MWR INFOSECURITY

Nils
Rafael Dominguez Vega

PinPadPwn

Black Hat 2012

25th July 2012

Agenda

Introduction
Chip&Pin
Practical EMV Testing
Case Studies (Demos)
Conclusion

Introduction

About Us

- Rafa:
  - Security Consultant for 6 years
  - USB research
  - Smart-card research
- Nils:
  - Head of Research @ MWR
  - PWN2OWN Winner 2009/2010
  - Android research

Why Research Payment Terminals?

- There wasn't as much money in Android Exploits as we thought ☺
- Widely used
- Payment Information Entry Point
- Payment authorisation
  - From Merchant Perspective
- More and more powerful – Larger attack surface
- Single Point of Failure
  - For Merchants
  - And Card Holders

Previous Attacks

- Terminal Skimming
  - Modifying the Hardware
- Replacing Terminals
- Manipulated Applications
  - Rogue Developers/Engineers

- “Understanding Terminal Manipulation at the Point of Sale” - MasterCard

Previous Attacks

WorkingBase

WorkingBase Projects | Third Party Projects | Post Project for Free | Find a Freelancer

Verifone, POS, Verix, Data Logger, Offline, Dummy, Firmware

Budget: $3000 - 5000 **Posted:** 1 year ago **Type:**

C++ Programming Cryptography Electrical Engineering Embedded Software Microcontroller

Need someone who can modify a VeriFone vx670 wireless pin pad, so that it will record track1, track2 and PIN # for debit, cc transactions. Need to print out receipt and need to have a menu for Debit and Credit. If card with chip is entered, it should say "error please swipe card".

This will be a dummy machine, if you know what that means. In order to do this software modification I believe it is necessary to have access to the Verix Developer Toolkit.

Please have a look at the attached notepad file for some more details with regards to this project.

Software Security

- Payment Terminals are small computers
  - Payment Applications
  - Same Vulnerabilities as in other Software

- Attack Surface
  - Magnetic Stripe
  - Chip&Pin
  - Communication
  - Serial
  - JTAG
  - Setup Menu

Research Approach

- Goal
  - **Find and Exploit Software Vulnerabilities**
- Complete Black-Box perspective
- Only using publicly available information
- Buying from eBay and “other providers”
- Using second hand terminal from retailers
  - Payment applications installed
  - Terminals configured
    - Refunds anyone?

MWR LABS

Our Lab

Common Setups

- Dumb Terminal
  - Connected to POS
- Terminals with Payment Application
  - From Vendor
  - Third Parties
  - Vendor modifications
  - Connectivity
    - To Internal systems
    - Third Party Payment Providers

Payment Terminal Ecosystem

Manufacturer

SDK & Cert

Developer

Provide Terminals

Develop Applications

Acquirer

Set up Terminals

Personalise Applications

Retailer

Chip&Pin

Chip&Pin

- Major improvement over Mag Stripe
- Widely implemented in Europe
- Offline and Online Payments
- Chip allows for better user authentication
  - More static data than Mag Stripe
  - “Signing of Payments”
    - Cryptogram
- PIN replaces signature and ID
- US about to adopt Chip&Pin

Smartcard 101

- Answer-to-Reset (ATR)
- Communication with APDUs
  - Application Protocol Data Units
- Hosts system sends commands
  - C-APDU
- Smart Card always only responds
  - R-APDU

Chip&Pin - EMV

- Short for Europay, Mastercard and Visa
- De-Facto Standard for Chip&Pin Payments
  - Also Gift Cards
- Contactless (NFC) Payments use EMV
  - Some older implementation don't
- Defines aspects such as
  - Multiple Card Applications
  - Data Storage (PIN, Expiry Date)
  - PIN Verification

