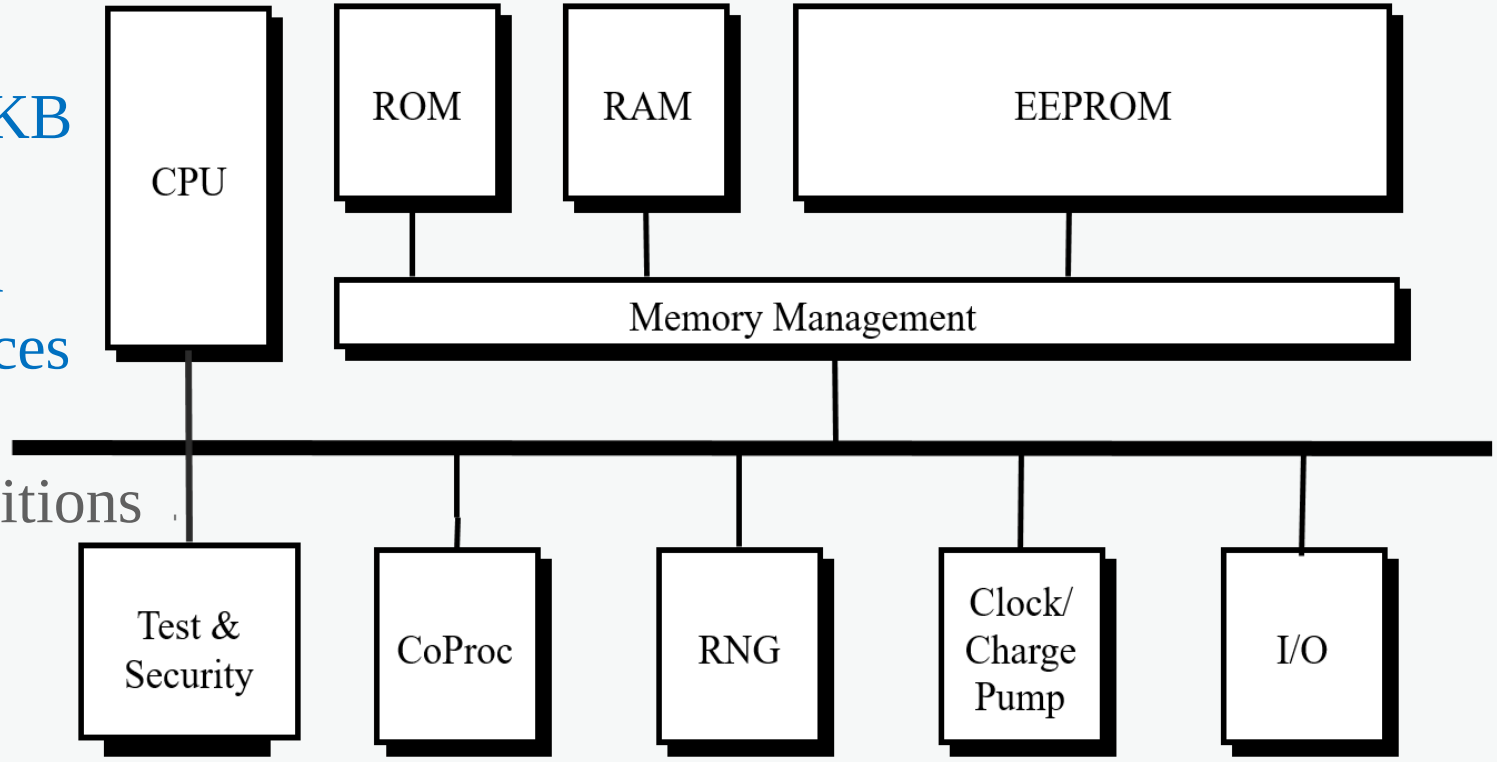
Chip Cards – Dummy

- They only store information
  - securely or insecurely
- Optional Passcode Protection
- Optional Secure Communication
- Can’t Process Information

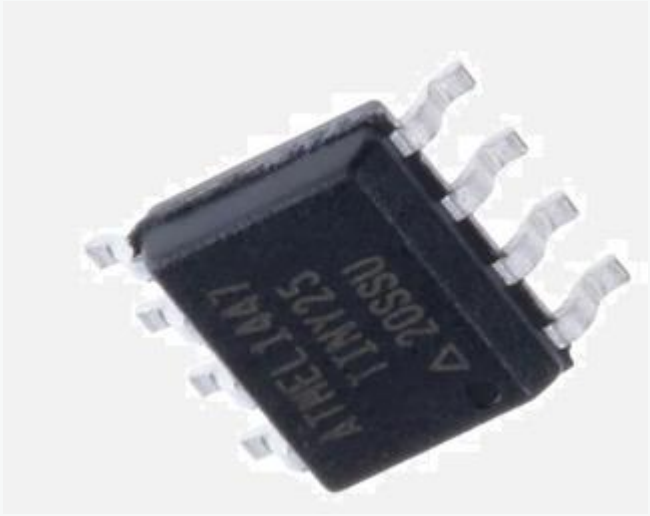
Sector	Block	Byte Number within a Block 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Description
15	3	Key A						Access Bits				Key B						Sector Trailer 15
	2																	Data
	1																	Data
	0																	Data
14	3	Key A						Access Bits				Key B						Sector Trailer 14
	2																	Data
	1																	Data
	0																	Data
:	:																	
:	:																	
:	:																	
1	3	Key A						Access Bits				Key B						Sector Trailer 1
	2																	Data
	1																	Data
	0																	Data
0	3	Key A						Access Bits				Key B						Sector Trailer 0
	2																	Data
	1																	Data
	0																	Manufacturer Block

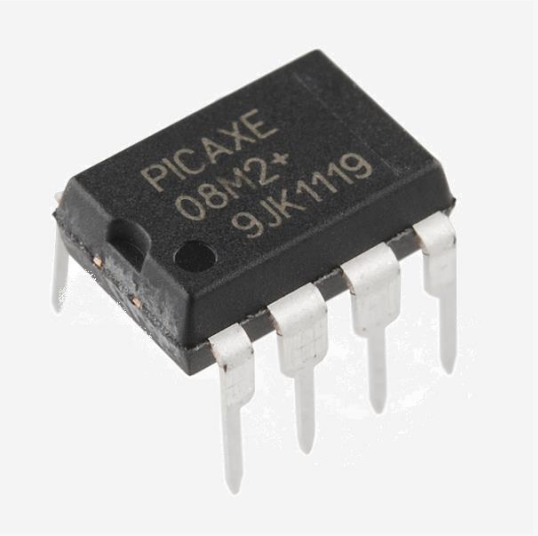
Microcontroller

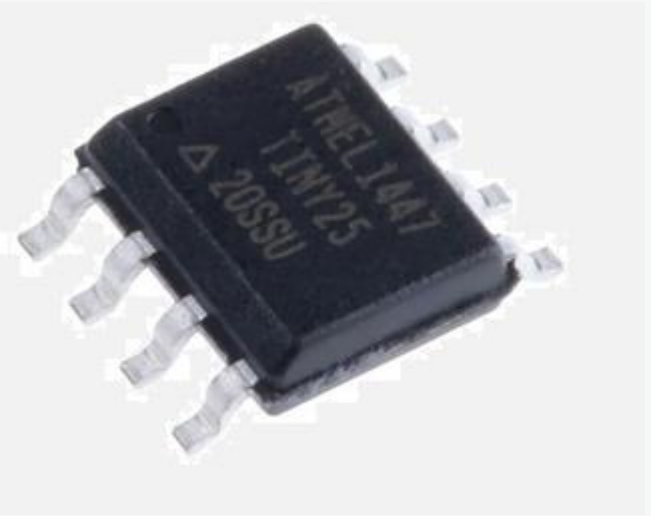
- CPU ~ 8,16,32 bit
- RAM ~ 3 K
- EEPROM/Flash ~ 40 … 200 KB
- Crypto Co-Processor
- Store and Process Information
- Provides Cryptographic Services
- Security Logic
  - Detecting Abnormal Conditions
- Test Logic
  - Self-Test Procedures



