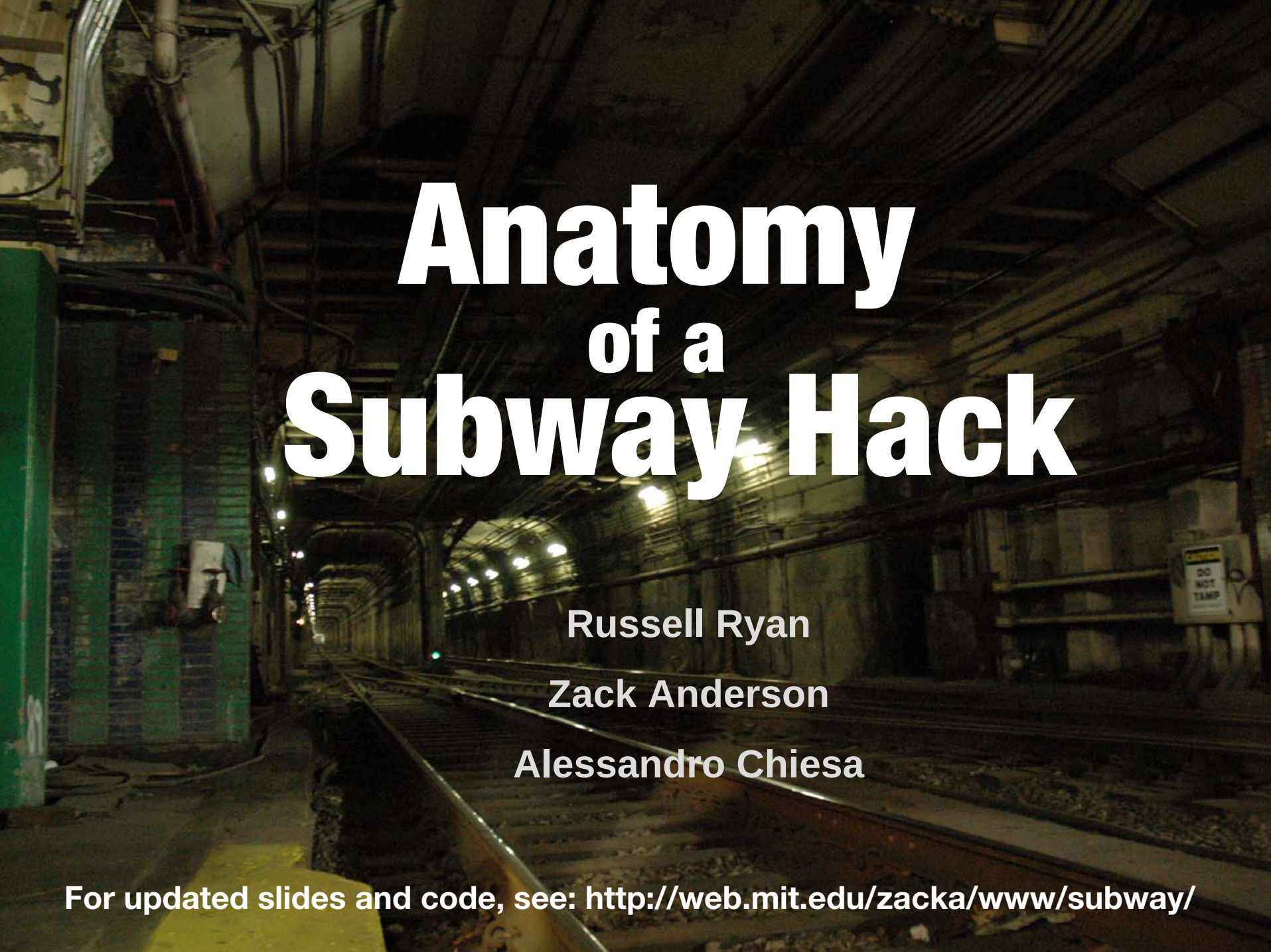



# Anatomy of a Subway Hack

Russell Ryan

Zack Anderson

Alessandro Chiesa

For updated slides and code, see: http://web.mit.edu/zacka/www/subway/

# what this talk is:

Pen-testing a subway system

# what this talk is not:

evidence in court
(hopefully)

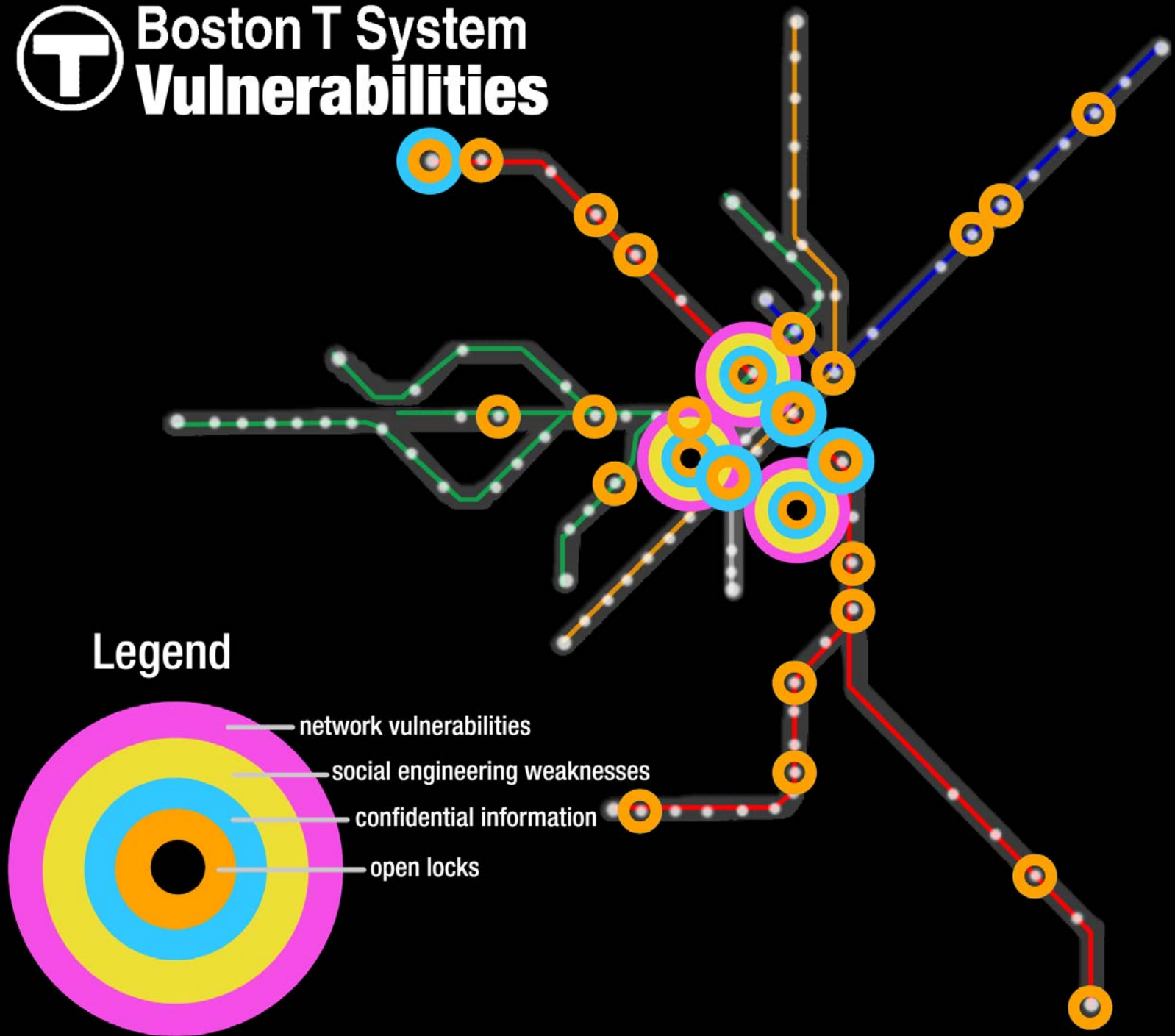
# You'll learn how to

- Generate stored-value fare cards
- Reverse engineer magstripes
- Hack RFID cards
- Use software radio to sniff
- Use FPGAs to brute force
- Tap into the fare vending network
- Social engineer
- WARCART!

# AND THIS IS VERY ILLEGAL!

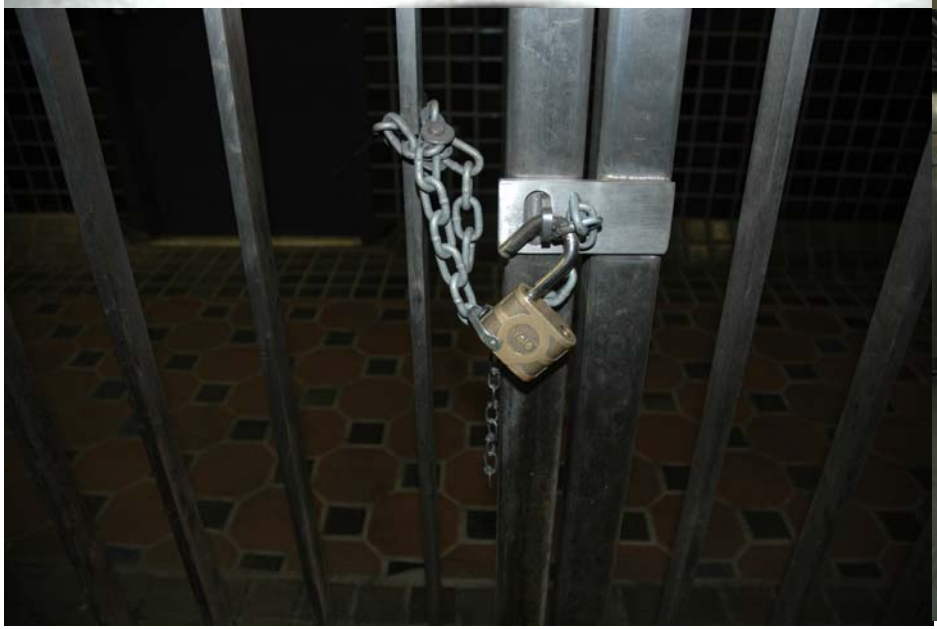
So the following material is for educational use only.