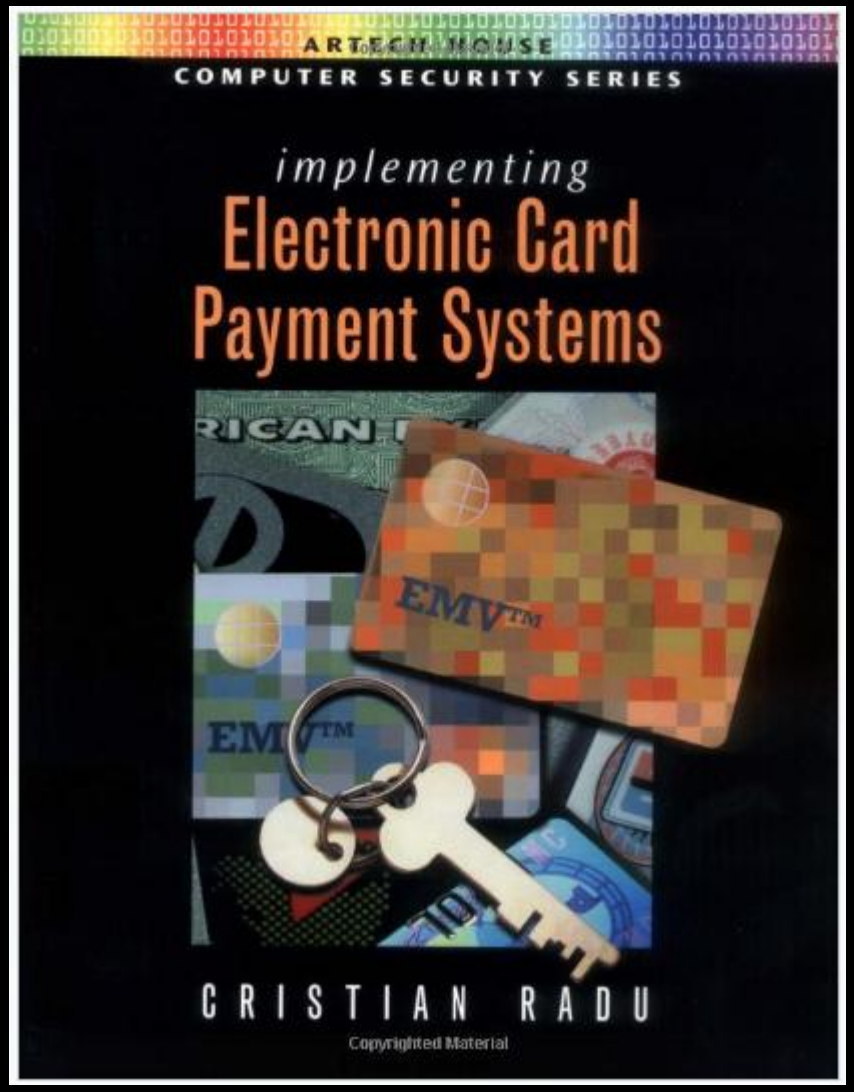
- Useful resource: *www.**emv**lab.org*

EMV Records

- READ RECORDS command reads EMV records
- TLV data format
  - Error Prone
- Example:

```
70 50 -- Record Template (EMV Proprietary)
      5f 24 03 -- Application Expiration Date
               12 03 31 (NUMERIC)
      5f 25 03 -- Application Effective Date
               09 02 05 (NUMERIC)
      5a 08 -- Application Primary Account Number (PAN)
            54 11 11 88 88 88 88 82 (NUMERIC)
      …
```