-[Smart Cards]- Native Cards Were Headache

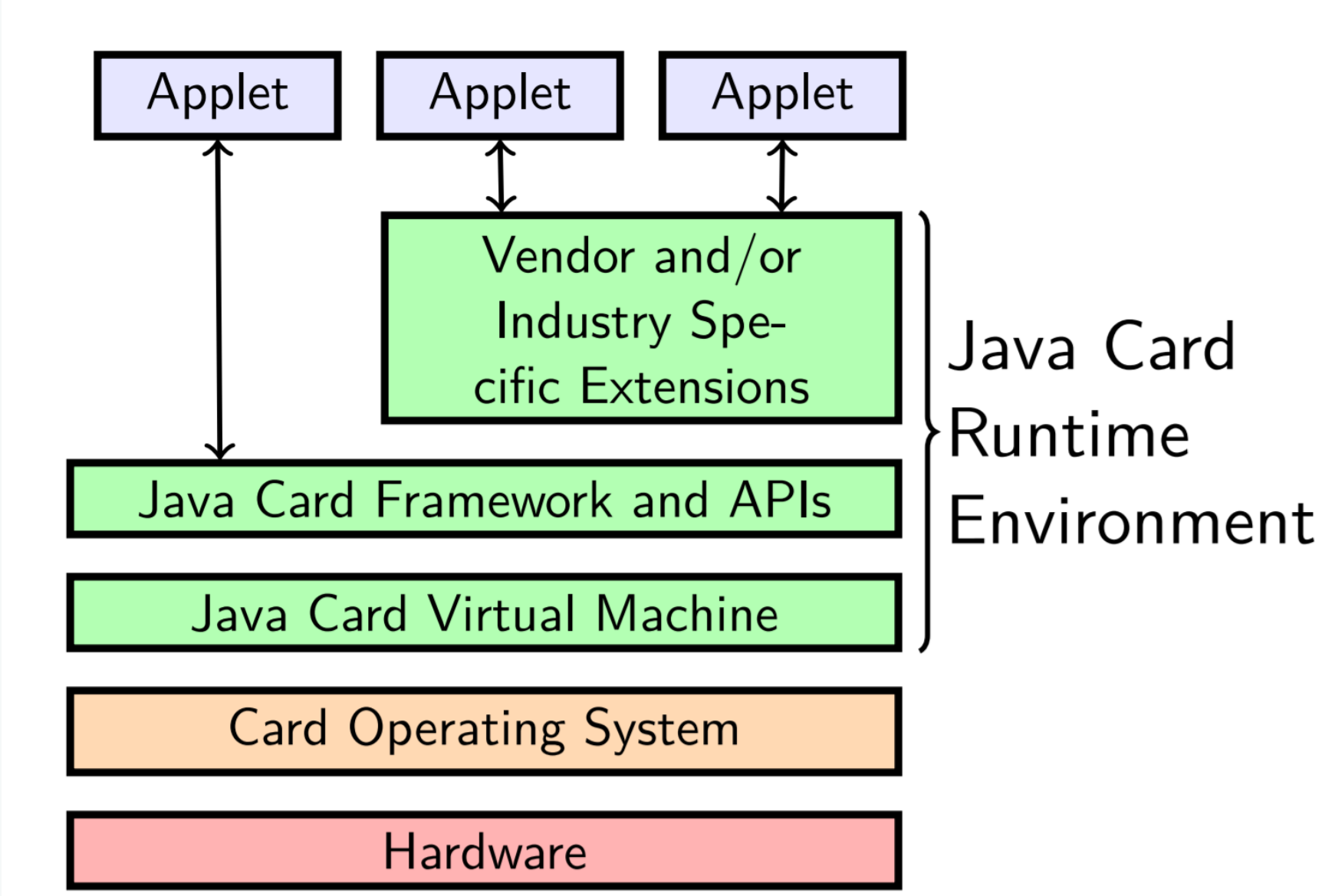




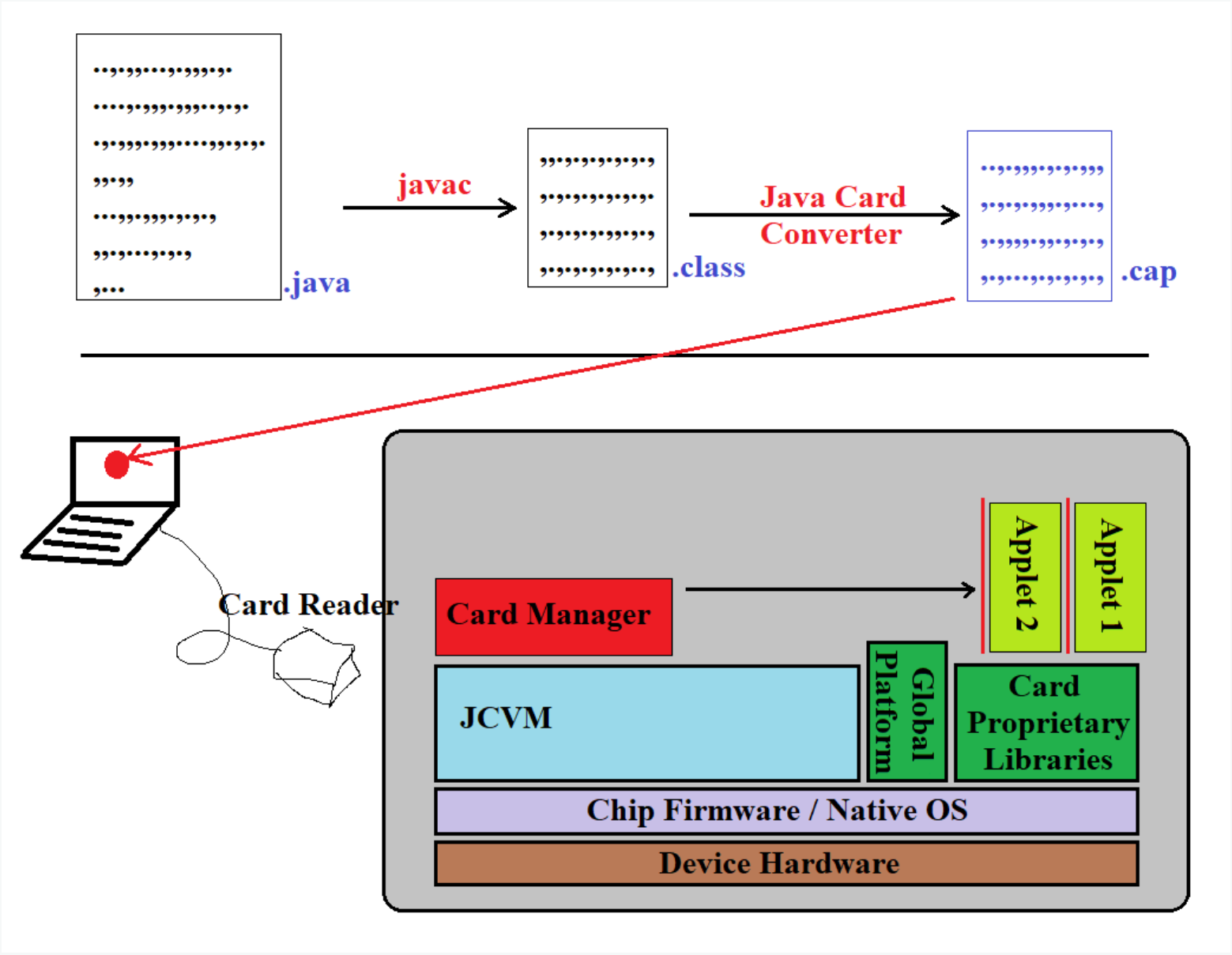


OFFSEC
www.offsec.ir


Hey Manufacturers!
Teach me your shitty Low-Level language and features and I will translate the developer's orders for you.
Hey Developers!
Don't worry about the hardware.
Talk to me in Javacard language.
Sun
microsystems


Applet
Applet
Applet
Vendor and/or Industry Specific Extensions
Java Card Framework and APIs
Java Card Virtual Machine
Card Operating System
Hardware
Java Card Runtime Environment

Javacards Applets



Applications

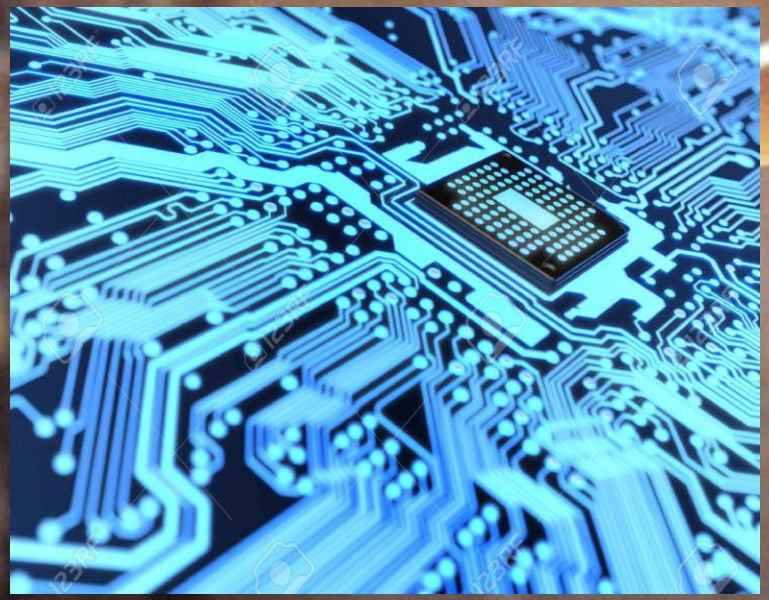
- Government
  - Identification
  - Passport
  - Driving License
- E-banking
  - Access to account
  - Electronic wallets
- Education and Office
  - Physical access control
  - Time registration
- Retail
  - Copyright Protection
  - Vending Machines
- Communication
  - SIM Cards
- Entertainment
  - Pay TV
  - Public event access control
- Transportation
  - Card Protection
  - Parking
- Health care
  - Electronic card for insurance data

Why?

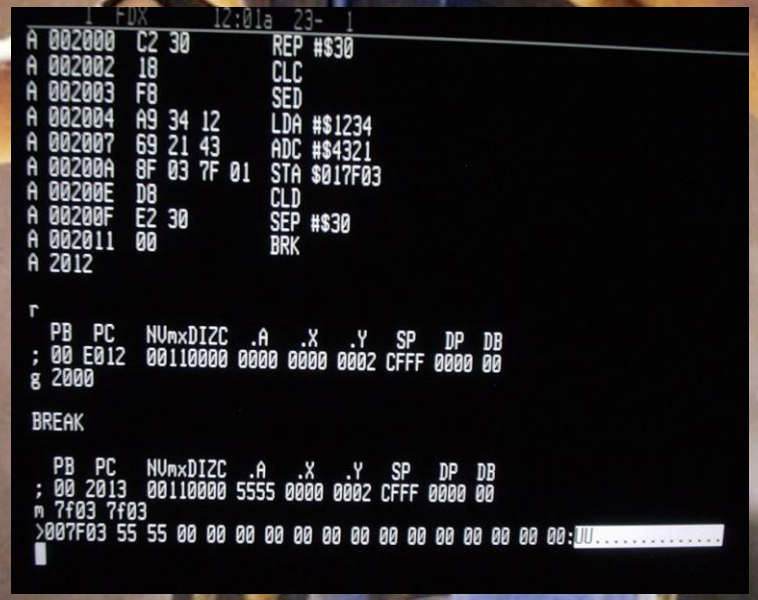
- Not all smart cards are secure! (Certificates are important)
- Using a smart card by itself doesn't lead to a better security.

-[Attacks]- Types

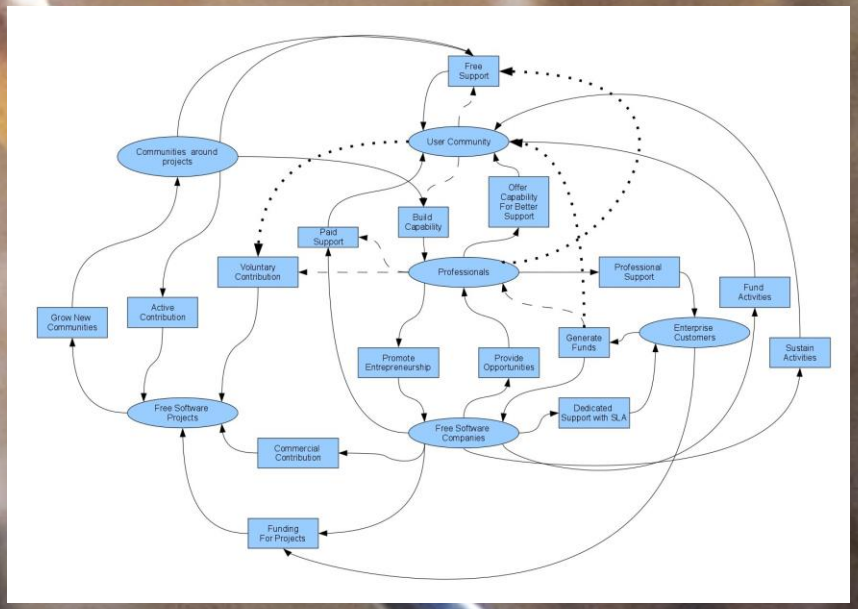
Hardware



Software



Ecosystem



-[Attacks]- Types

- Non-Invasive
  - Misconfiguration and Default Keys
  - Cryptanalysis and Implementation/Protocol Vulnerabilities/Weaknesses
  - Side-Channel Attacks ???
- Invasive
  - Probing
  - Fault Injection Attacks
  - Attacking Providers
    - Stealing the Keys
    - Reverse Engineering the Applet's "CAP" File
  - Non-Secure Programming
  - Looking For Bugs in the JCVM or the Card's Proprietary APIs ???
  - Reverse Engineering the Chip and Memory Contents
  - Command Scan and File System Scan

Misconfiguration & Default Keys

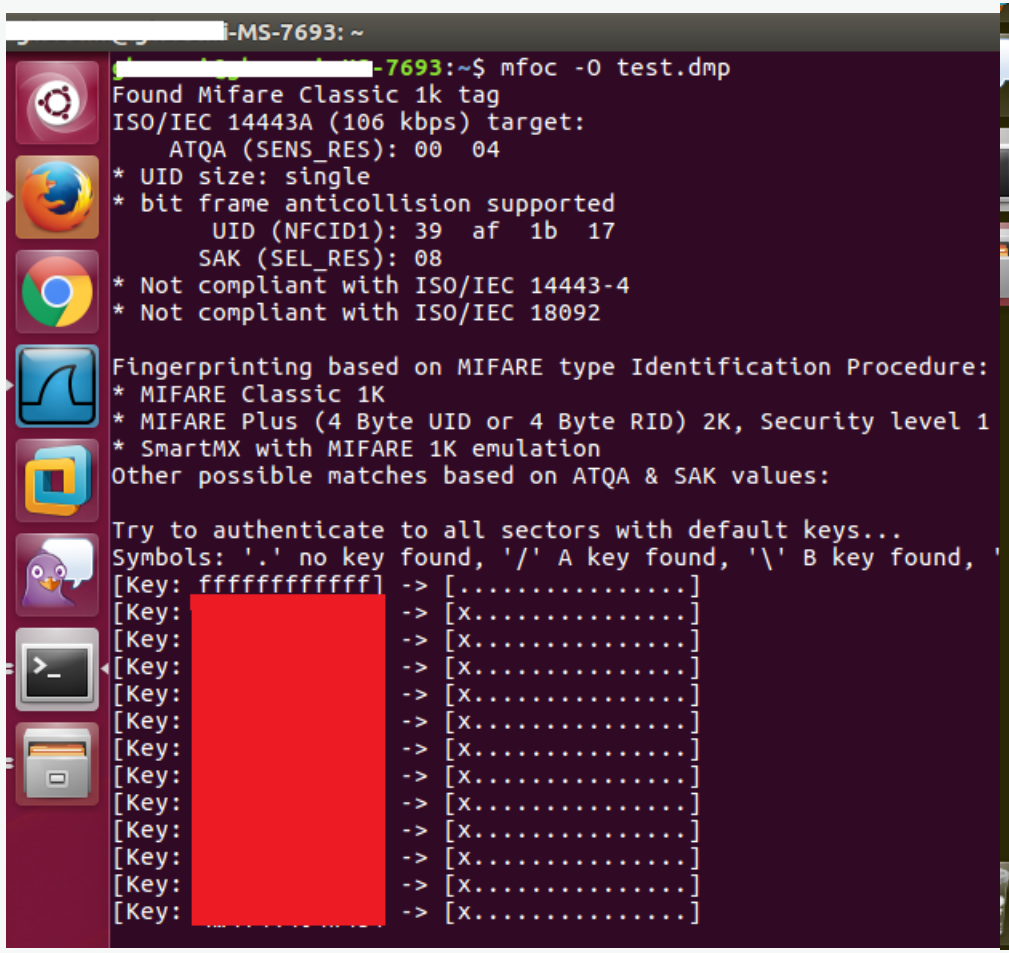
- Change all the keys before sending the cards to the WILD!
  - Global Platform Keys
  - OTA Keys
  - PIN/PUK numbers
  - Ki in SIMs
- Change the card's life-cycle and the applet's life-cycle if necessary.
  - Disable Personalization Functions.
  - Disable PIN/PUK Reset without key.