# Boston T System Vulnerabilities



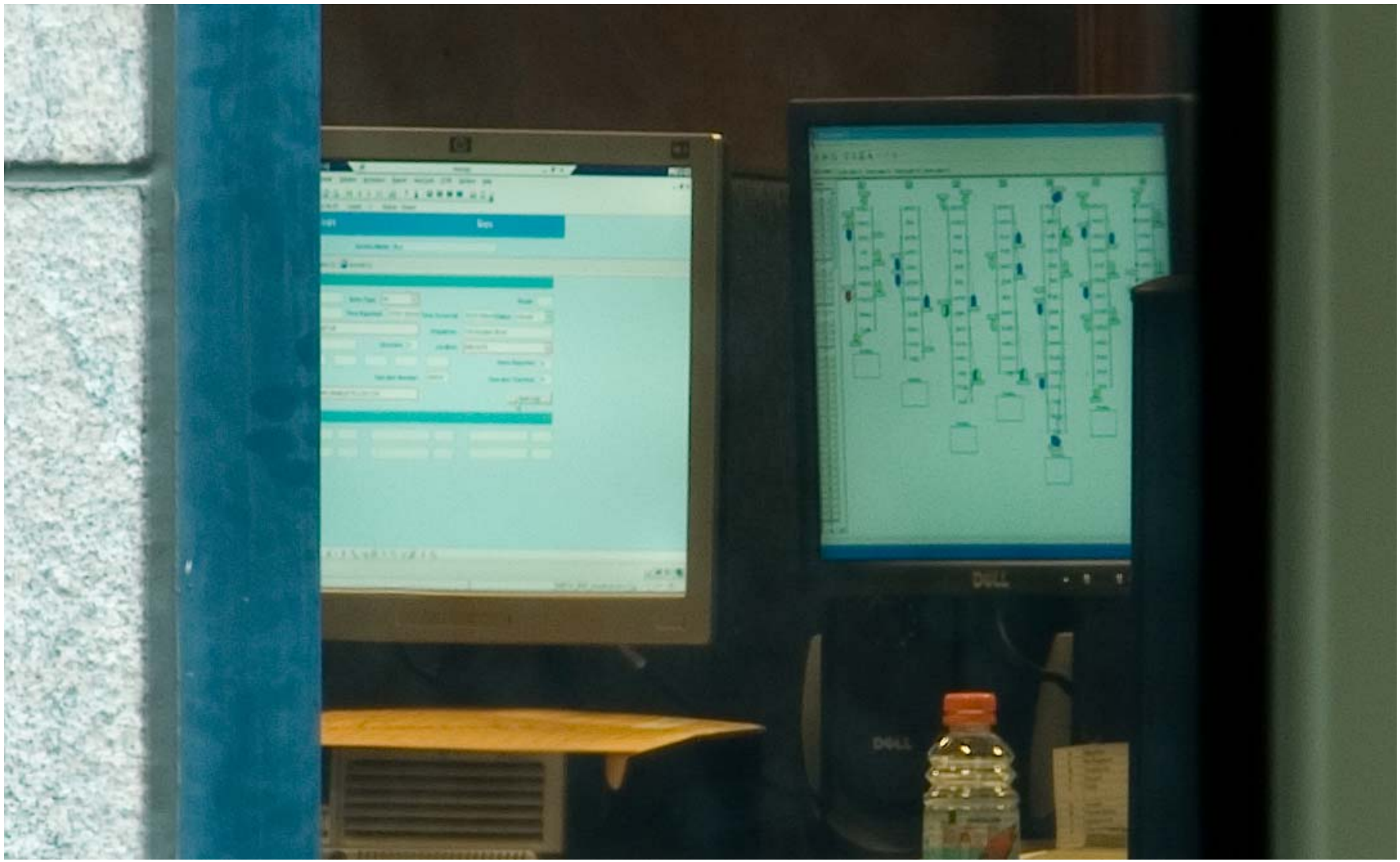

ATTACK
PHYSICAL
SECURITY

# there is almost always a free way to get in



# turnstile control boxes open… almost everywhere

# computer screens visible through windows

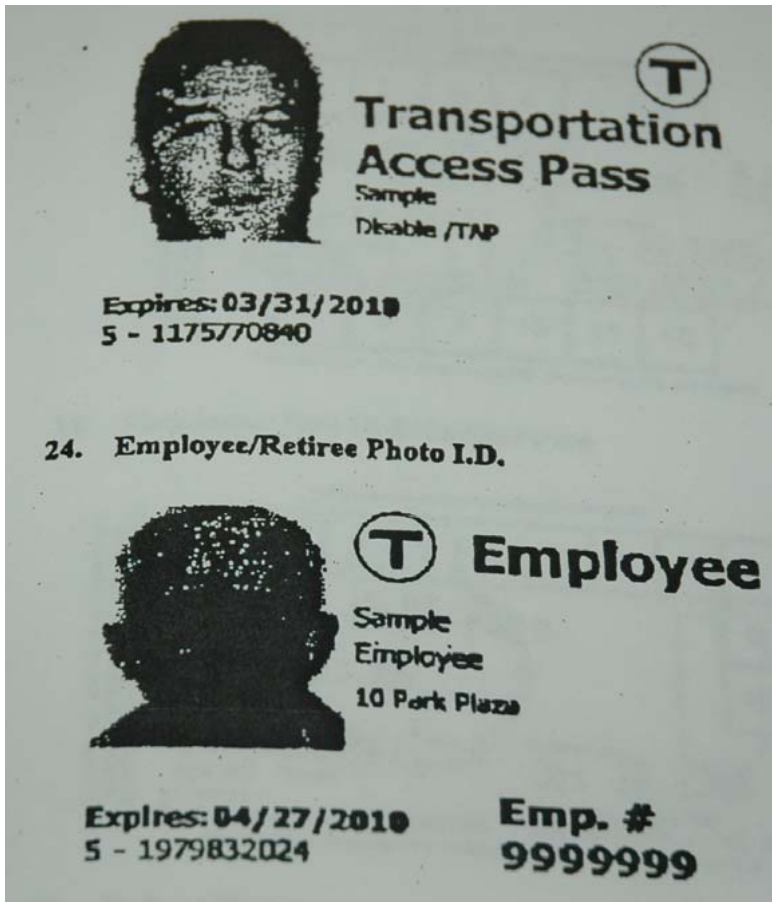


# door keys left in open boxes

# door keys left in open boxes

# state-of-the-art surveillance… often unattended

# documents left in the open



Transportation Access Pass
Sample
Disable /TAP

Expires: 03/31/2010
5 - 1175770840

24. Employee/Retiree Photo I.D.

Employee
Sample
Employee
10 Park Plaza

Expires: 04/27/2010
5 - 1979832024

Emp. #
9999999

| | | |
|---|---|---|
| Card/ticket does not work and CSA CANNOT verify value. | • CSA gives patron a refund ticket.<br>• Log refund on daily station report<br>• Customer given a Customer Claim Form to send with card/ticket to MBTA. | • Refund ticket |

# Employee

**Zackary Anderson**
**Director of Operations,**
**Red Line**
**10 Park Plaza**

**Expires 04/27/2010**
**5 - 1979832024**

**Emp. #**
**9358211**

Sign in

Categories ▾ | Motors | Express | Stores

Back to list of items  Listed in category: Computers & Networking > Printers

# Fargo DTC515 Thermal Card Printer

**Bidder or seller of this item?** Sign in for your status

1 of 2

View larger picture

Current bid: **US $79.99**

Your maximum bid: US $ [ ] Place Bid >

(Enter US $80.99 or more)

End time: **Jun-29-08 19:43:35 PDT** (2 days 1 hour)

Shipping costs: **US $30.12**
UPS Ground
Service to 02142, United States

Ships to: United States

Item location: Minneapolis, Minnesota, United States

History: 1 bid

High bidder: 1***o ( 804 ★ )

You can also: Watch This Item

Get SMS or IM alerts | Email to a friend

# what we found on Ebay




# ATTACK THE MAGCARD

# pick the hardware

**$5<**

Homebrew reader

With inserts, can read 3-tracks

stripesnoop.sourceforge.net

**$139.95**

Spark Fun Electronics

3-Track Lo-Co

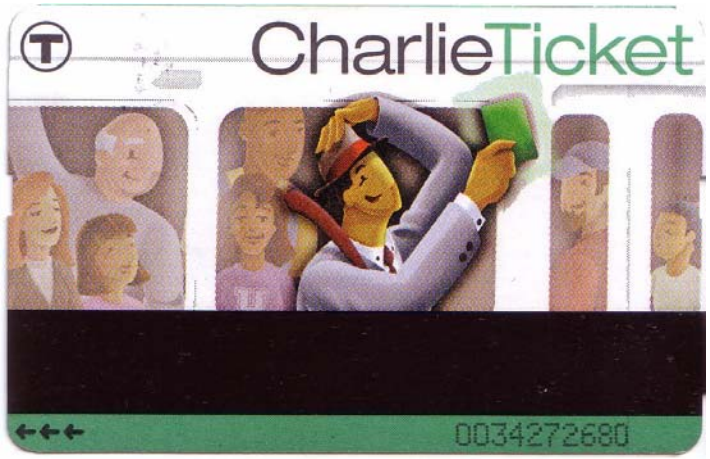
Includes source code

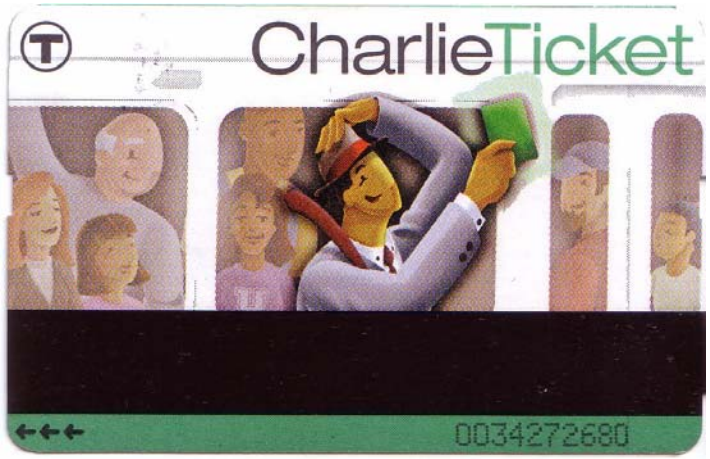

**$300**

MSR206 or MAKStripe

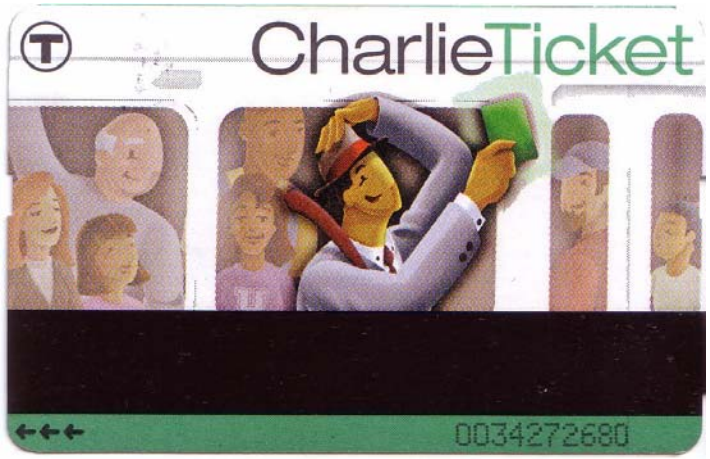
3-Track Hi/Lo-Co

Works with our GPL’d software


CharlieTicket
0034272680


CharlieTicket
0034272680


MSR206





```
EC9010402AC9D000000005B800C80150342248A
84EBD132BE10280002000000002025D0000FD60
```

Is value stored on the card?

**try a cloning attack**

If yes, then

**you now have free subway rides for life**

Oh,

**but you want more than that, eh?**

# reverse engineering

The Charlie Ticket

# reverse engineering

**Everybody talks about it,**
**But where do you start?**

1) Make a guess about what's in the data
2) Change a single variable; see what changes
3) Repeat many times with varying data
4) Compare similar and dissimilar data
5) Ignore constant regions
6) Build/use tools

# reverse engineering

## Isolate Variables method

To locate a single variable:

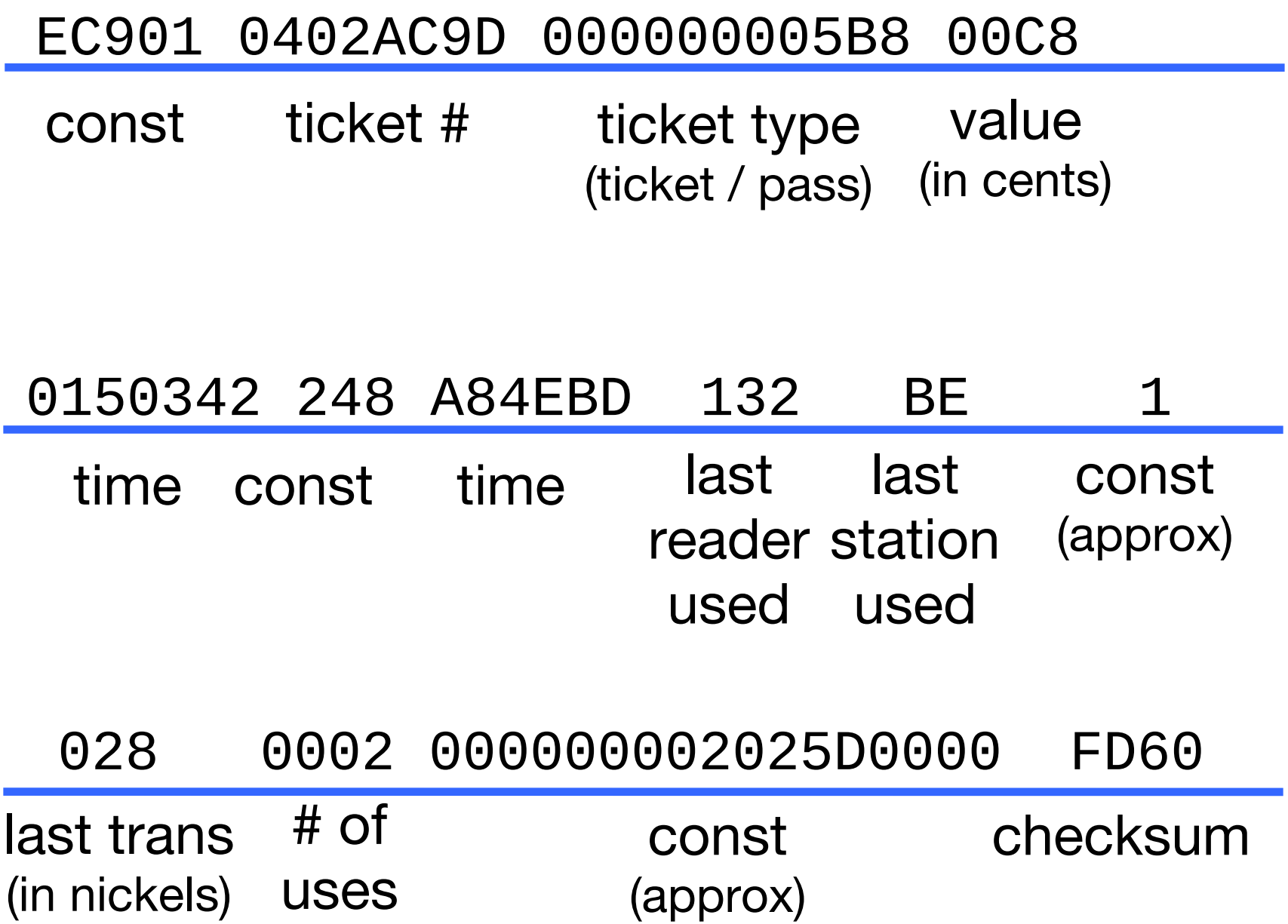
- Group data by that variable
- Ignore global similarities (between different groups)
- Ignore differences within groups