EMV – Further Reading



Practical EMV Testing

MitM Smart-card Sniffing

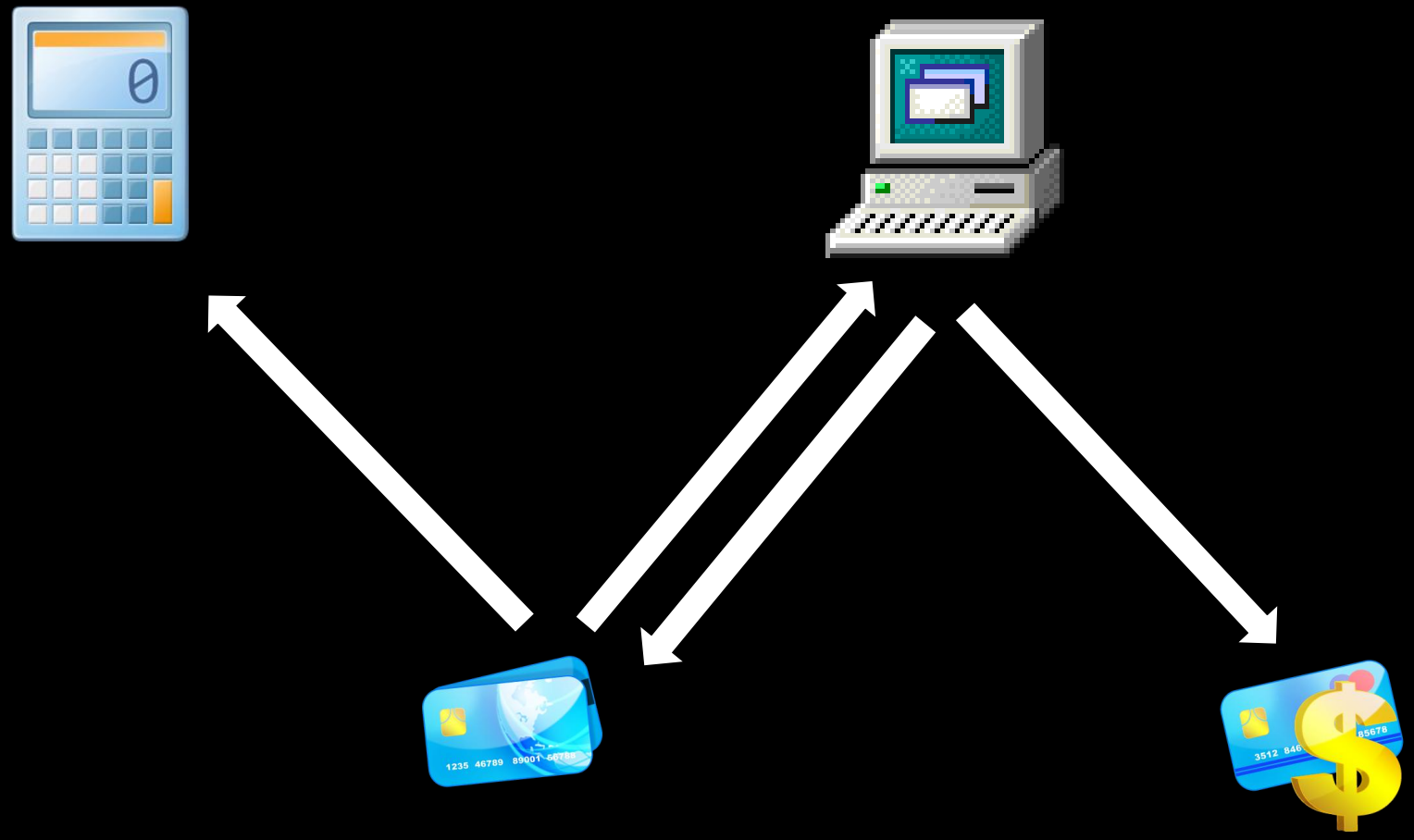
- Season 2 Board
  - Sat TV hacker toolkit
- Sits between the card and the terminal
- Allow sniffing of data via RS-232
- We got mixed results

- Other Hardware exists
  - Smart Card Detective

Logging/Programmable Smart-card

- Custom Smartcard
  - Log APDUs from Terminal
  - Programmable from our Scripts
    - Sequence of responses
- We used BasicCards
  - JavaCard IDE is impossible to set-up