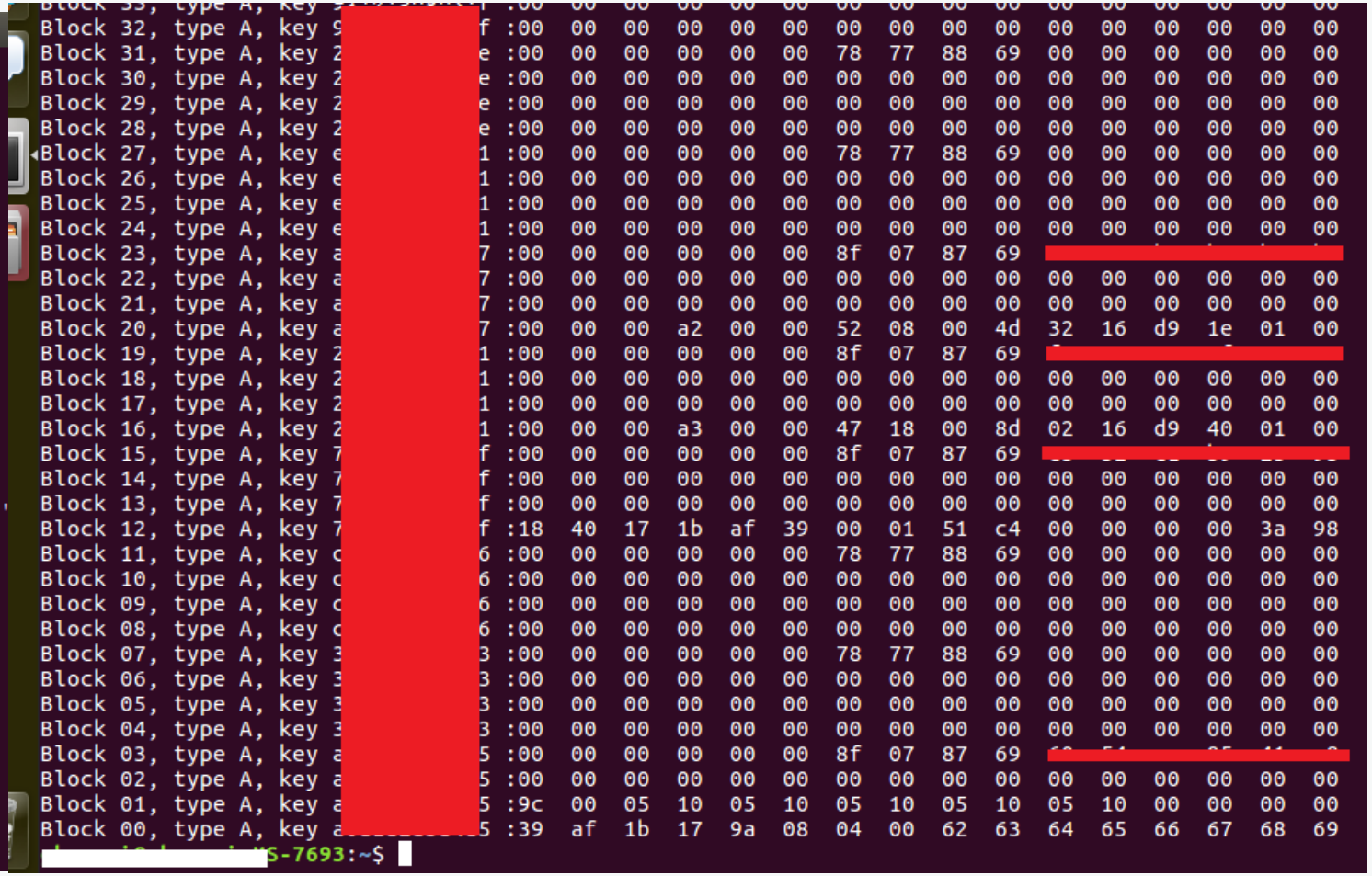
```
286
287
288 Default keys (AKA mother-key / master-key)
289 -- 404142434445464748494a4b4c4d4e4F
290 -- 47454D5850524553534F53414D504C45 (Ascii = GEMXPRESSOSAMPLE)
291
292
```

- Mifare Classic “Crypto-1” Protocol
- SIM Cards with DES signature on OTA command responses.
- “RSALib” library provided by Infineon Technology.

-[Attacks]- Cryptanalysis and … – Crypto-1

- RNG depends to the time between power up and authentication request!
- https://github.com/nfc-tools



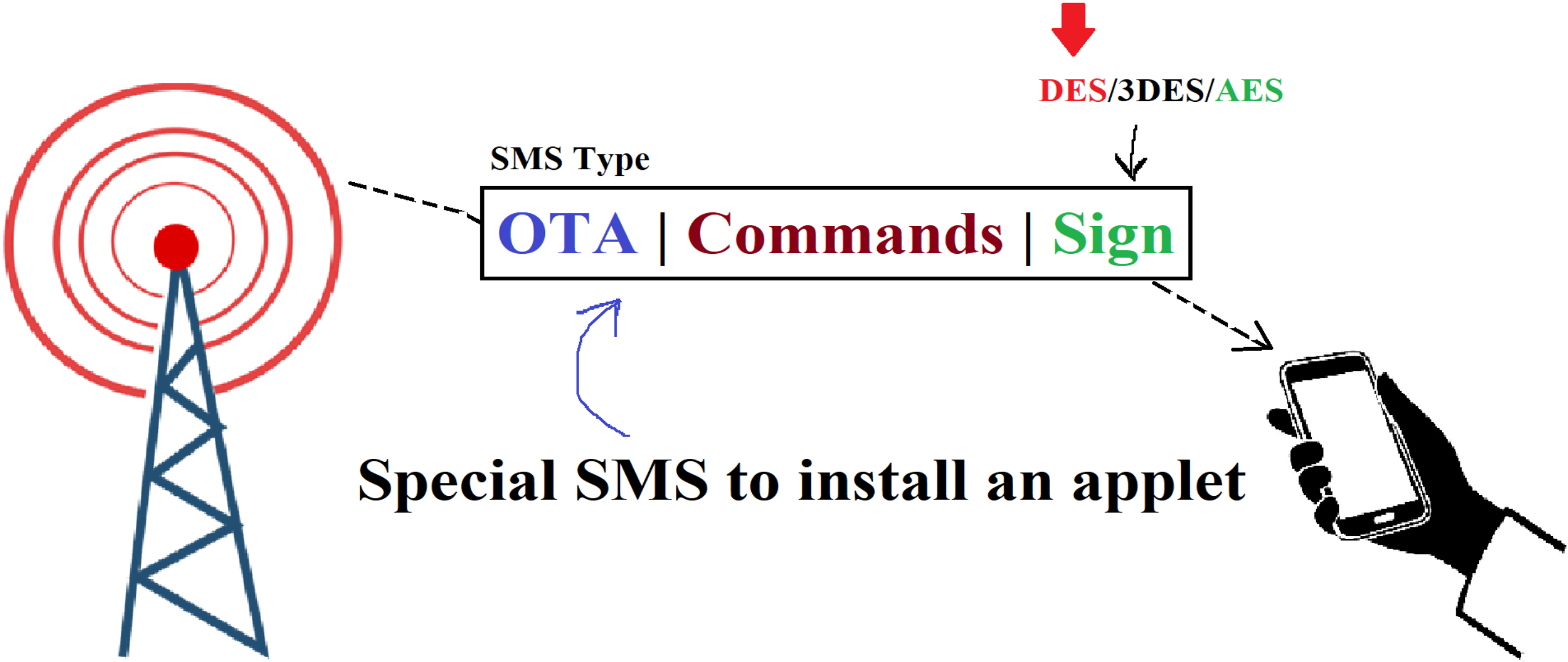


- And Finally After a Little Reverse Engineering …

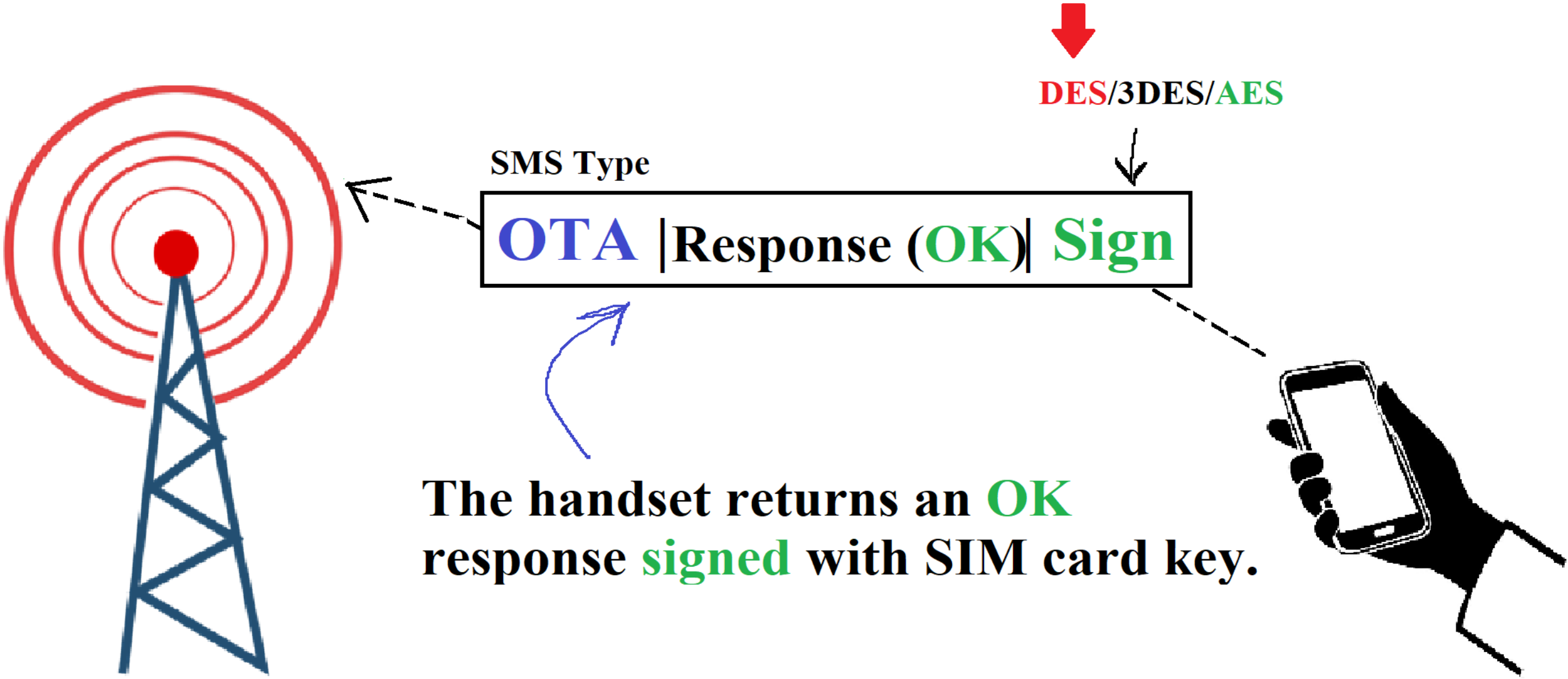
❑ How to Install an Applet on a SIM Card?