*Resulting locations are probably where the data is stored*

EC901 0402AC9D 000000005B8 00C8

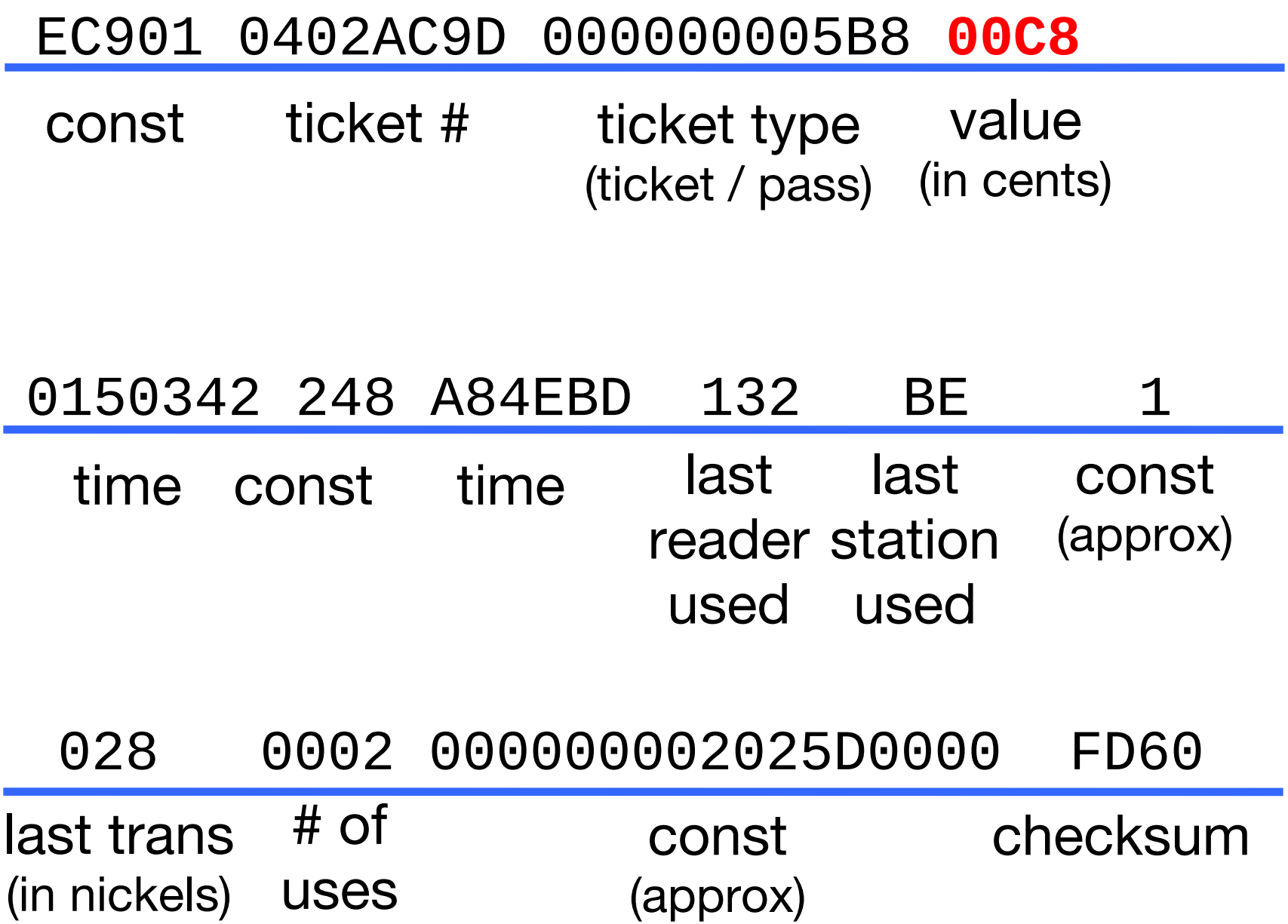
0150342 248 A84EBD 132 BE 1

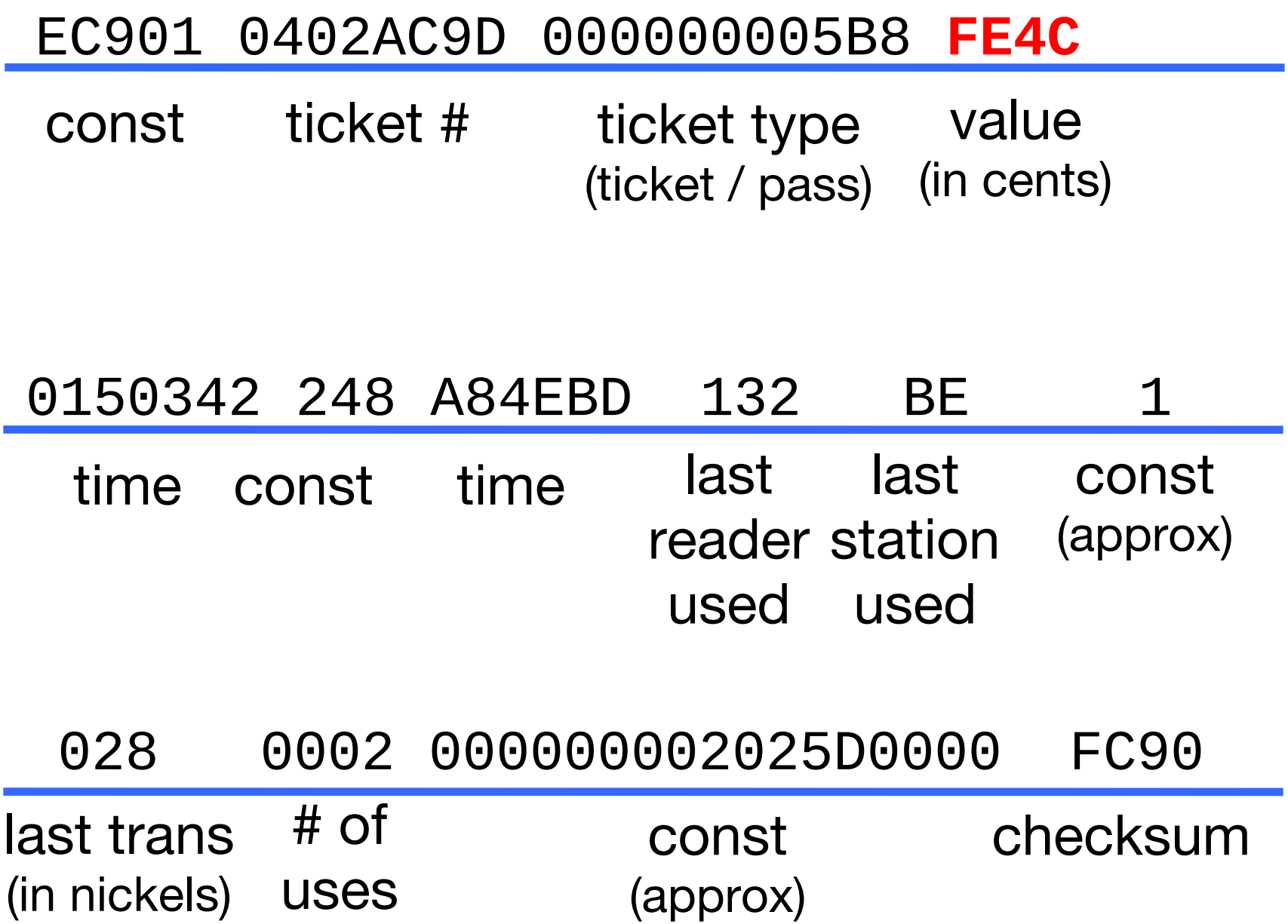
028 0002 000000002025D0000 FD60


EC901 0402AC9D 000000005B8 00C8
const
ticket #
ticket type
(ticket / pass)
value
(in cents)
0150342 248 A84EBD 132 BE 1
time
const
time
last reader used
last station used
const
(approx)
028 0002 00000000202 5D0000 FD60
last trans
(in nickels)
# of uses
const
(approx)
checksum

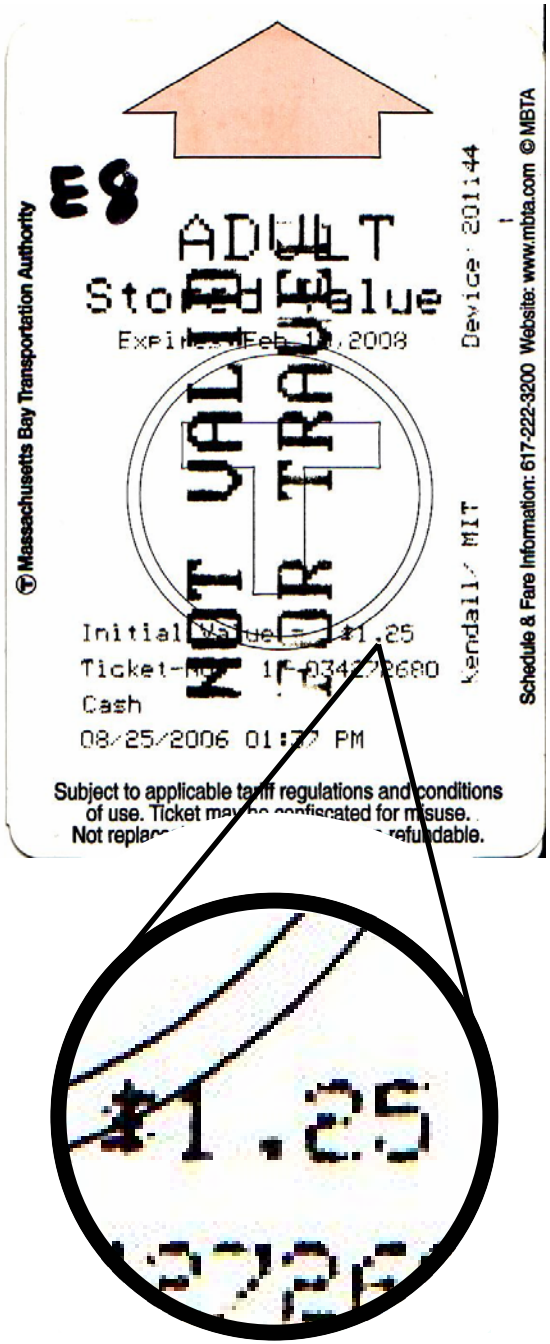
# forging

The Charlie Ticket


EC901 0402AC9D 000000005B8 00C8
const
ticket #
ticket type
(ticket / pass)
value
(in cents)
0150342 248 A84EBD 132 BE 1
time
const
time
last
reader
used
last
station
used
const
(approx)
028 0002 000000002025D0000 FD60
last trans
(in nickels)
# of
uses
const
(approx)
checksum


EC901 0402AC9D 000000005B8 FE4C
const
ticket #
ticket type
(ticket / pass)
value
(in cents)
0150342 248 A84EBD 132 BE 1
time
const
time
last
reader
used
last
station
used
const
(approx)
028 0002 00000000202 5D0000 FC90
last trans
(in nickels)
# of
uses
const
(approx)
checksum

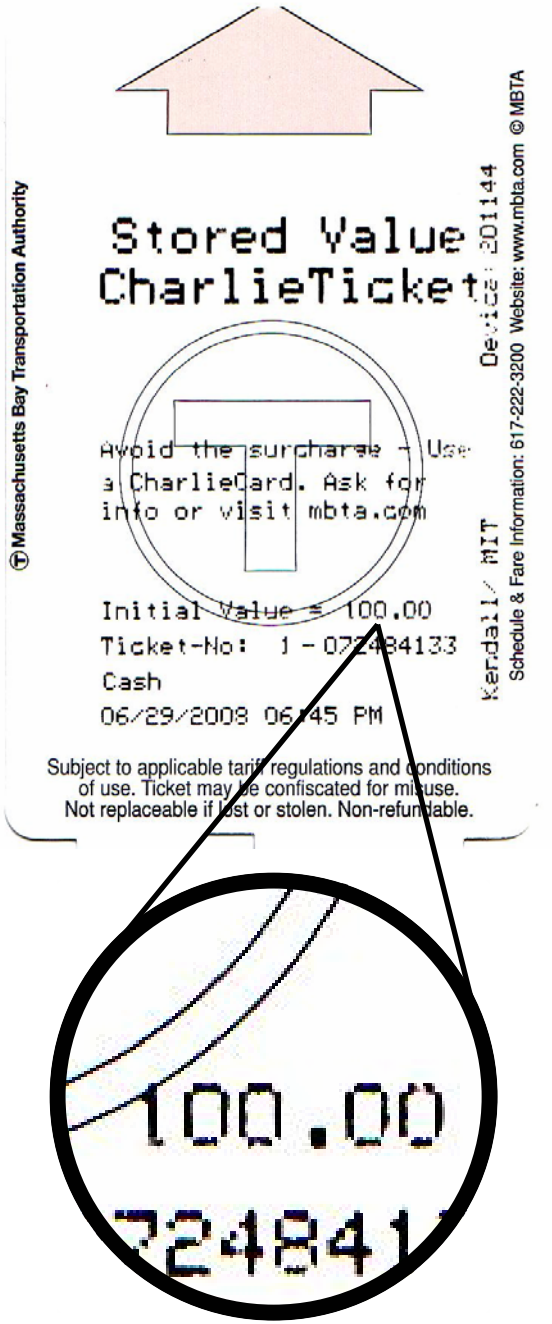

SV Adult
Remaining Amount: $ 653.00
Make Your Selection
Add Value
Card / Ticket Information
Go Back
Cancel
VISA
DEBIT CARDS


