PV204 Security technologies LABS

Introduction to smart cards

Petr Švenda *svenda@fi.muni.cz* *@rngsec*
Centre for Research on Cryptography and Security, Masaryk University

CROCS
Centre for Research on Cryptography and Security

The masterplan for this lab

1. Communicate with smart card (GPPro tool)
   - ATR, basic info, CPLC
2. Communicate with card programmatically
   - Java java.smartcardio.* or C/C++ PC/SC API
   - CPLC data
   - Obtain list of supported instructions from unknown card
- Project work

1. Communicate with smart card (GPPro)

- Contact PC/SC readers + cards
- GlobalPlatformPro tool
  - https://github.com/martinpaljak/GlobalPlatformPro/releases
  - Basic smart card commands, sending APDUs
  - Management of GlobalPlatform cards (JavaCard)
  - Type `gp --help` for all functionality
  - We will use basic functionality now, rest next week

gp --info

- Obtain information about smart card
  - gp --info
  - Obtain ATR (Answer To Reset)
  - Parse using https://smartcard-atr.appspot.com/parse?ATR=xxx

- Who is the probable manufacturer of card?
- What is the probable environment for this card?
- Is it an open JavaCard?
- What is this card's circuit serial number?
- When was the card produced?
- What is different if 'gp --info --debug' is executed?

gp --apdu APDU_in_hexa --debug

- Send APDU command from command line
- Try gp --info --debug
  - Can you spot APDU command to obtain CPLC info?
- Send get CPLC APDU separately
  - gp --apdu 80CA9F7F00 --debug

- Can you relate card's response data and gp --info?
- What is the response status word?

2. Communicate with card programmatically

- SimpleAPDU project (IS, NetBeans)
  - Uses Java's javax.smartcardio.* API
  - CardMngr.java – utility functions for card communication
- Obtain list of available readers
  - List readers = TerminalFactory.getDefault().terminals().list();
- Connect to card
  - CardTerminal.isCardPresent(), CardTerminal.connect("*");
- Obtain ATR: Card.getATR().getBytes()
- Send APDU:
  - ResponseAPDU resp = CardChannel.transmit(apdu)

3. Communicate with card programmatically

- Try to send get CPLC command
  - Pre-prepared in GetCPLCData() method
  - Necessary to set proper APDU
- Parse response buffer

- Can you relate card's response data and gp --info?
- What is value of response status word?

Supported commands

- Card responds to some APDU commands
  - Generic ones (e.g., get CPLC data)
  - Custom ones (what card's owner wants)
  - Usually CLA/INS/P1 only (P2 sometimes)
- How to get list of commands supported by a card?
  - Look into documentation / standard (e.g., SIM commands)
  - Try to probe card (limited number of possible commands)
    - Be careful – many failed attempts may block your card!
- Task: find at least one ADU supported by card

Optional: BF complete list of all supported commands

- Write code that will try all combination if CLA/INS
- Observe response codes
- Make list of CLA/INS which returns interesting code
- Analyse with curiosity!

IMPORTANT: use only with the provided blue card or card you can afford to brick (e.g., old unused banking card)

SOLUTIONS (KIND OF ☺)
(VALID FOR BLUE CARDS FROM LAB)

gp --info

- Who is probable manufacturer of card?
  - Gemplus/Gemalto
- What is probable environment for this card?
  - Possibly JavaCard with MPCOS applet
- Is it open JavaCard?
  - No (no CardManager with known keys)
- What is card's circuit serial number?
  - ICSerialNumber: 02006FC1 (Note: your card will be different)
  - Good to consider also other ICxxx values for uniqueness
- When was card produced?
  - ICFabricationDate: 1105
  - Probably 15th May 2011 (105th day of year ending with 1)

Probing unknown commands

- Probing possible because of:
  - limited space of command values
  - error message side channel
  - missing failed tries counter