Logging/Programmable Smart-card



Case Studies

Case Studies

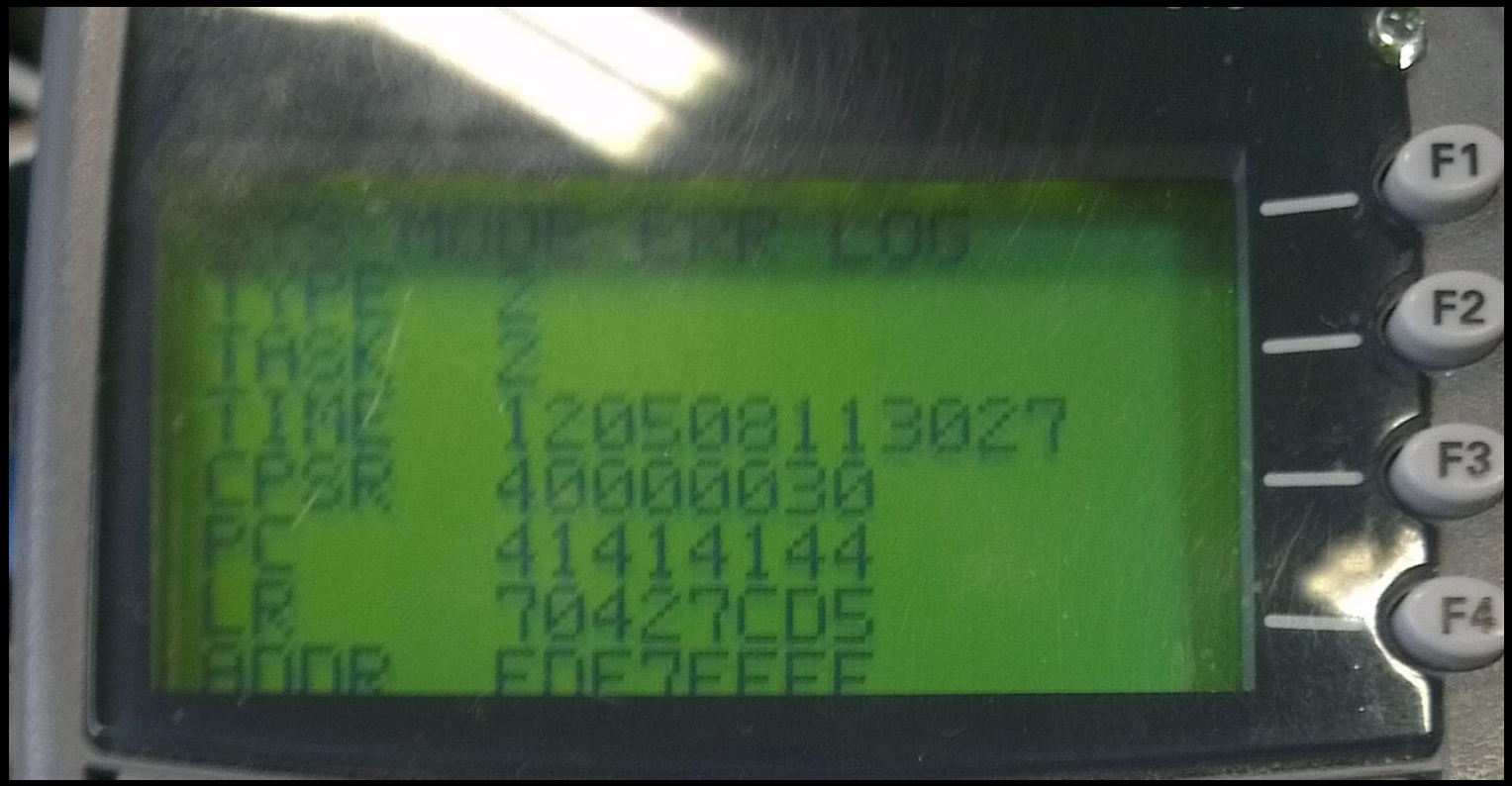
- 3 Case studies
- 3 Different terminals
- Vulnerabilities currently with vendors
- No vendor names
- Unfortunately
  - No specific vulnerability details