E8
ADULT
Stored Value
Expires Feb 12, 2008
NOT VALID FOR TRAVEL
Initial Value = $1.25
Ticket-No: 1 - 0342?2680
Cash
08/25/2006 01:?? PM
Subject to applicable tariff regulations and conditions of use. Ticket may be confiscated for misuse.
Not replaceable if lost or stolen. Non-refundable.
Ⓣ Massachusetts Bay Transportation Authority
Device: 201144
Kendall/ MIT
Schedule & Fare Information: 617-222-3200 Website: www.mbta.com © MBTA
$1.25

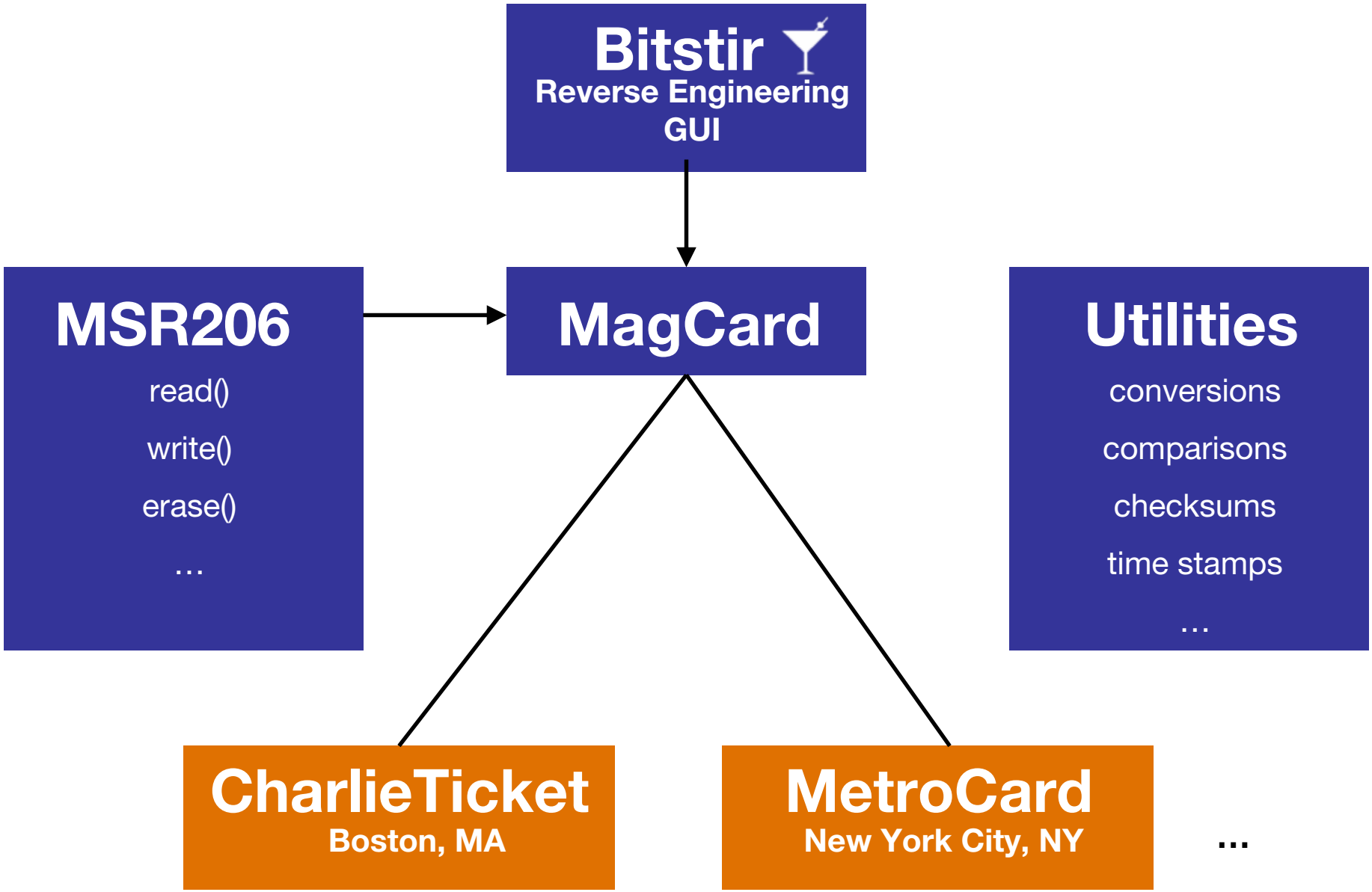
+


FEDERAL RESERVE NOTE
THE UNITED STATES OF AMERICA
ONE DOLLAR

=


Stored Value
CharlieTicket
Avoid the surcharge - Use a CharlieCard. Ask for info or visit mbta.com
Initial Value = 100.00
Ticket-No: 1 - 072484133
Cash
06/29/2008 06:45 PM
Subject to applicable tariff regulations and conditions of use. Ticket may be confiscated for misuse.
Not replaceable if lost or stolen. Non-refundable.
Ⓣ Massachusetts Bay Transportation Authority
Device: 201144
Kendall/ MIT
Schedule & Fare Information: 617-222-3200 Website: www.mbta.com © MBTA
100.00

# MagCard Reverse-Engineering Framework



# Demo: MagCard and Reverse Engineering Toolkit

- wrote Python libraries for analyzing magcards
- integrated with the MSR206 reader/writer
- GUI helps visualize and organize data

**Can Now Forge Cards**

# what about other subways?

- Most subway fare collection systems in US are made by two major integrators

- **Scheidt & Bachmann** made Boston T, San Francisco Bart, Long Island Railroad, Seattle Sound Transit, London Silverlink, etc. systems
- **Cubic Transportation** made NYC MTA, Washington DC WMATA, Chicago CTA, Shanghai Metro, etc. systems

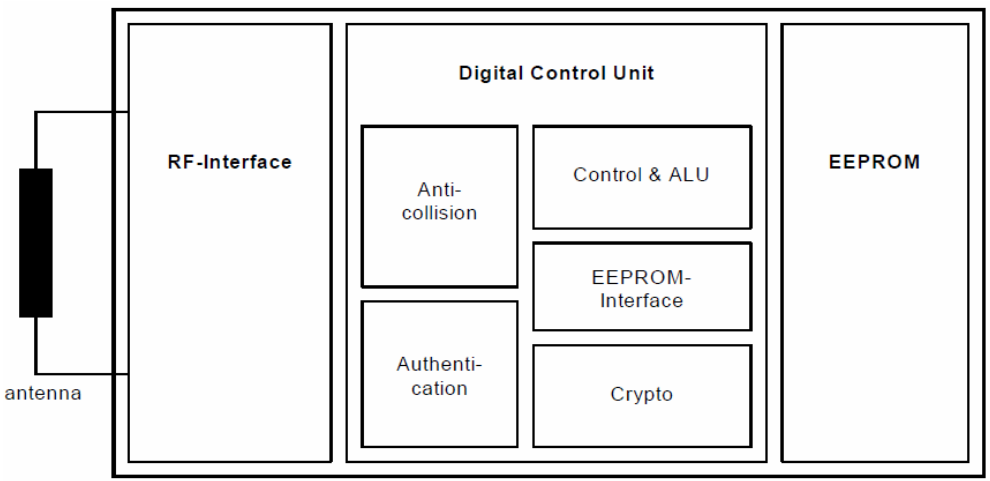
Are they hackable? Yes!

# ATTACK THE RFID

# learn about your RFID card

# MIFARE Classic

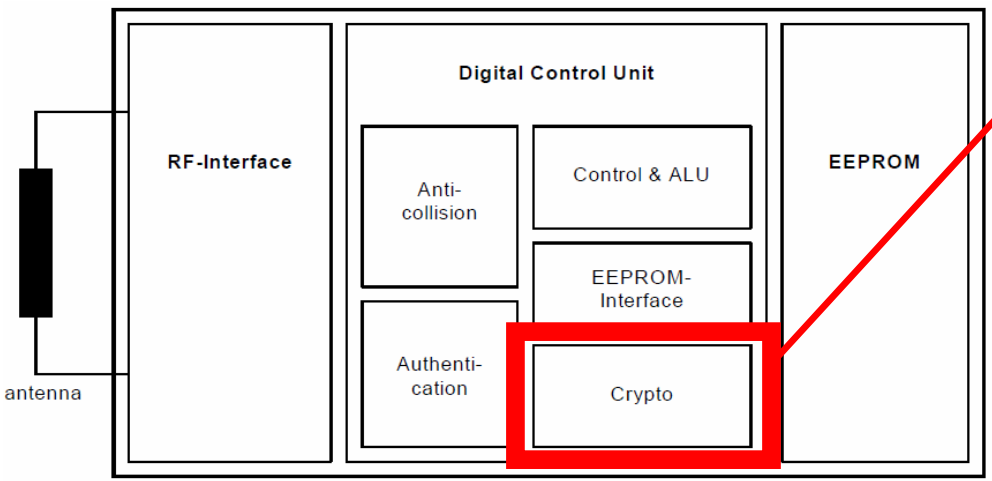
- 13.56MHz RFID smartcard
- End-to-end proprietary “crypto” (Crypto-1)
- 1K memory & unique identifier on card
- Over 500 million tags in use



# Crypto-1 Cryptanalysis

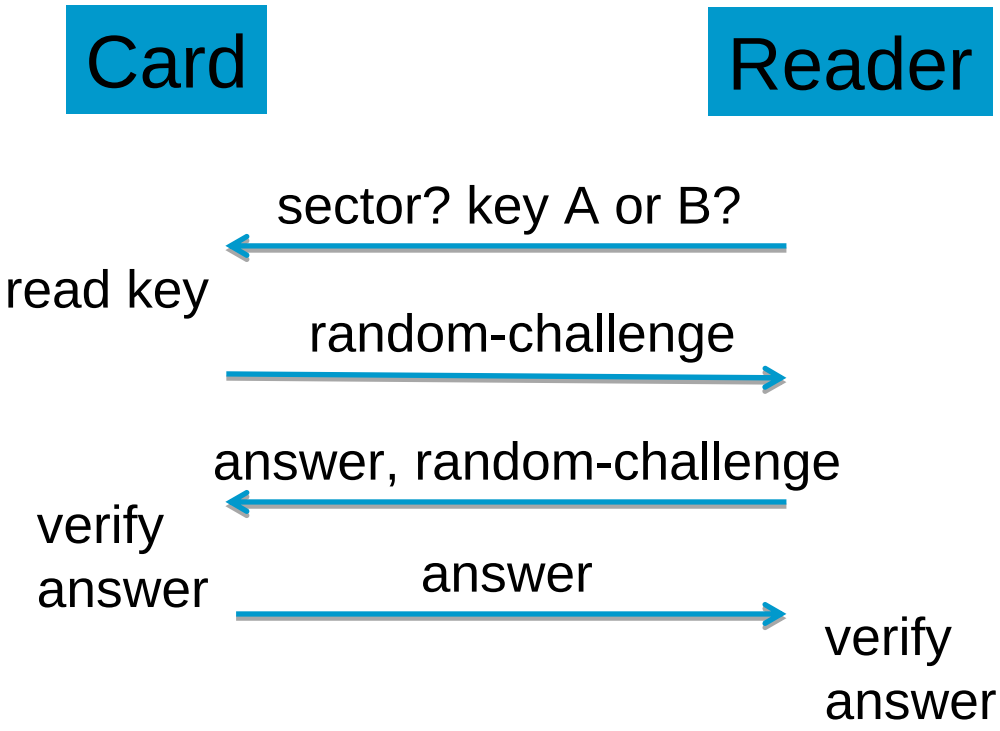
Crypto-1 reverse engineered by Karsten Nohl, University of Virginia, 2007:

- Etched and inspected silicon wafer using high-powered imagery.
- Found and reconstructed crypto portions from over 10k gates.
- Found vulnerabilities in the cipher and implementation