1) It’s a Smart Card! So the Smart Card Way
2) SIM Card Way ➔ Over The Air

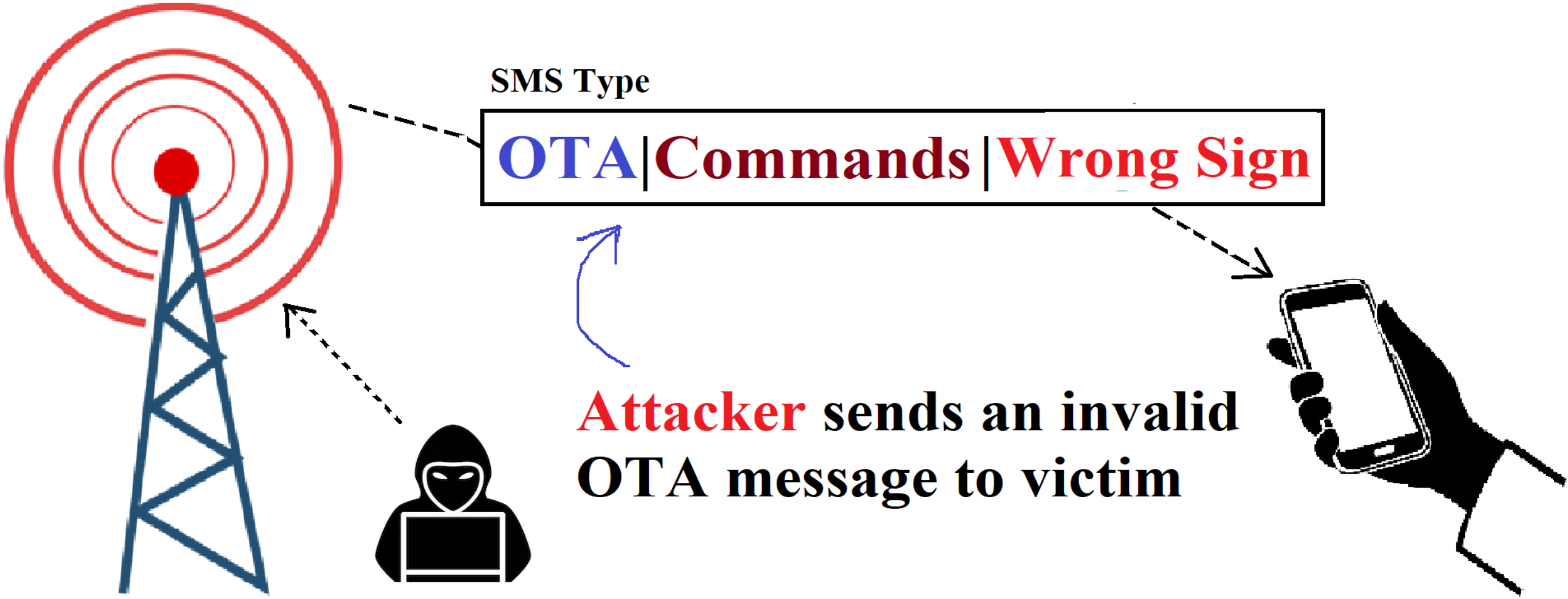
How does it works?



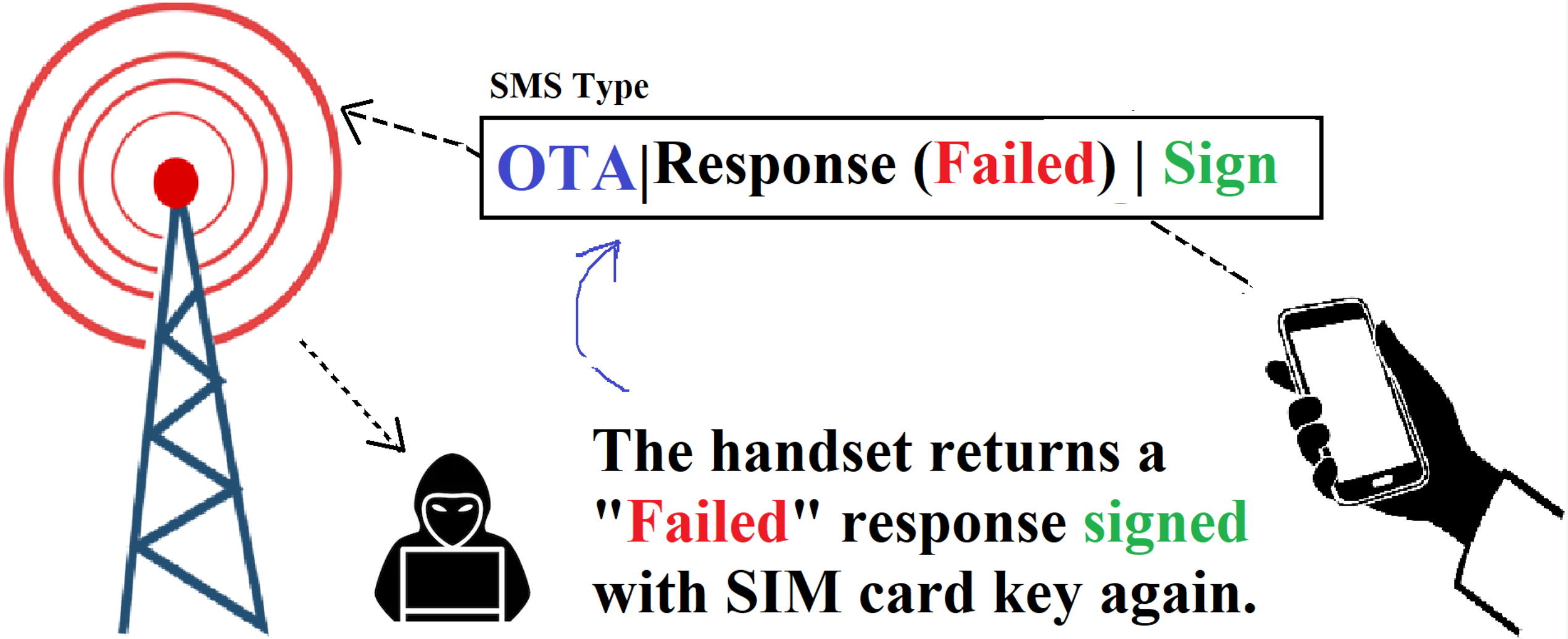
How does it works?



❑ How to attack? Step 1



❑ How to attack? Step 2



Finally!

What does he/she have?

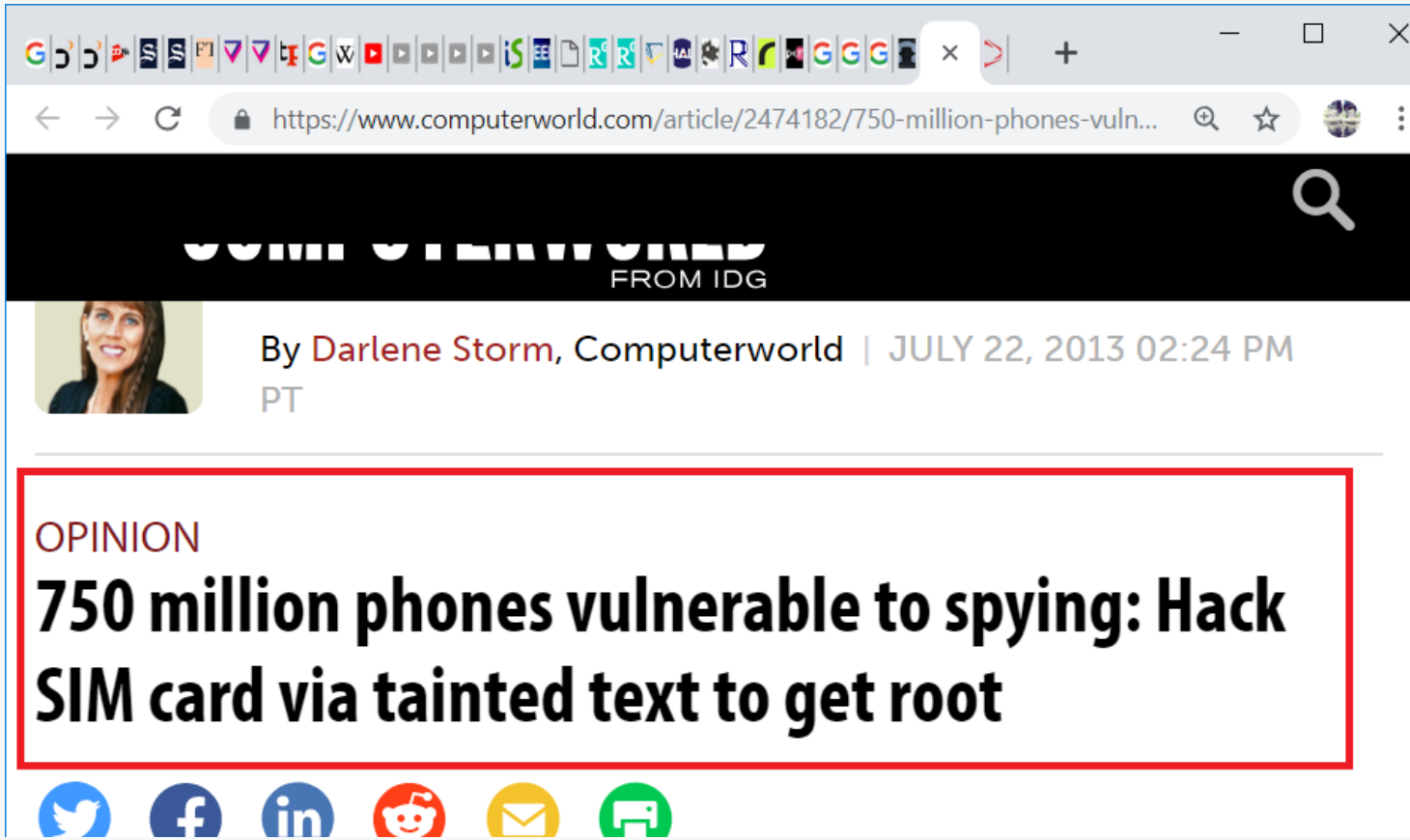
→ Text (Error Code)

→ Signature

→ DES Rainbow tables!

Countermeasures:

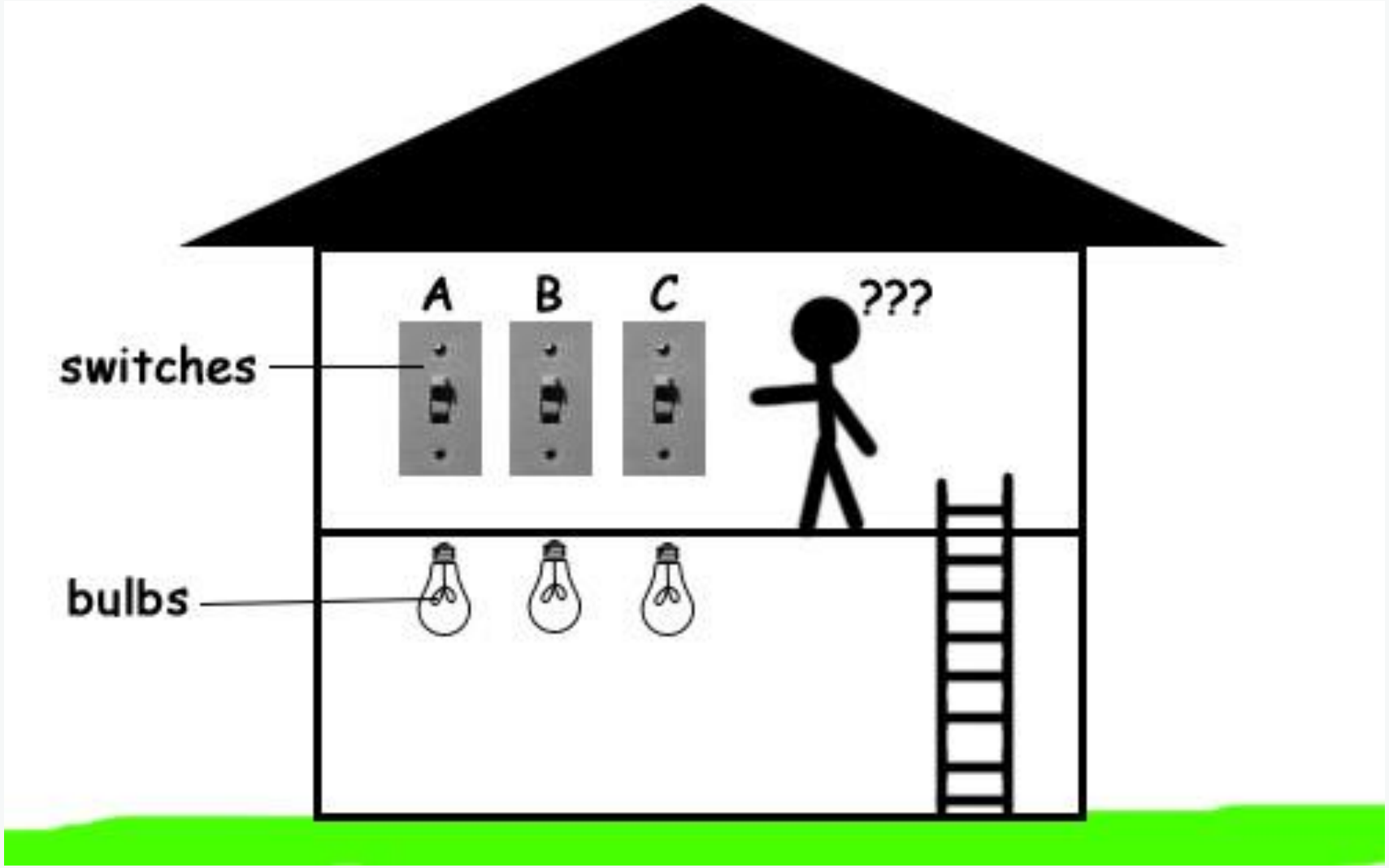
- ❑ Handset SMS firewall.
- ❑ In-network SMS filtering



Cryptanalysis and … – RSALib

- The library is incorporated in many smart cards and Trusted Platform Module (TPM) implementations.
- “ROCA” vulnerability: CVE-2017-15361
- A problem in RSA Key pair generation
- Allows the private key of a key pair to be recovered from the public key
- https://github.com/crocs-muni/roca
  - To check your key pairs

Side Channel Attacks – Example 1



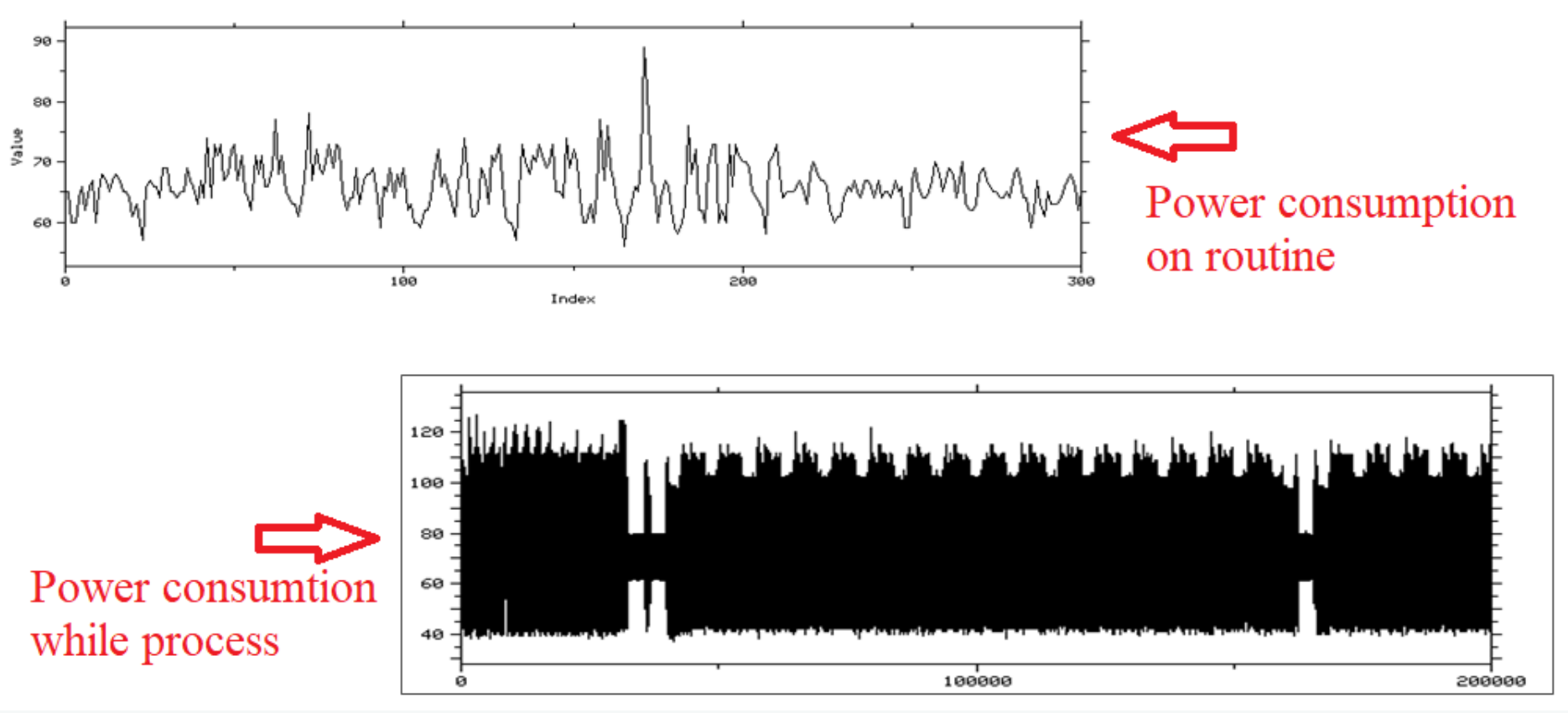
Side Channel Attacks – Example 2

- Heartbeat rate and body temperature

- Generated sounds

Side Channel Attacks – Smart Cards

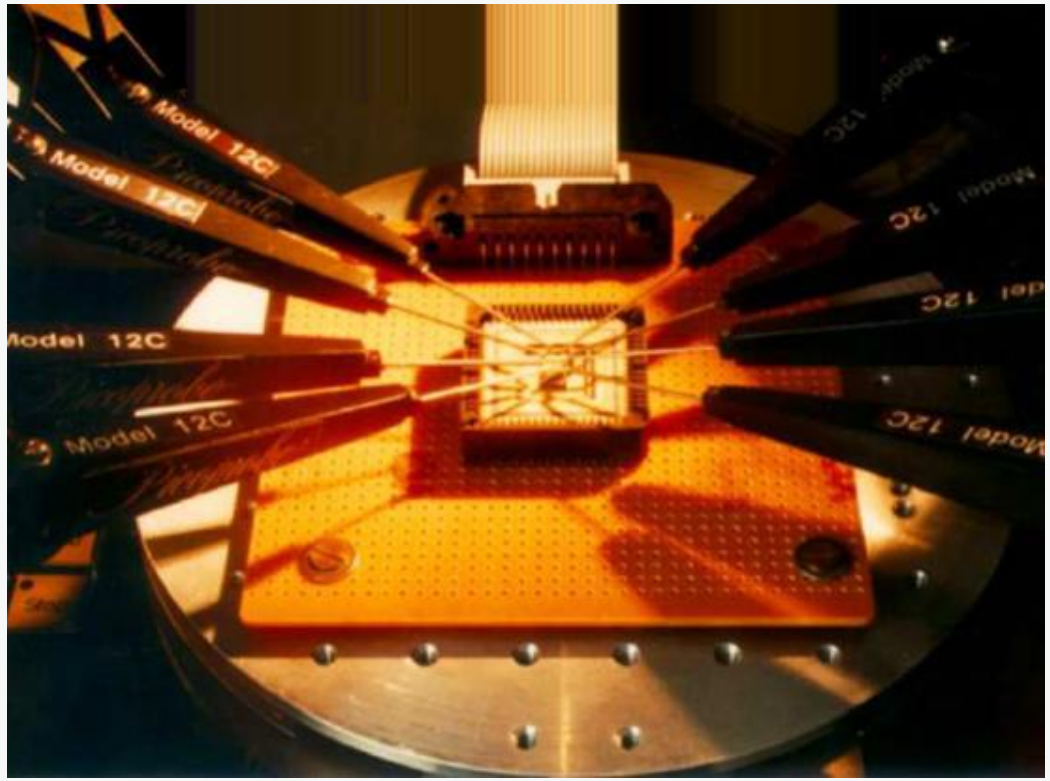
- Hidden Signals
  - Process execution time
  - Power consumption
  - Electromagnetic emission



Side Channel Attacks – Countermeasures

- Random wait states (NOP for example)
- Careful designing and coding of crypto algorithms
- Add noise to decrease signal to noise ratio
- Newer hardware design and newer technology

- Observe data on the chip data bus during operations using needles.
- Reverse engineering
  - Extracting sensitive data
  - Modifying data on the way
- Countermeasures
  - Using smaller circuits
  - Protective layers and sensors on the chip
  - Scrambled or encrypted bus

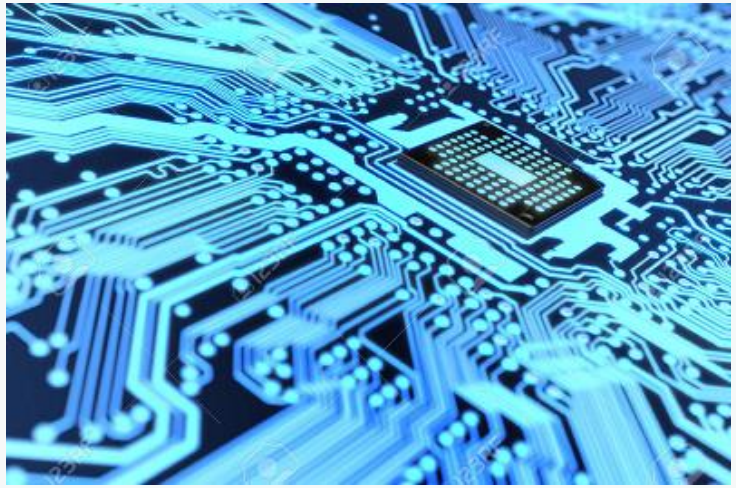


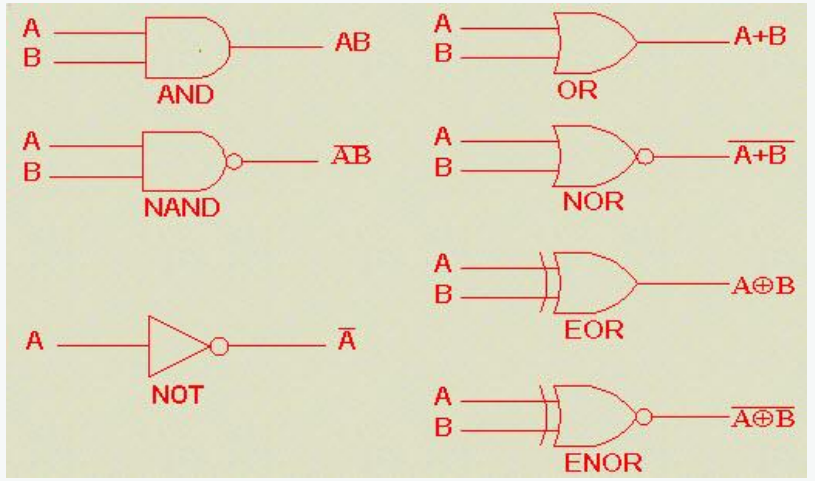
Fault Injection (AKA Confusion) – What is it?