Case Study 1
Payment Terminal 1

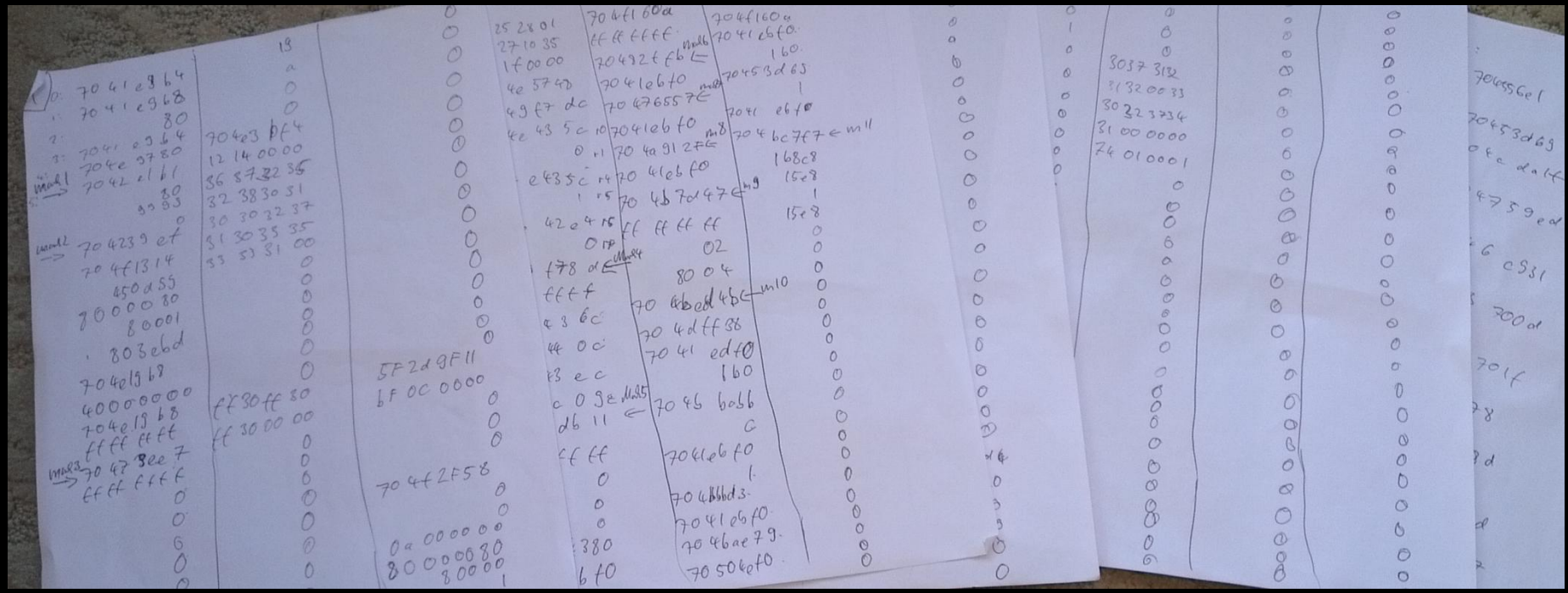
Case Study 1


F1
F2
F3
F4
ALPHA
QZ 1
2 ABC
3 DEF
GHI 4
5 JKL
6 MNO
PRS 7
8 TUV
9 WXY
,'"*
0 -SP
#
Cancel
Clear
Enter

First Vulnerability - Network



Low Tech Memory Dump



The OS and App

- Dump memory regions through ethernet
- Allowed us to analyse the Application
  - System calls