# security of the MIFARE card

Mutual 3-pass authentication

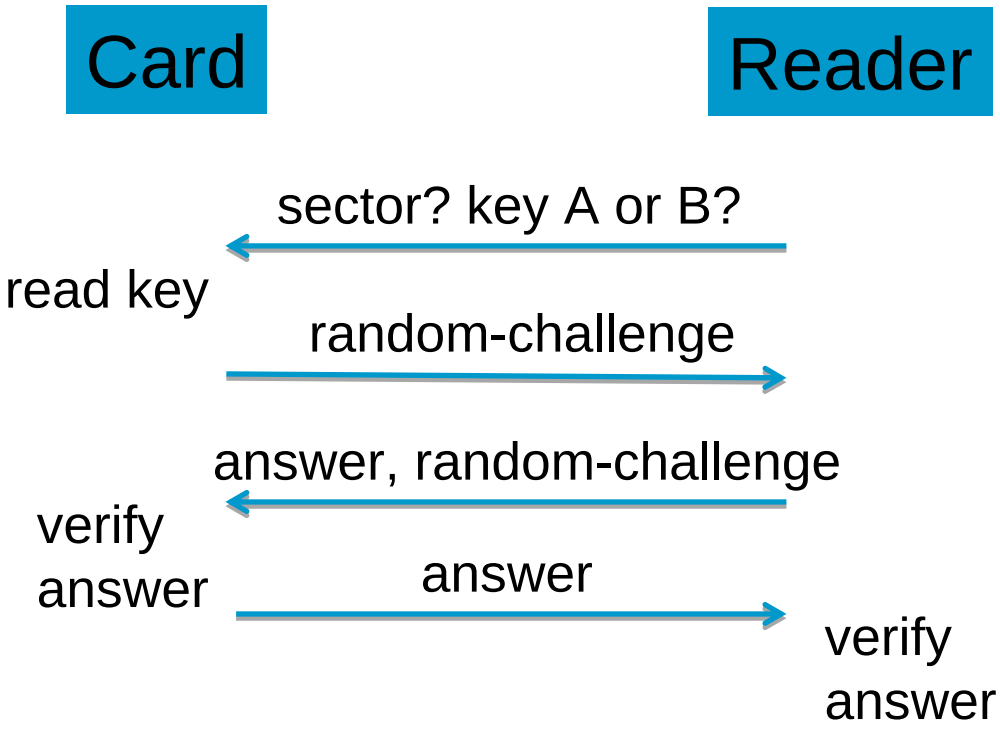



Each sector two keys

Non-linear filter functions

# security of the MIFARE card

Mutual 3-pass authentication




KEY IS 48bits!

Non-linear filter functions

# security of the MIFARE card

KEY IS 48bits!

PRG IS WEAK!

Non-linear filter functions

# security of the MIFARE card

PRG IS WEAK!

KEY IS 48bits!

BIASED Filter Functions

to execute these attacks we need to interact with the card

# choose your hardware

## $50

MiFare RFID Reader/Writer

Comes with source code

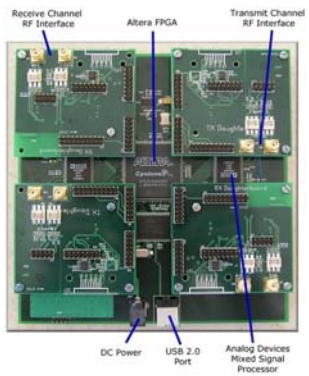
Hard to hack, but doable

## $220

OpenPCD + OpenPICC

Open design 13.56MHz RFID reader + emulator

Free schematics (www.openpcd.org)



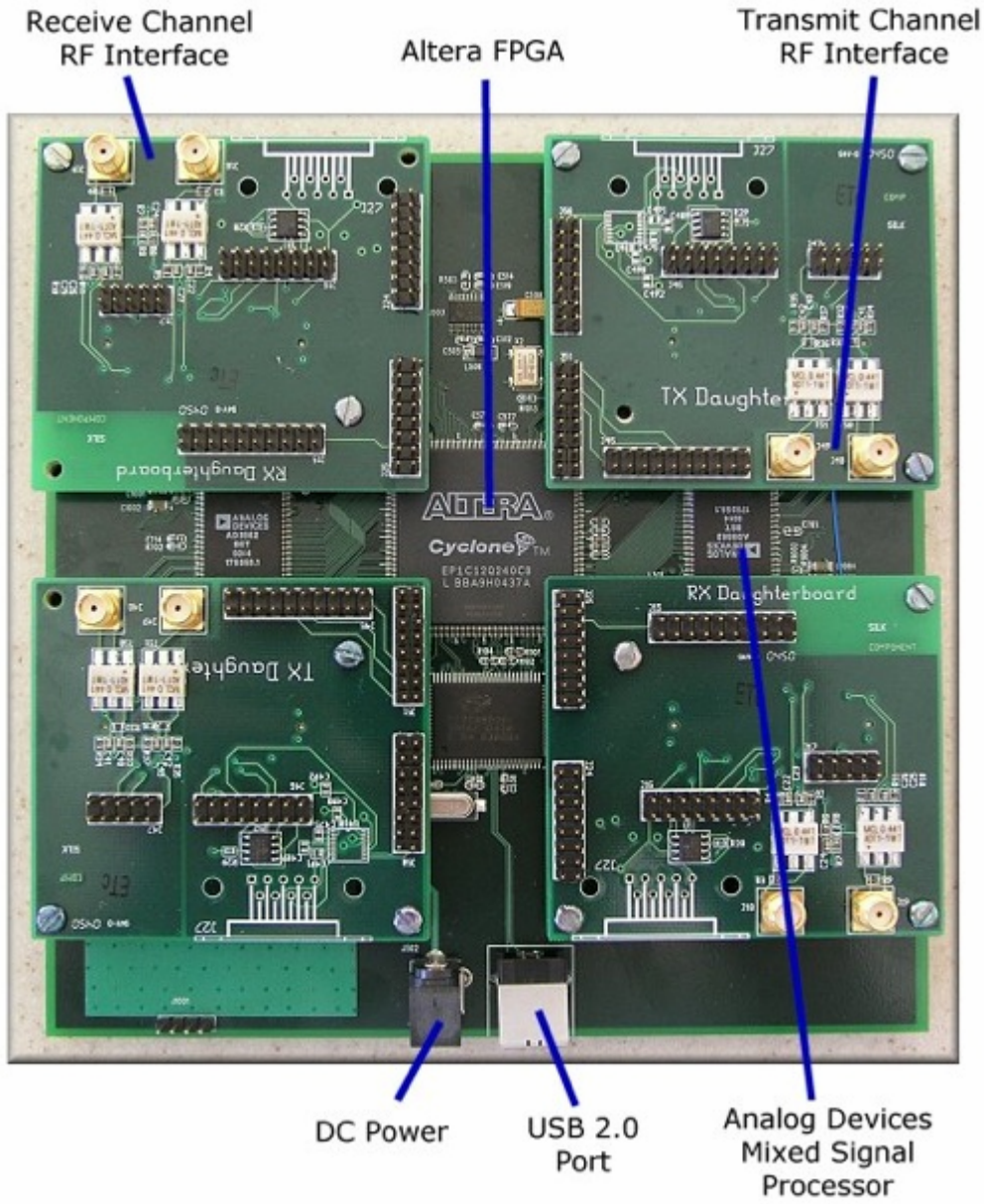

## $700

USRP

Full control over signal input/output

Works with GNU Radio + our plugin

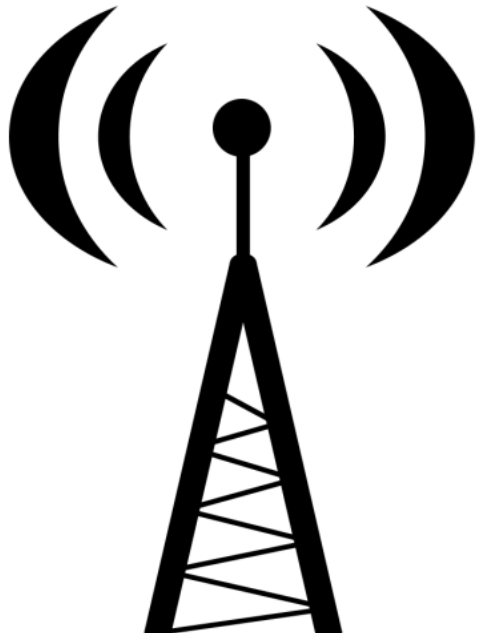
# USRP


Receive Channel
RF Interface
Altera FPGA
Transmit Channel
RF Interface
DC Power
USB 2.0
Port
Analog Devices
Mixed Signal
Processor