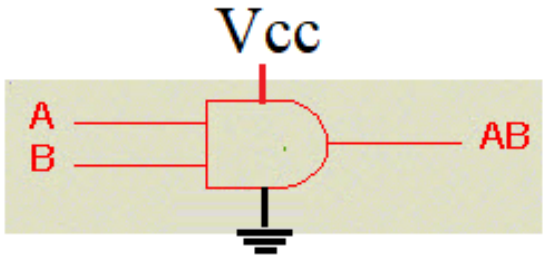
- Smart Card's environmental variables
  - Power Supply
  - Clock Frequency
  - Temperature
  - Environment Electromagnetic emission or ionizing radiation!
- Manipulating an environmental variable to:
  - Change a value read from (or write to) memory to another value
  - Prevent Execution of a CPU instruction.

- Countermeasures
  - Low/High voltage and frequency sensors
  - CRC and error detection mechanisms
  - Randomize timing of operations using NOP instructions
  - Electricity Capacitors!

Fault Injection (AKA Confusion) – Why?







V_{AB}	< 0.6	0.6 <
Logic	0	1

A	B	AB
0	0	0.3
0	1	0.4
1	0	0.4
1	1	4.5

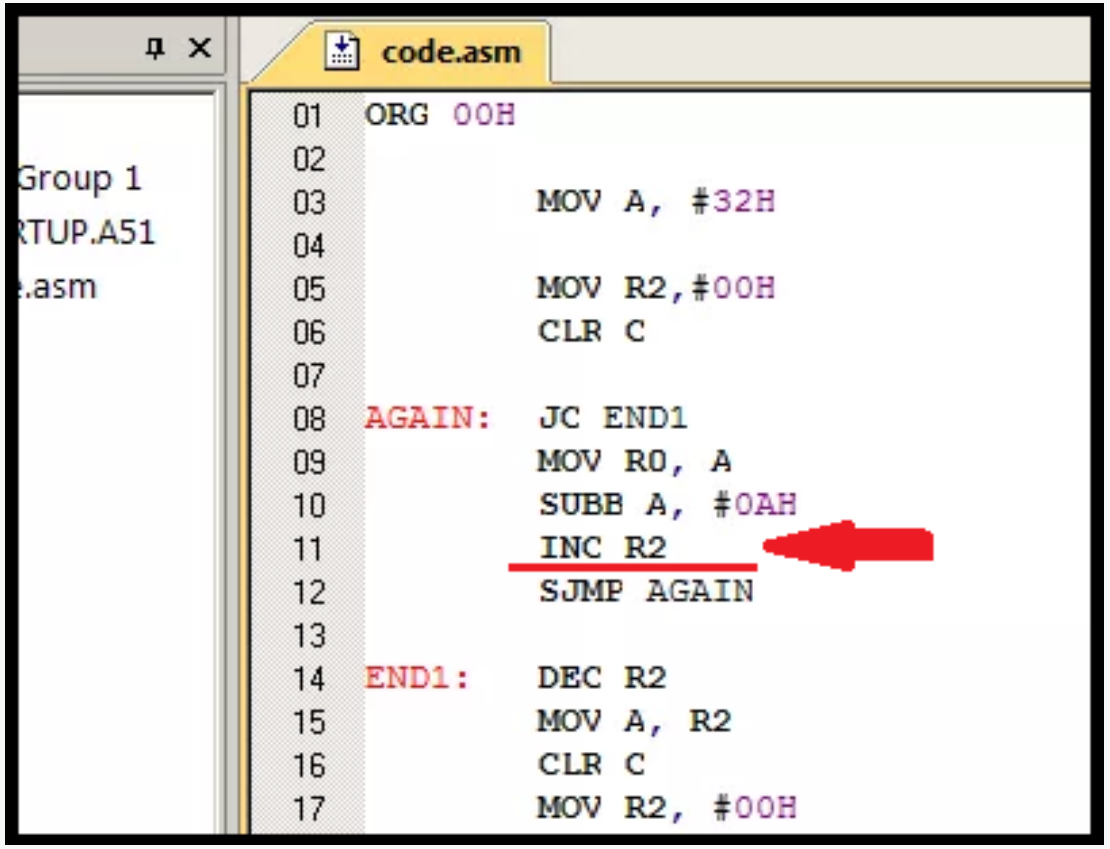
Vcc = 5v

A	B	AB	
0	0	0.3	0.5
0	1	0.4	0.7
1	0	0.4	0.7
1	1	4.5	

Vcc = 6v

Fault Injection (AKA Confusion) – Why?

- Overclocking on specific instructions
  - MOV ins or Memory block read/write
- Power cut on special instruction
  - Example: Failed PIN tries counter increment

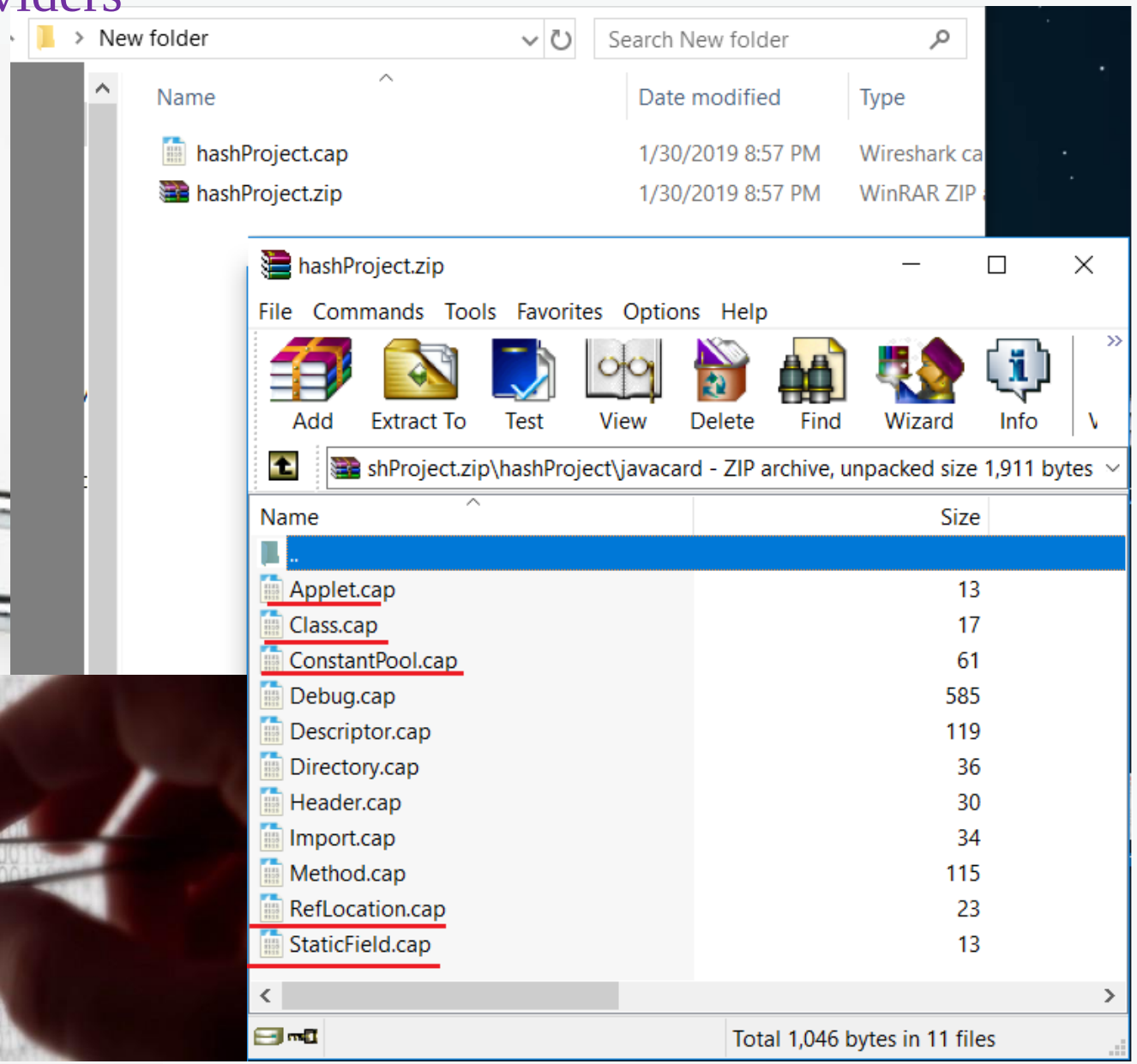


Fault Injection (AKA Confusion) – Goal?

- Prevent an EEPROM write:
  - PIN wrong tries counter
- Read Memory Contents as Zero
  - Applet's life cycle status
  - PIN value
  - A crypto-key
- Prevent Execution of a CPU instruction
  - Decreasing value of *PIN_Wrong_Tries_Counter*
  - Changing applet's life cycle status
- Malfunction
  - Generate a fixed number as a "Random" number
  - Induce errors to reveal internal states of cryptographic modules

Attacking Providers

- Security is a chain
- Modifying source codes
- Looking for “cap” files and then Reverse Engineering
- Looking for authentication keys.



```
1  package testPack;
2
3  import javacard.framework.*;
4
5  public class TestApplet extends Applet {
6
7      private TestApplet() {
8      }
9
10     public static void install(byte bArray[], short bOffset, byte bLength)
11             throws ISOException {
12         new TestApplet().register();
13     }
14
15     public void process(APDU apdu) throws ISOException {
16
17         byte[] buffer = apdu.getBuffer();
18         byte[] temp = new byte[100];
19
20         doSomething(buffer, temp);
21     }
22 }
```

No Automatic Garbage Collection in Javacards

Not-Secure Programming - Secure

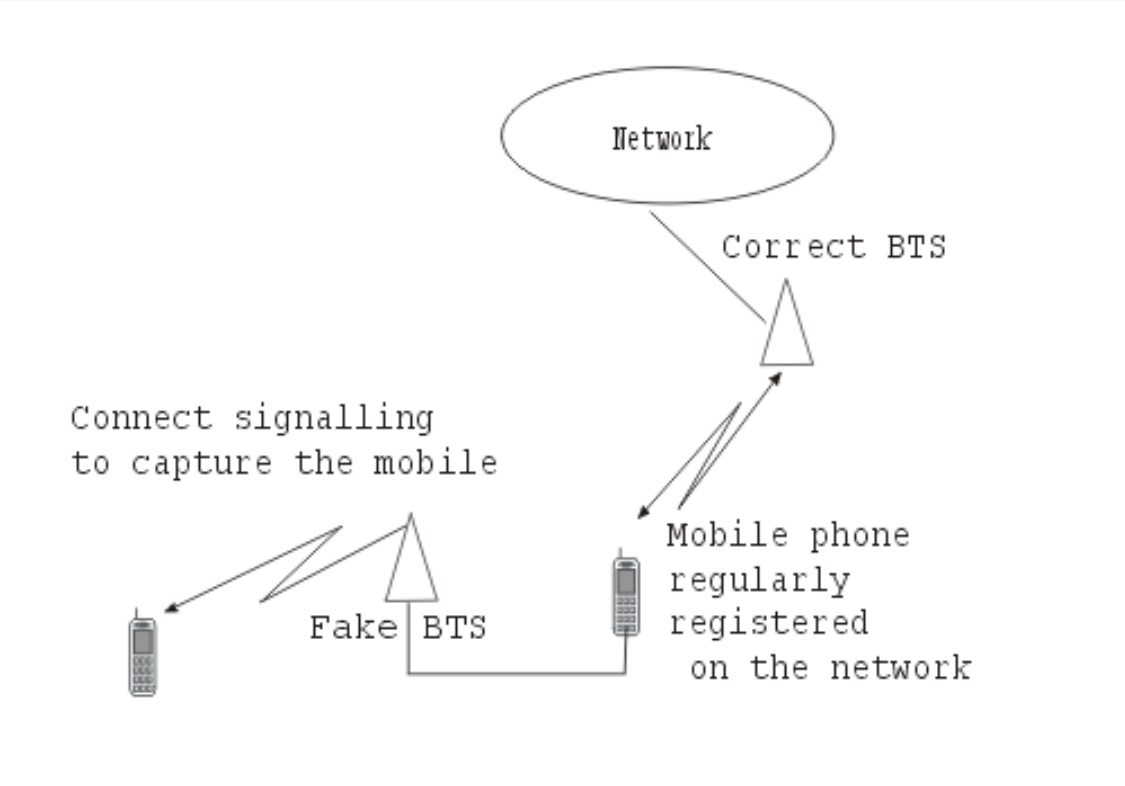
```
package testPack;

import javacard.framework.*;

public class TestApplet extends Applet {

    public static byte[] temp;

    private TestApplet() {
    }

    public static void install(byte bArray[], short bOffset, byte bLength)
            throws ISOException {
        temp = new byte[100];
        new TestApplet().register();
    }

    public void process(APDU apdu) throws ISOException {

        byte[] buffer = apdu.getBuffer();
        doSomething(buffer, temp);
    }
}
```

Reuse the 100 bytes for all commands.

