Automated Reverse Engineering using Lego®

Georg Chalupar and Stefan Peherstorfer
University of Applied Sciences Upper Austria

Erik Poll and Joeri de Ruiter
Radboud University Nijmegen
Introduction

- Used automated learning techniques to reverse engineer e.dentifier2
- Results in state machines
- Previously done for bank cards
e.dentifier2

- Developed by Todos (now Gemalto)
- EMV-CAP
- Can be used with or without USB
- With USB:
  - See-What-You-Sign
  - “the most secure sign-what-you-see end user device ever seen”
  - Good idea!
EMV-CAP

PIN

challenge

bitfilter(AC)

PIN

OK

GENERATE AC (challenge,...)

AC
Protocol e.dentifier2

Host PC → USB reader → Smartcard

ASK-PIN → Display shows "ENTER PIN" → User enters PIN

PIN OK → SIGndata-DATA number → SIGndata-TEXT text → Display shows text → User presses OK

User pressed OK → GENERATE-AC → g(ARQC) → ARQC → GENERATE-AC f(text, number) → AAC
Protocol \textit{e.dentifier2}

- \textbf{Host PC}:
  - ASK-PIN
  - PIN OK
  - GENERATE-AC $f(\text{text, number})$
  - $g(\text{ARQC})$

- \textbf{USB reader}:
  - Display shows "ENTER PIN"
  - User enters PIN
  - SIGNDATA-DATA \textit{number}
  - SIGNDATA-TEXT \textit{text}
  - Display shows \textit{text}
  - User presses OK
  - GENERATE-AC $f(\text{text, number})$
  - ARQC
  - GENERATE-AC $f(\text{text, number})$
  - AAC

- \textbf{Smartcard}:
  - VERIFY \textit{pin guess}
  - PIN OK
  - PIN OK
Protocol e.dentifier2

Host PC → USB reader → Smartcard

ASK-PIN

Display shows "ENTER PIN"

User enters PIN

PIN OK

SIGndata-DATA number

SIGndata-TEXT text

Display shows text

Generate-AC

Generate-AC $f(\text{text, number})$

ARQC

Generate-AC $f(\text{text, number})$

AAC

$g(\text{ARQC})$
Automated learning

• Used LearnLib
  • Implementation of adapted L* algorithm

• Complete Mealy machine

• Equivalence queries approximated
  • Random traces
  • W-method
Using automated learning

- Reverse engineering
  - Manual inspection of correctness and security
- Fuzzing or model-based testing
  - Use as basis for automated fuzz testing
- Formal verification
  - Use as basis for model checking
Automated reverse engineering

- Two different versions of the device
- Programmable smart card
  - All PIN codes accepted
  - Responses fixed
- Physical interaction needed
Robot

- Built using Lego
- Controlled by Raspberry Pi
  - 3 motors: OK, Cancel, digit
  - Power USB line
  - USB commands
Robot
Robot
Results
Model checking

- Converted output to labelled transition system
- Used model checker CADP
- Checked property in modal logic
  - Is valid cryptogram generated only after OK button is pushed?
- Resulted in an attack trace for the old device
Conclusions

- Automated learning techniques
  - Useful in security analysis for embedded devices
  - Can automatically find security vulnerabilities
  - Good excuse to play with Lego
Conclusions

- Automated learning techniques
  - Useful in security analysis for embedded devices
  - Can automatically find security vulnerabilities
  - Good excuse to play with Lego

Thanks for your attention!