# card/reader communication


**13.56MHz**
ASK modulation
Modified miller encoding

**13.56MHz**
**+/- 847kHz**
OOK modulation
Manchester encoding



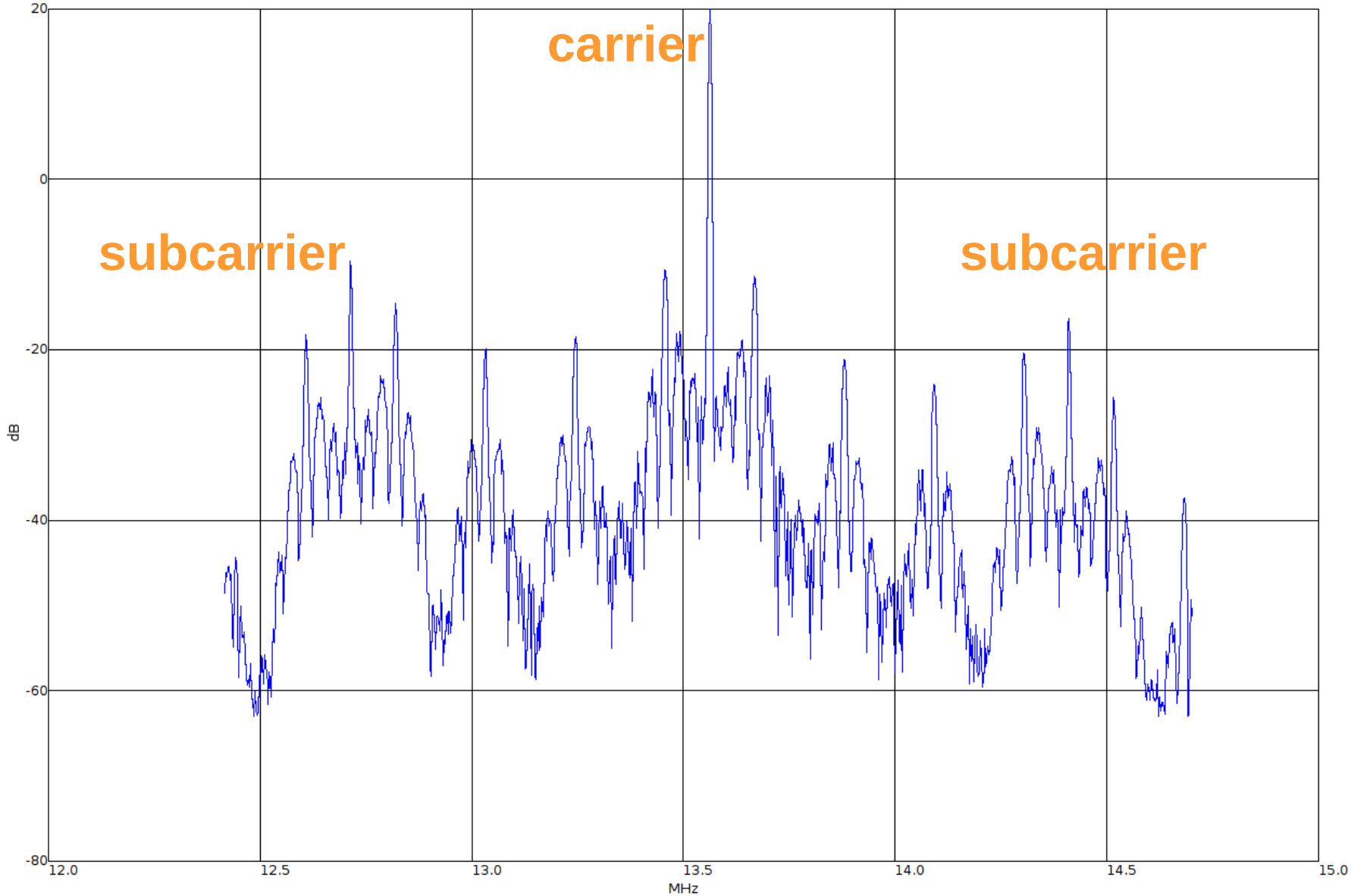
# GNU radio RFID toolchain

**Tune Radios**

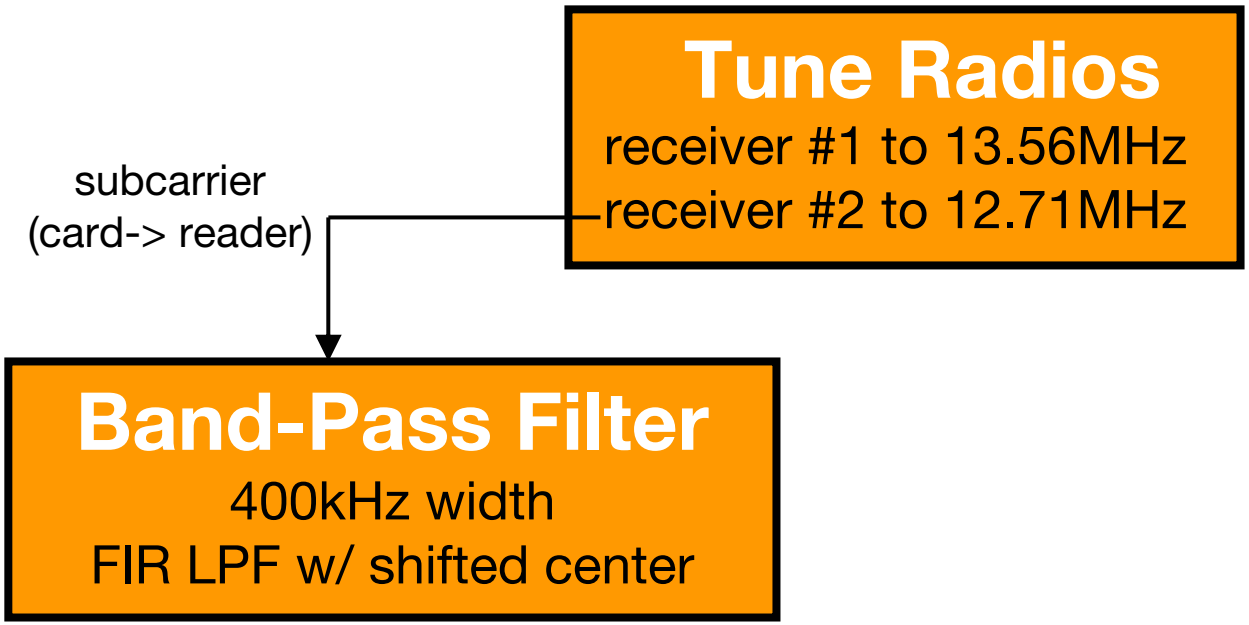
receiver #1 to 13.56MHz

receiver #2 to 12.71MHz

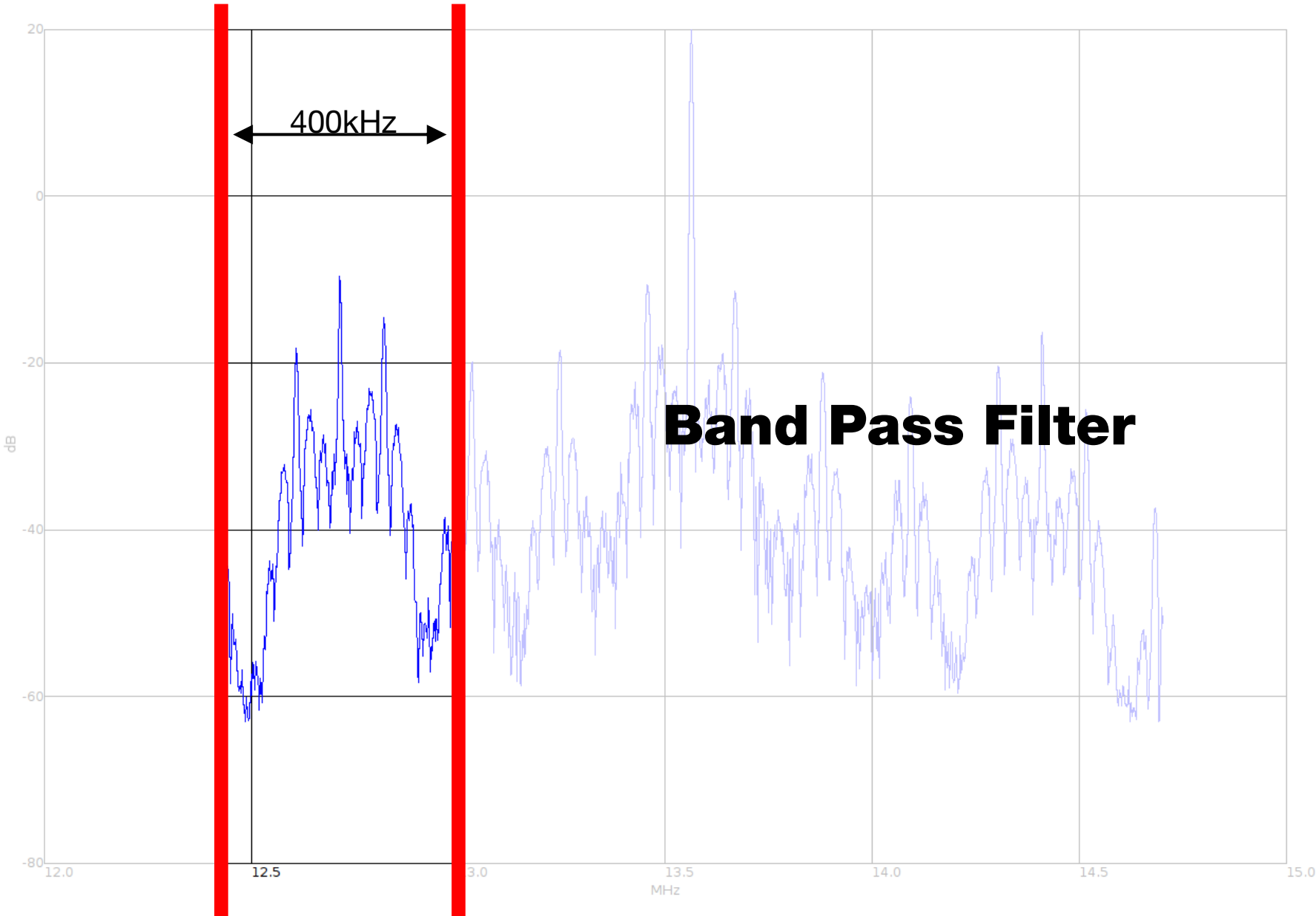
# charlie card + reader FFT



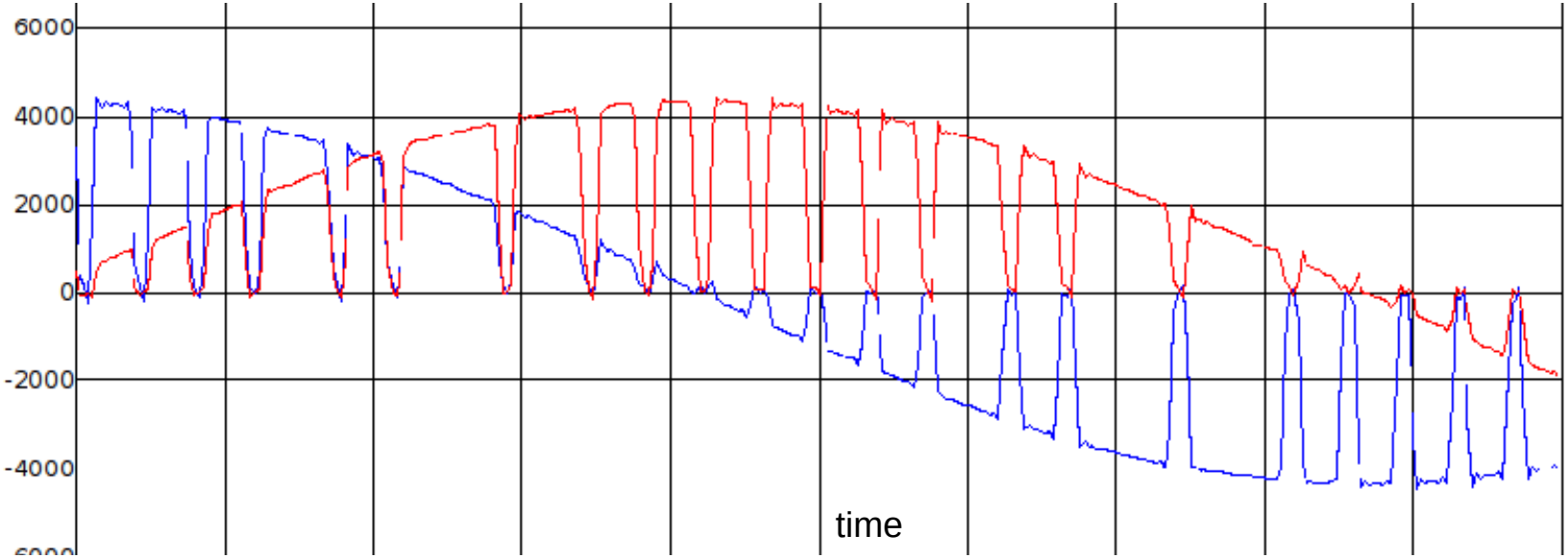
# GNU radio RFID toolchain



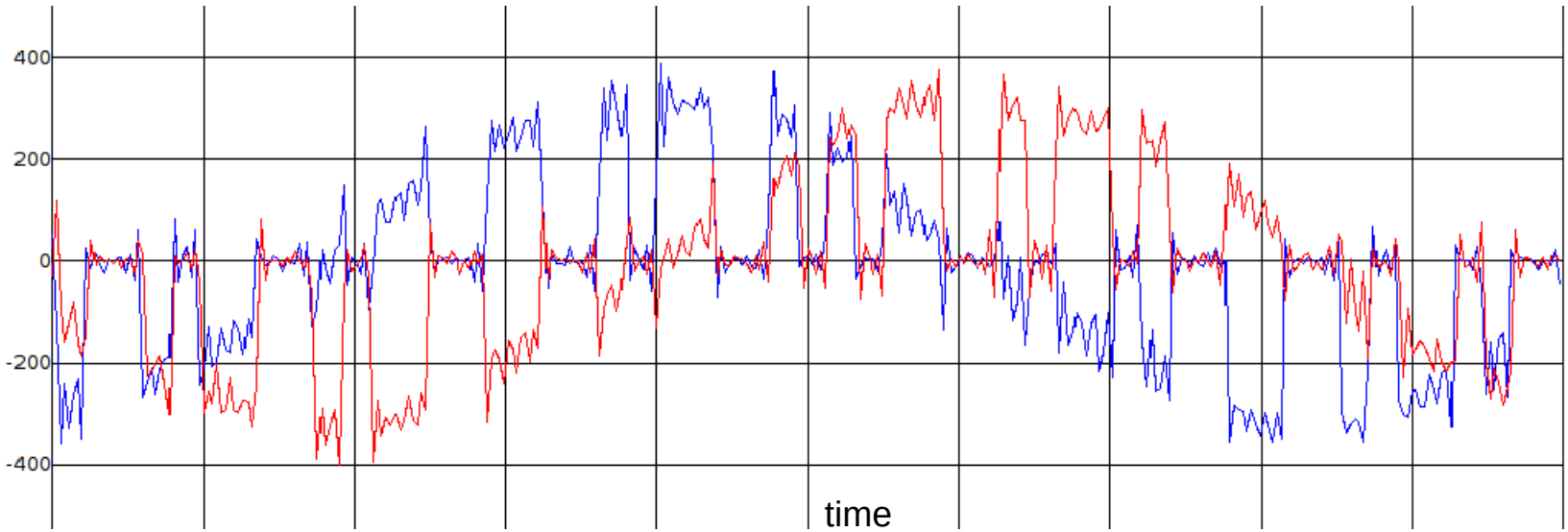
# charlie card + reader FFT



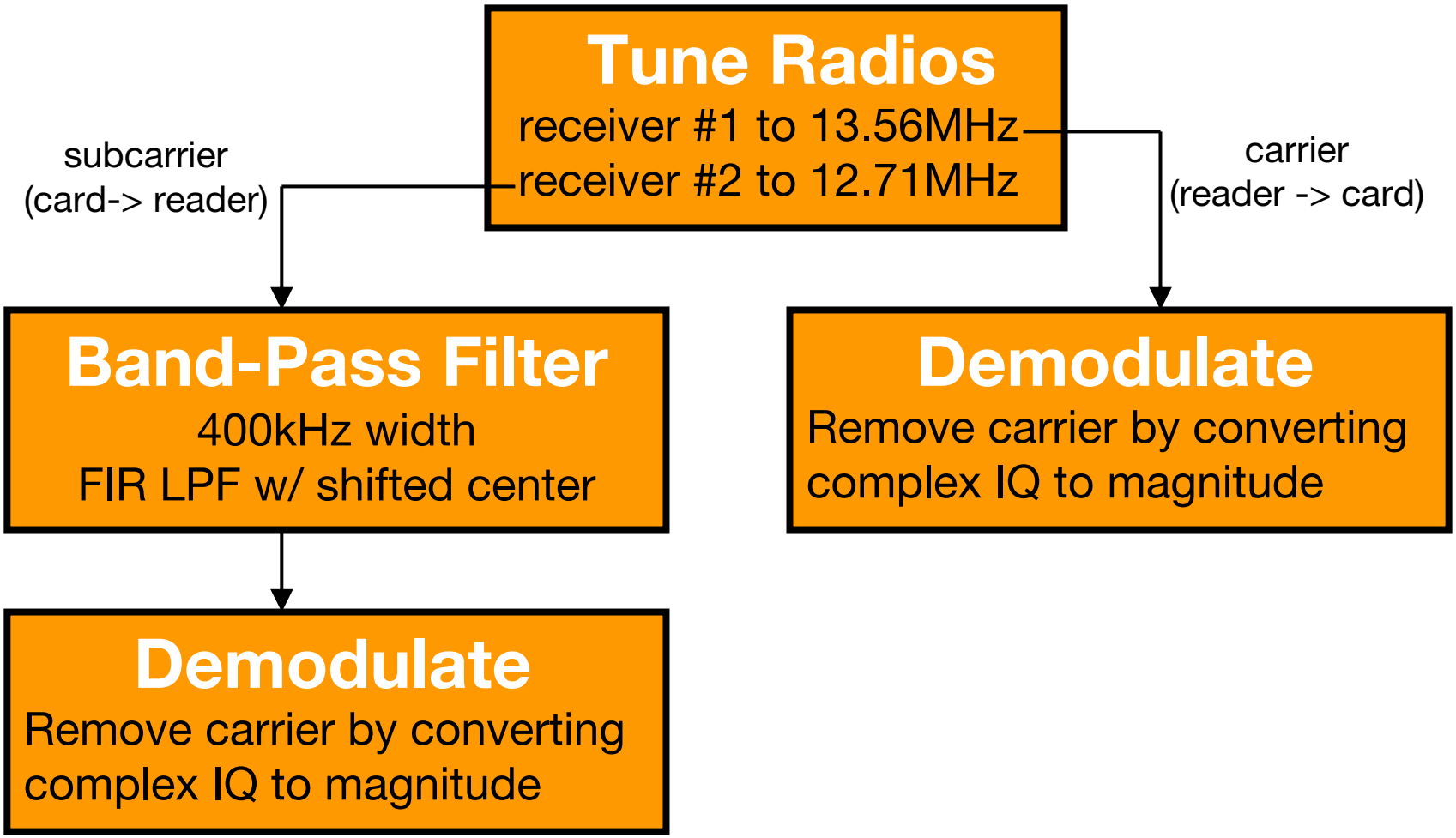
# 13.56MHz reader -> card transmission



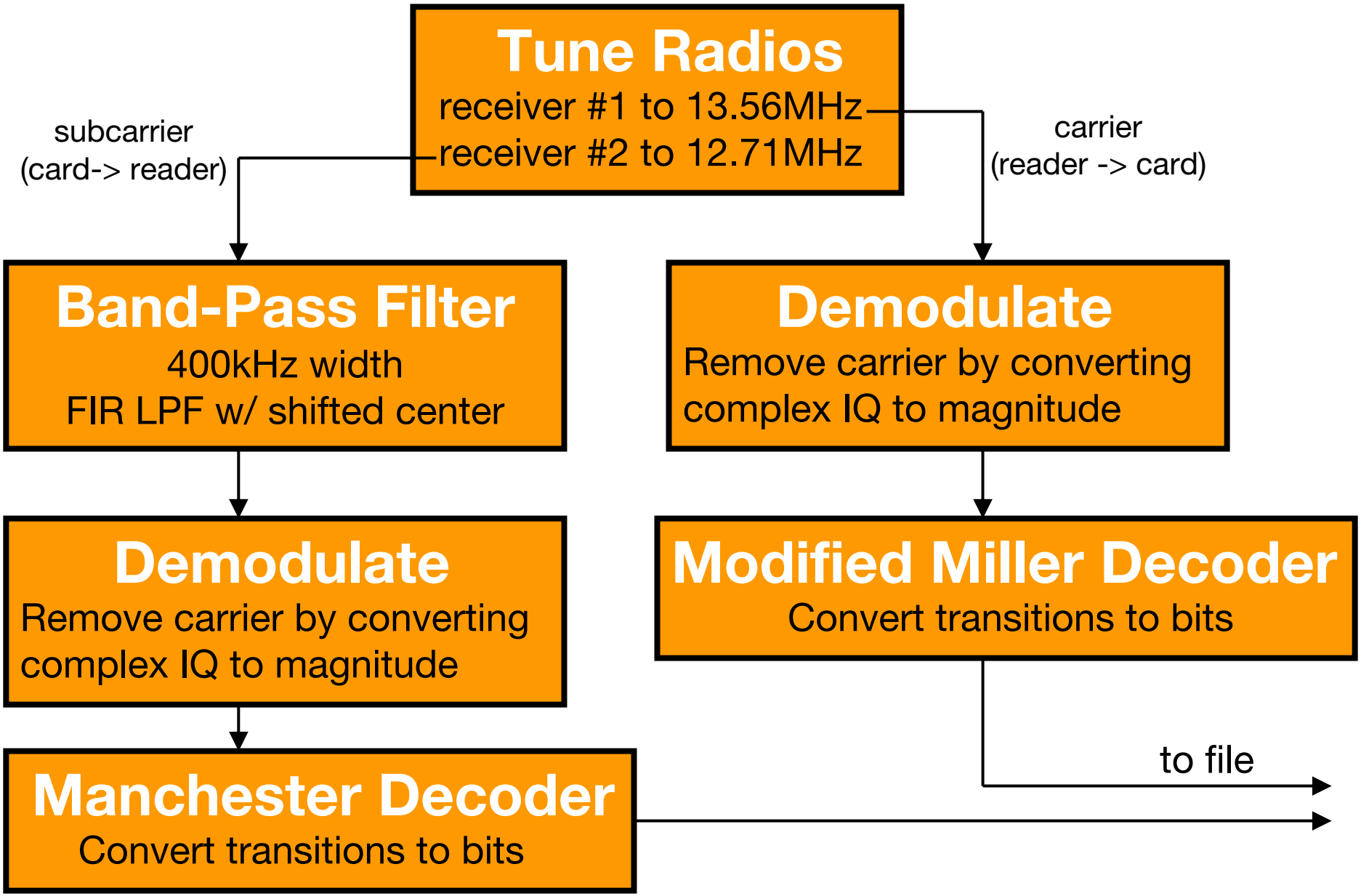

# 12.71MHz card -> reader transmission



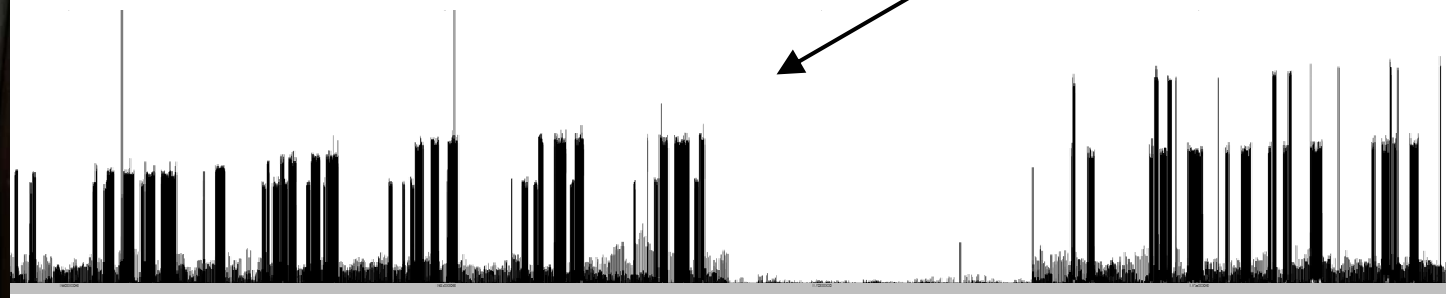
# GNU radio RFID toolchain



# GNU radio RFID toolchain




challenge/response pairs



# sniffing the turnstile

# attacks on the MIFARE card

Goal: get secret key (can clone card with it)

## Brute Force

sniff handshake and use an FPGA to crack key.

**Filter function weaknesses reduce key space.**

See:

www.cs.virginia.edu/~kn5f/Mifare.Cryptanalysis.htm

For info on reducing key space

# attacks on the MIFARE card

Goal: get secret key (can clone card with it)