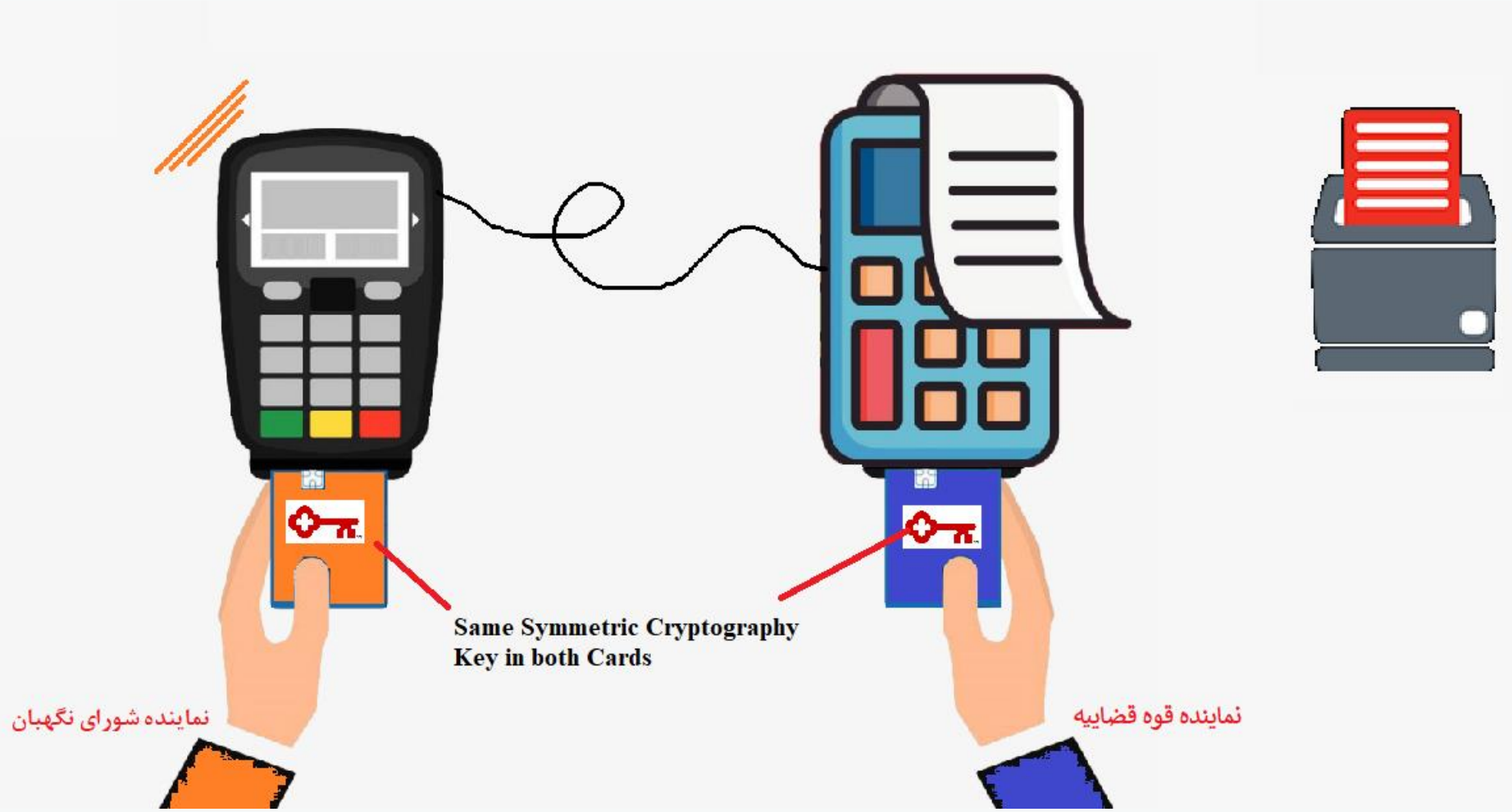
*** Transient object are also available**

Not-Secure Programming – Single End Point Authentication

- Both terminal and card can be forged. Attacker may introduce himself as another end-point

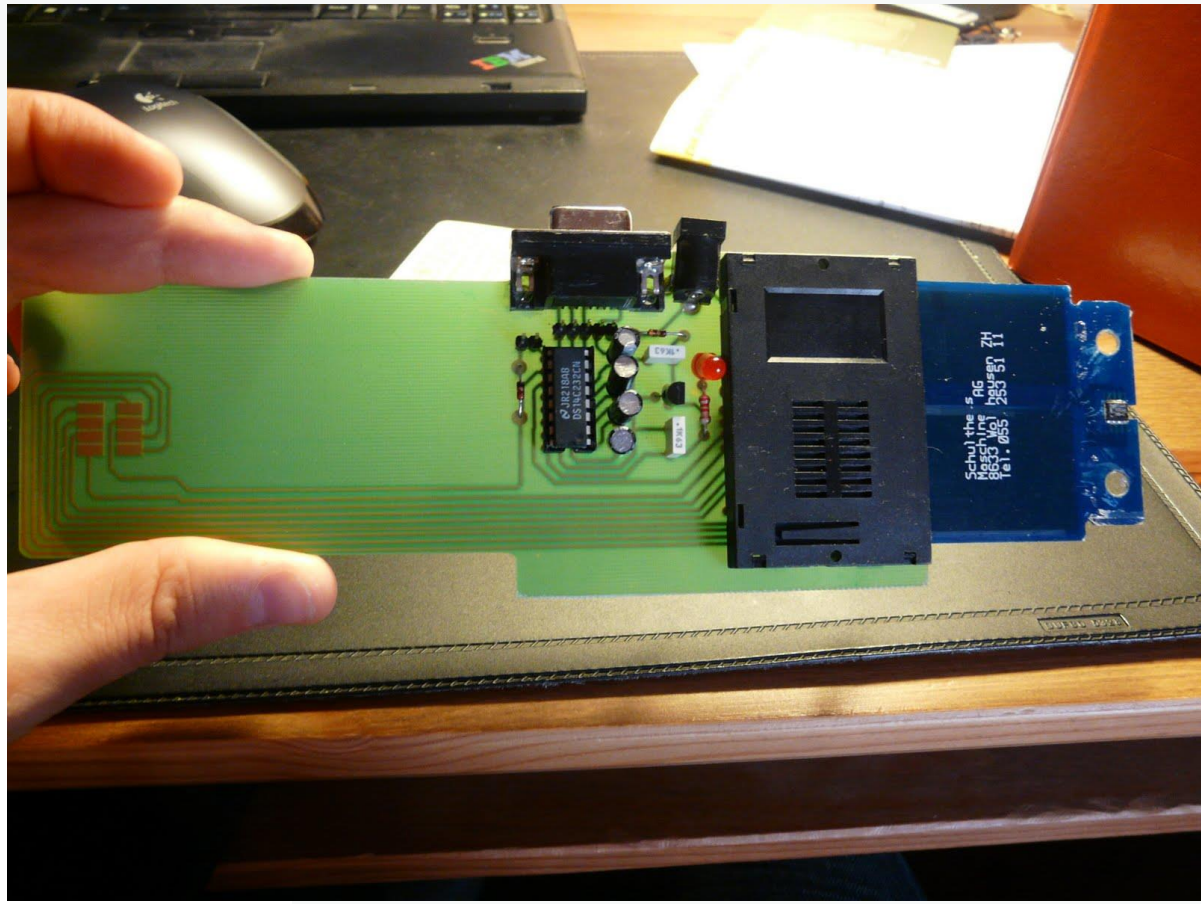


- E-Voting devices (**Laboratory** version)



Not-Secure Programming – Single End Point Authentication

- Communication Logger
  - Hardware:: Sniffer
  - Software:: Logger Applet



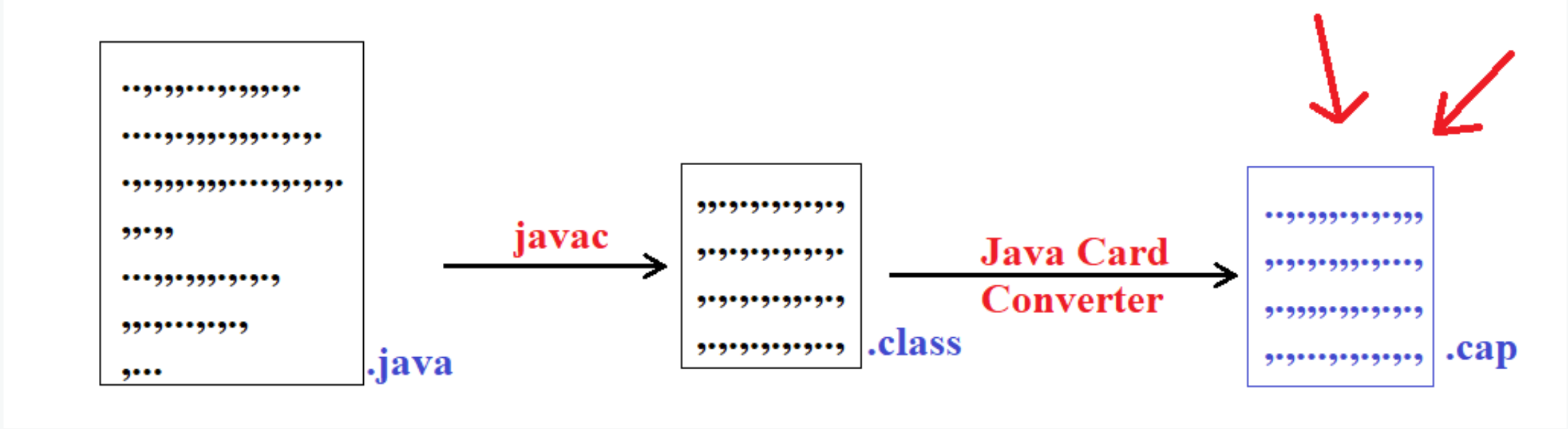
```
public class LoggerApplet extends Applet {

    public static byte[] log_array;
    public static short index = 0;

    private LoggerApplet() {
    }

    public static void install(byte bArray[], short bOffset, byte
            throws ISOException {
        log_array = new byte[100];
        new LoggerApplet().register();
    }

    public void process(APDU apdu) throws ISOException {

        byte[] buffer = apdu.getBuffer();
        JCSystem.arrayCopyNonAtomic(buffer, (short)0, log_array, (short) index,
                                    (short)(buffer[ISO7816.OFFSET_LC] + 5));
        index += (short)(buffer[ISO7816.OFFSET_LC] + 5);

        ..... // Switch Case to reply commands or to return log_array contents.
    }
```

Not-Secure Programming

- A “Write_To_EEPROM/FLASH” function is publicly available?
  - Memory wear out
- PIN verification or Card access authentication are available through contactless interface?
  - Multiple tries on PIN/Key verification with wrong values to break the card.