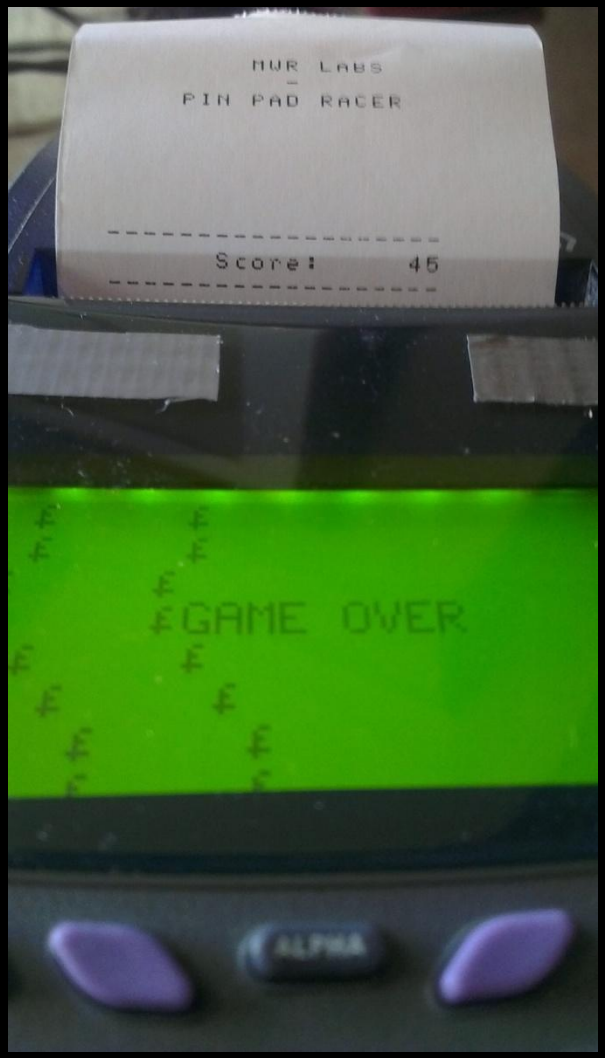
- And find more weaknesses
  - Hardcoded Passwords
  - Chip&Pin Vulnerabilities

Chip&Pin Vulnerability #1

- Fairly straight forward stack-buffer overflow
- Handling EMV tags
- Allows for arbitrary code execution
- Payload staging
  - ROP to retrieve more data from card
  - Shellcode to retrieve even more data
    - Almost arbitrary
  - 870 bytes final stage

PinPadPwn
DEMO #1

Demo #1


MWR LABS
PIN PAD RACER
Score: 45
GAME OVER
ALPHA


MWR
INFOSECURITY

Demo #1

- Code execution in context of payment application
- Reported to Vendor
- Printer and Display
- Anything could be done
  - Authorise Payments?
- More later ☺

Case Study 2
Feature Rich Payment Terminal

Case Study 2

Case Study 2

- Processor
  - 32 bit ARM
- Operating System
  - Embedded Linux
  - Even BusyBox
- User Interface
  - Touch Screen
  - Full colour display

Case Study 2

- Hardware Peripherals
  - Smart-card, SIM Card and Magnetic Card
  - Contactless
  - USB
  - Ethernet
  - RS-232

- Security Features
  - Binary Signing
  - Tamper Protection

Case Study 2

- Applications
  - Payment Application
  - Built-in Terminal Application

- Extra Functionality
  - Multimedia Advert Rendering
  - Remote Administrative Interface
  - Internet Access

MWR LABS

PinPadPwn
DEMO #2

Demo #2

- Full system compromise
- Running of our unsigned application
- Change root password and enable telnet

Case Study 3
Payment Terminal 3

Case Study 3


F1
F2
F3
F4
1 QZ.
2 ABC
3 DEF
4 GHI
5 JKL
6 MNO
7 PRS
8 TUV
9 WXY
*
0 -SP
#
Cancel
Clear
Enter

The OS and App

- Same custom OS as in Demo #1
  - That helped!
- Used on multiple devices
- Modified Vendor Application
- Code quality considerably better
- Still vulnerabilities
  - Deeper down the protocol
- Default and Hardcoded Credentials again …
  - “SuperMega” - Password

MWR LABS

Case Study 3

PinPadPwn
DEMO #3

The Cards

- Infecting the device:

- Retrieving CC# and PINs:

Demo #3

- Code execution in Context of Payment Application

- Reported to Vendor
  - Reported beginning of July
  - Patch exists already (< 3 Weeks)
  - Will take some time to make it to the terminals

Future Work

- OS Security
  - Privilege Escalation
  - Firmware Updates
    - Signing and Encryption
- Wifi, Bluetooth, Network Stacks
- More advanced Payloads
- Persistence on the Device
  - Some ideas
- NFC

Conclusion

- Too much trust into Payment Terminals
- Default Passwords were not changed
  - Sometimes Hardcoded Passwords
- Much effort into Physical Security Measures
  - Anti-tamper mechanisms
- Software vulnerabilities
  - Handling user controlled input
  - Memory corruption issues
  - Code injection


MWR
LABS