## Brute Force

sniff handshake and use an FPGA to crack key.
**Filter function weaknesses reduce key space.**

## Manipulate PRG Timing

"random" challenge depends on clock cycles since powered up – thus **it is not random**.

*This enables replay attacks:*
Timing allows selection of specific challenges. With deterministic challenges, data can be replayed.

**Keep on transmitting those "add $5" commands**

# attacks on the MIFARE card

Goal: get secret key (can clone card with it)

## Brute Force

sniff handshake and use an FPGA to crack key.
**Filter function weaknesses reduce key space.**

## Manipulate PRG Timing

"random" challenge depends on clock cycles since powered up – thus **it is not random**.

## Algebraic Attacks

write Crypto-1 as system of multivariate quadratic equations combined with sniffed data, convert to SAT and then solve it with a SAT-solver… currently being worked on by Courtois, Nohl, and O'Neil

when all else fails

**brute force it**

# Why Brute Force with an FPGA?

## Because it's fast!

microprocessor

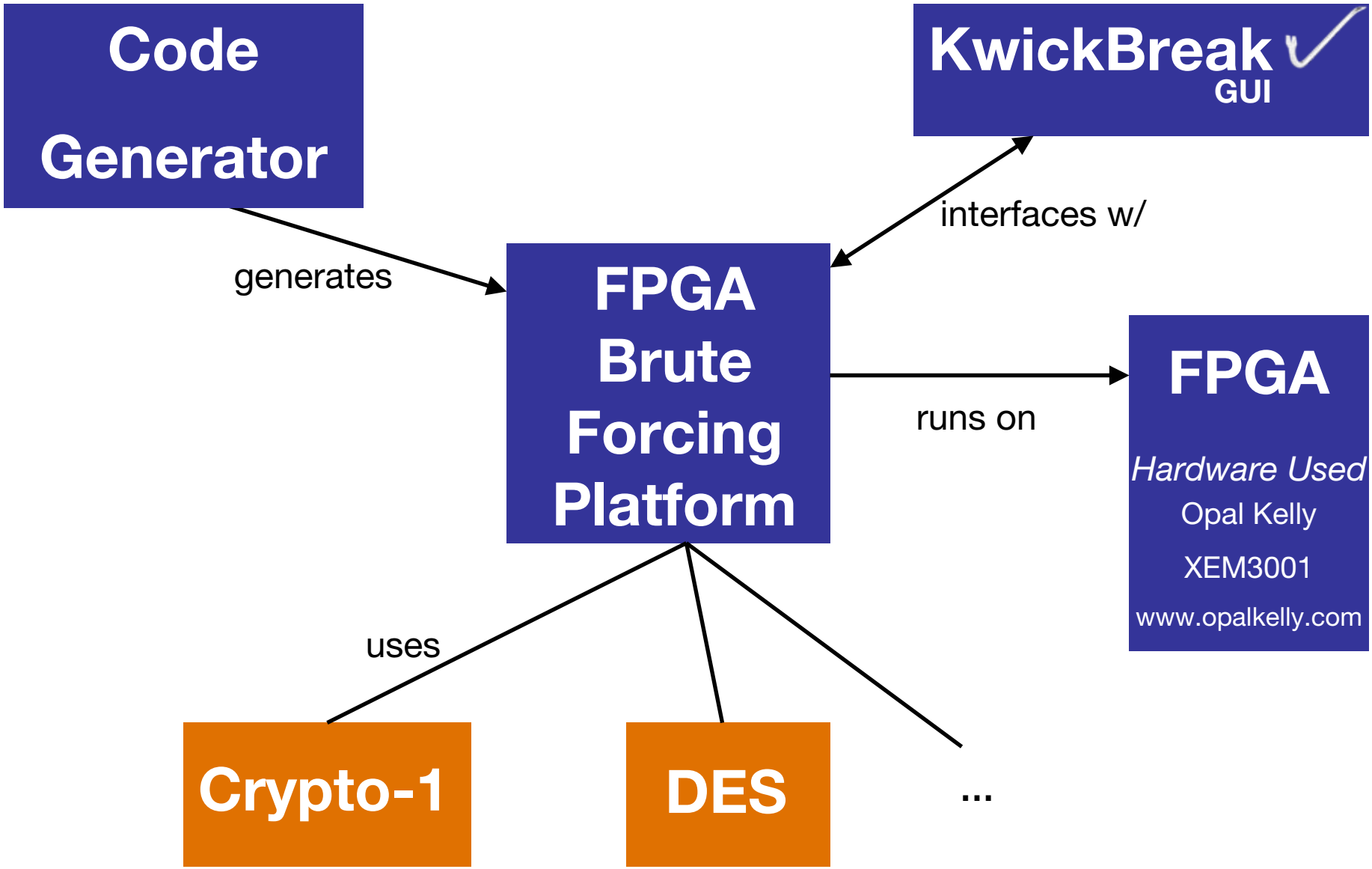

- General purpose device
- Finite instruction set
  (Uh, oh. Sounds RISCy)
- 1-8 parallelizations