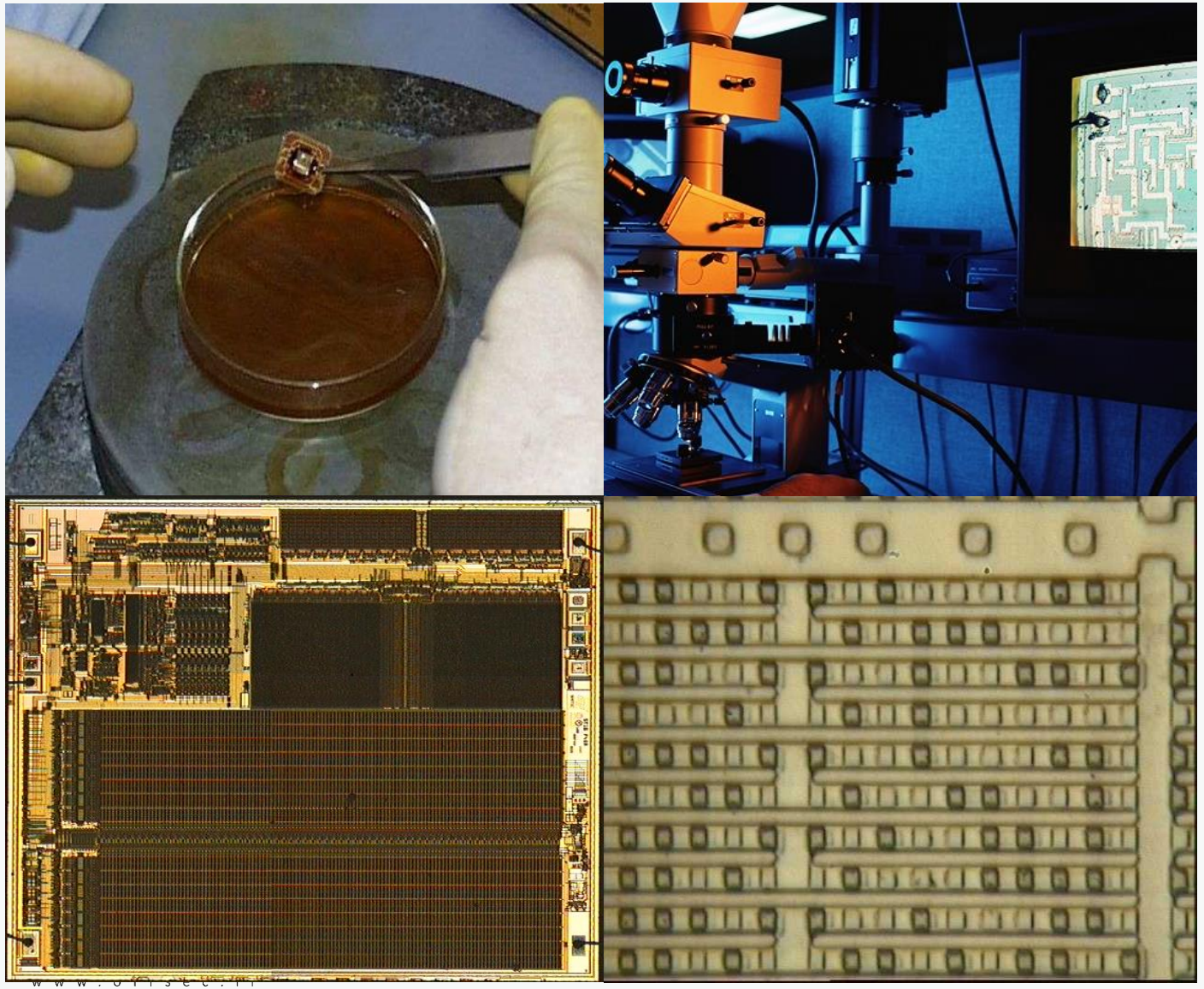
-[Attacks]- Looking For Bugs in the JCVM or in the Card's Proprietary APIs

- Follow Stackoverflow and Oracle community Javacard questions
  - Feitian smart cards and list of installed applets when a package with "Long AIDs" is present on the card.
  - JCOP Card's object deletion and power down
- After the Off-card ".class" file verifier. (the .class to .cap Converter)



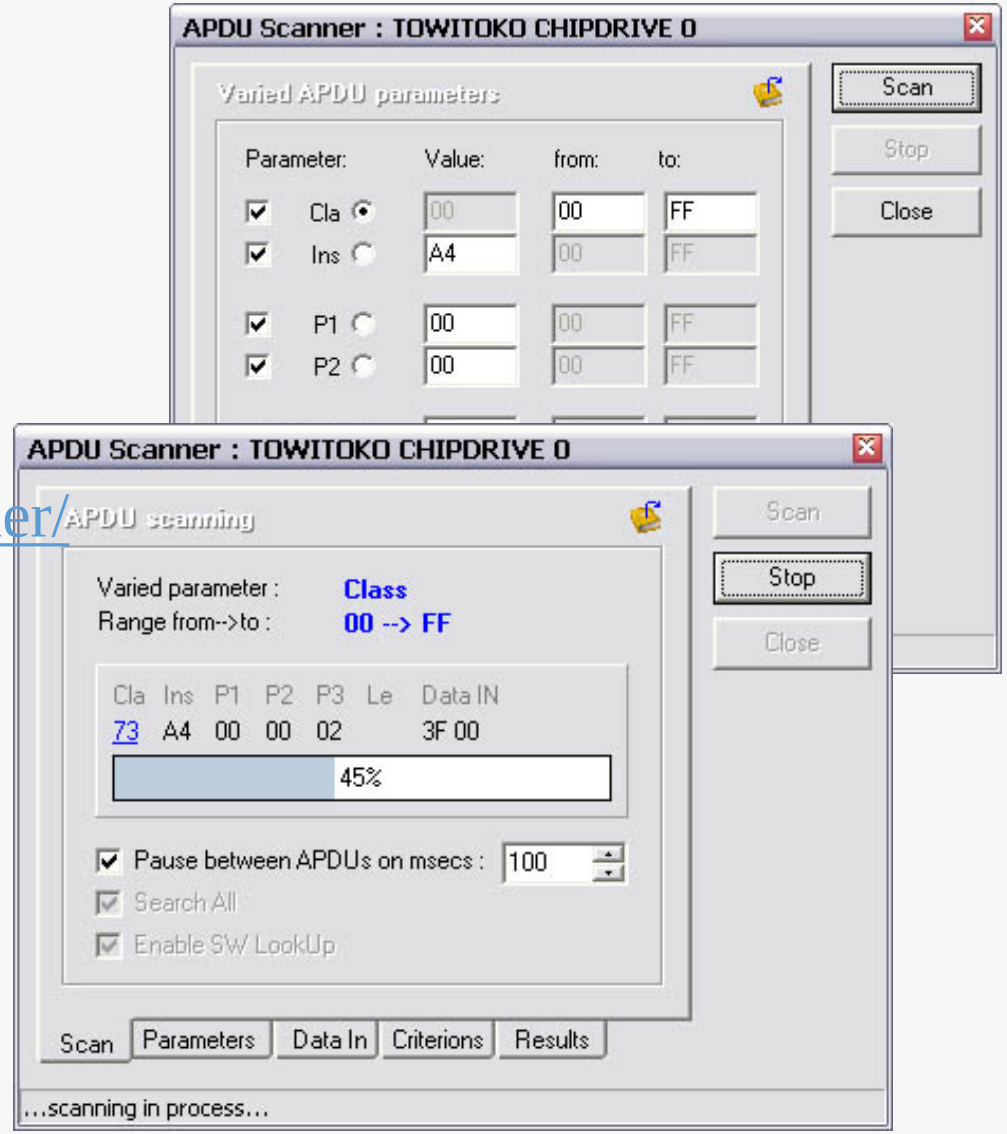
-[Attacks]- Reverse Engineering the Chip and Memory Contents

- Remove chip from the SC.
- Use chemicals to remove epoxy resin and the top silicon/metal layer of the chip.
- Microscope
- Focused ION Beam (FIB)
  - Not only to observe, but also make changes.
- Looking at 1's and 0's in memory
- Reverse Engineering
- Reading/Modifying the Memory
  - Resetting the security lock bit.



Command Scan and File System Scan

- 5-Bytes commands
  - 2^40
  - The order is important → (2^40)!
  - Life-Cycle matters
- Restriction on file access
- You must trust the card manufacturer
- https://sourceforge.net/projects/apduscanner/
- Scanning all APDU commands?
  - In theory and practically impossible!



-[Conclusion]- ....

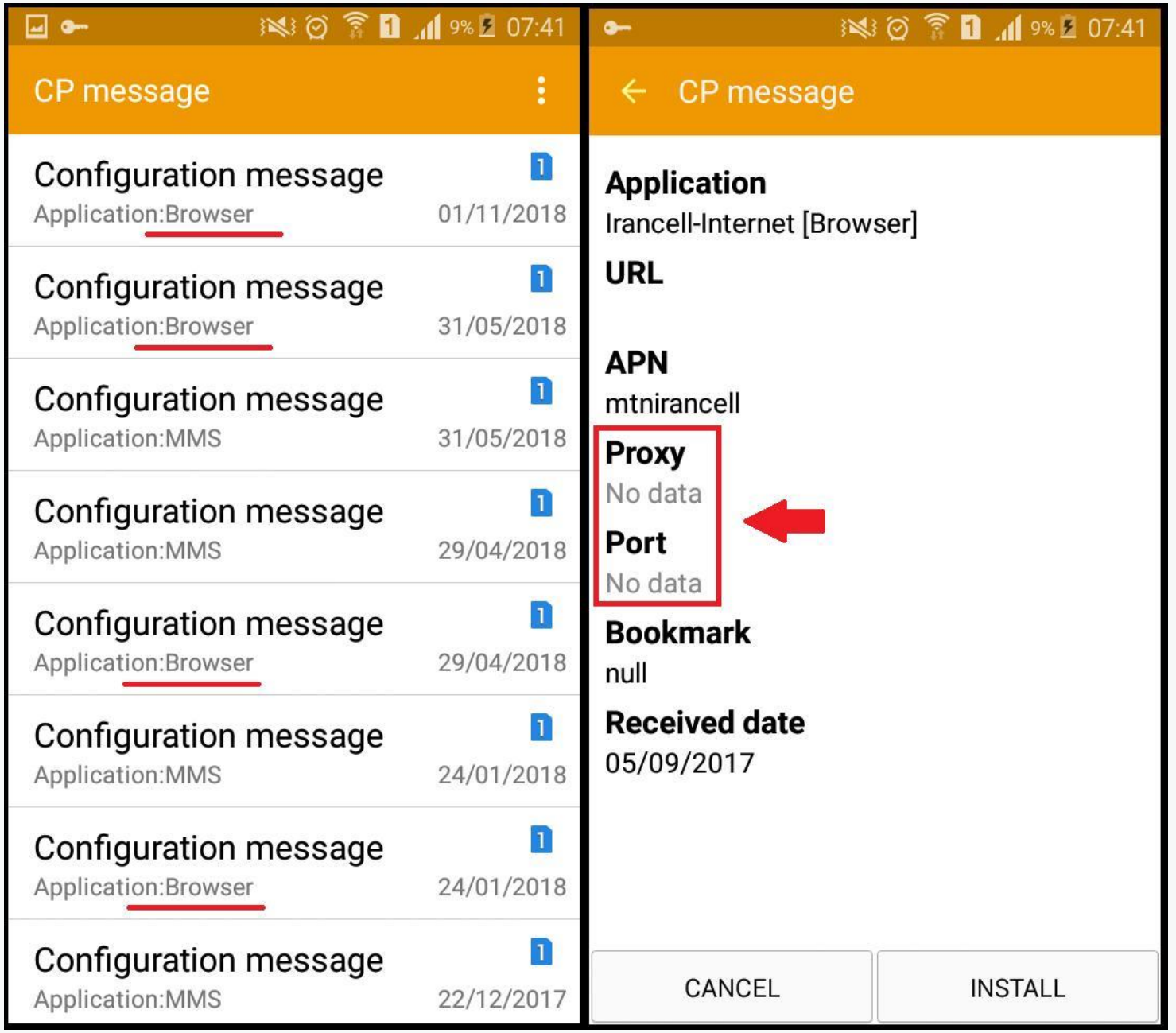
- Perfect security does not exist.
- Smart cards can be broken by advanced analysis techniques.
- A bad applet can destroy the system
- Users of security systems should think about:
  - What is the value of our secrets?
  - What are the risks (e.g. fraud, eavesdropping)?
  - What are the costs and benefits of fraud?
- Certifications are important
  - FIPS 140 U.S. Government Security Standard
  - Common Criteria (AKA ISO/IEC 15408)
  - NIST

-[Questions]-

Thank you for your attention.

-[Miscellaneous]-

- SMS
  - Text
  - Binary
    - Configuration SMS
- GSM Modem
- http://www.nowsms.com
- MITM
- Malware Infection



-[Miscellaneous]-

- *800*1#
- *800*1#
- *800*1#
- *800*1#
- *800*1#
- *800*1#
- *800*1#
- *800*1#
- *800*1#
- *800*1#