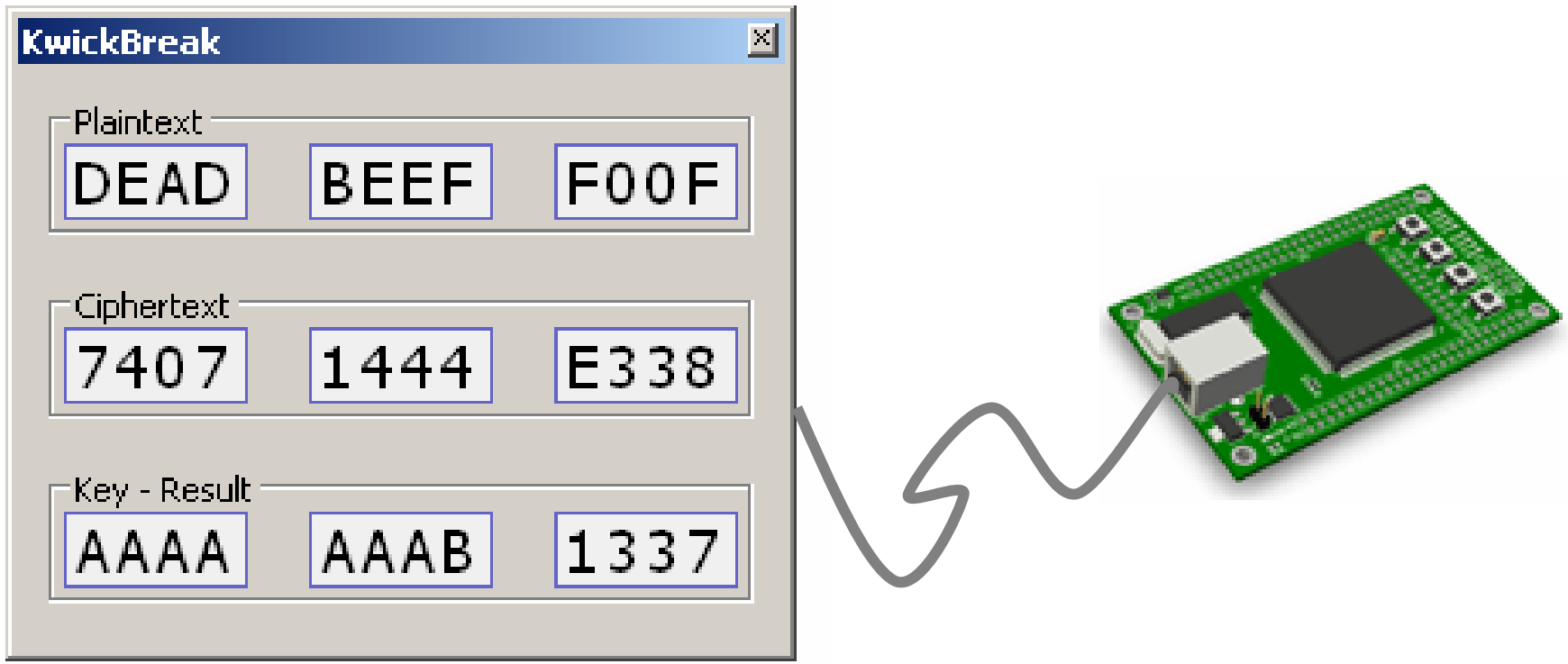
FPGA


- Dedicated logic
- Hardware description language defines hardware
- Hundreds of parallelizations

# KwickBreak FPGA Brute-Forcer

Executes known plaintext attack to recover key




KwickBreak
Plaintext
DEAD
BEEF
F00F
Ciphertext
7407
1444
E338
Key - Result
AAAA
AAAB
1337

# writing a (trivial) XOR module

```
module xorPlugin(
        input  wire clk,
        input  wire [47:0] key,
        input  wire [47:0] plaintext,
        output reg [47:0] encrypted,
        output reg ready);

always @(posedge clk) begin
        ready <= 1;
        encrypted <=  key ^ plaintext;
end
endmodule
```

# writing a (trivial) XOR module (cont)

```
./kwickbreakGenerator.py
>>>
Please enter your plugin module name, as written.
xorPlugin
Output filename (and path)
xorBruteForceUtil.v
How many cores would you like on the chip?
50
If you have a pipelined design, how many clock delays for valid data?
0
xorBruteForceUtil.v successfully written!
```

Now just create a new project in Xilinx ISE,
load the files, and synthesize

**Done!**

# Subways using MiFare Classic

- Boston (CharlieCard)
- London (Oyster Card)
- Netherlands (OV-Chipkaart)
- Minneapolis
- South Korea (Upass)
- Hong Kong
- Beijing
- Madrid (Sube-T)
- Rio de Janeiro (RioCard)
- New Delhi
- Bangkok

and more


ATTACK
THE
NETWORK

# network security

- Performed site surveys of T stations and offices (no WiFi found)
- Performed wireless device audit
- Found unguarded network switches

# fiber switches in unlocked room

connect fare vending machines to the internal network

# fiber switches in unlocked room

connect fare vending machines to the internal network


WIRESHARK®

# Social Engineering

Executed the “PHANTOM MEETING” attack

Gained access to internal network drops and computers

---

Nobody suspected a thing as we walked into offices and conference rooms…

So we took it up a notch.

first there was **wardialing**

c.**1983** - 2000 – 2001 – 2002 – 2006 – 2007 - 2008

then there was **wardriving**

c.1983 - **2000** – 2001 – 2002 – 2006 – 2007 - 2008

then there was **warwalking**

c.1983 - 2000 – **2001** – 2002 – 2006 – 2007 - 2008

then there was **warflying**
and **warboating**

c.1983 - 2000 – 2001 – **2002** – 2006 – 2007 - 2008

then there was **war-rocketing**

c.1983 - 2000 – 2001 – 2002 – **2006** – 2007 - 2008

then there was **warballooning**

c.1983 - 2000 – 2001 – 2002 – 2006 – **2007** - 2008

and now… **warcarting**

c.1983 - 2000 – 2001 – 2002 – 2006 – 2007 - **2008**

# WarCart

**Pan/Tilt Mechanism**
attachments include antennas or a smoke *grenade launcher*

**19dBi WiFi Antenna**
directional

**Two Laptops**
for control and data logging

**12dBi WiFi Antenna**
omnidirectional

**Scanner**
to pick up various communications

**25-1300 MHz Antenna**
general coverage, great for picking up the police

**Control Box**
w/ key switch for activation

**CCD Camera**
trip documentation

**Antenna Switch Box**
To toggle between antennas and radios

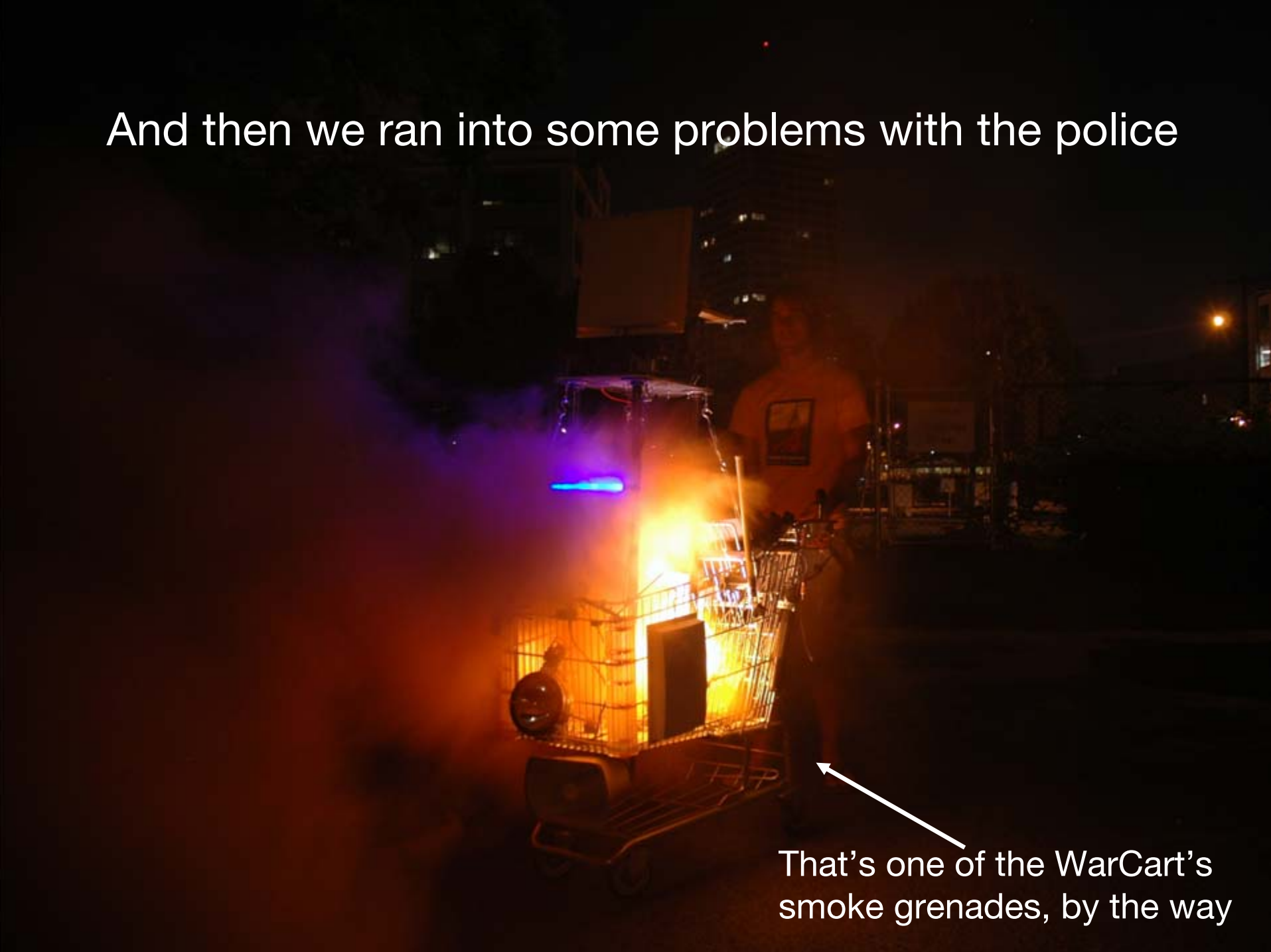
**Lights**
2M candlepower for night operations

**Flash Drive Dropper**
for U3 hacksaws

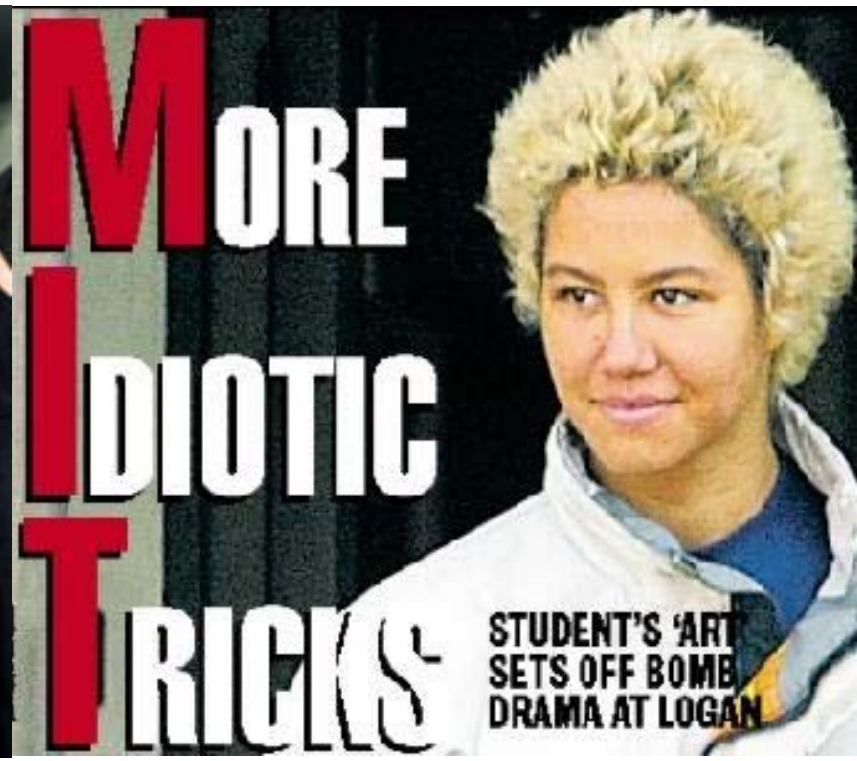
**PA Speaker**
For announcements and intimidating music

**900 MHz Antenna**
directional, great for cordless phones


We decided to take it to the MBTA headquarters
4 family lines
$25 each
MIT
UP TO
50%
OFF


And then we ran into some problems with the police
That’s one of the WarCart’s
smoke grenades, by the way

So to avoid ending up like this




We turned back


contributions

# contributions

1) **Exploited** physical security holes
2) **Reverse engineered** the CharlieTicket
3) Wrote code to analyze & **generate** magcards
4) Wrote a **toolchain** for analyzing 13.56MHz RFID transactions using the USRP+GNUradio
5) **Attacked** problems with the MIFARE Classic cards
6) Wrote **brute forcer**-generator to crack keys on an FPGA
7) Developed software to **reduce MQ to SAT**, allowing key recovery
8) Wrote code to **read and clone** MIFARE cards (given the key)