Fun with Wireless!

Picking up the (metadata) breadcrumbs people don't know they're dropping.

Who am I?

David E. Switzer

- **Experience**: 20+ Years .. Blah Blah.. Train industry, cable industry, and the ISP industry.   Choo-Choo.

- **Interests**: Whispers in the dark and connecting the dots (802.[11|15.1|15.4, metadata, and other things not related).

- **Cert roll call**: GSE #136, G[CIH|PEN|AWN|IA|SEC] (SANS-whore), CISSP, ITILv3 and CPR.

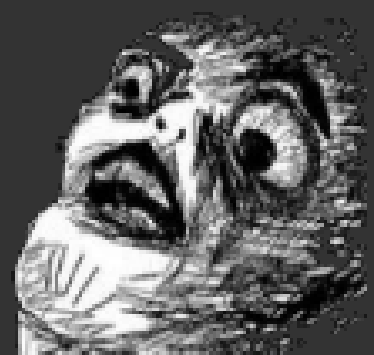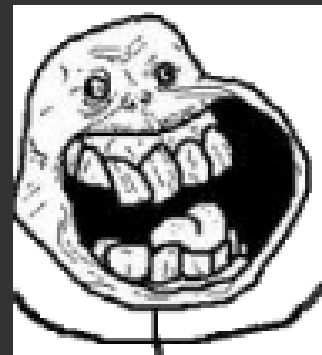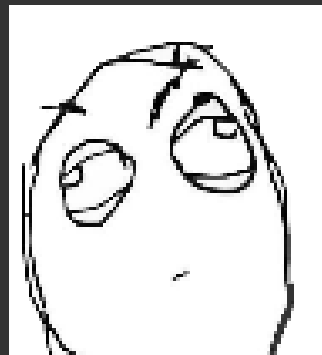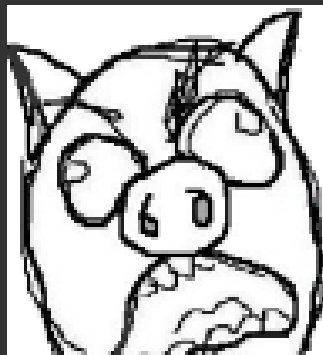- **Job**: Red Team Operator @ ReliaQuest in Tampa, FL

Fun Fact

Glossophobia is the fear of public speaking.

From the Greek γλῶσσα glōssa, meaning tongue, and φόβος phobos, fear or dread.

... no reason ...

Now on to the questionable use of found graphics!

Let me get my meme-to-slide ratio notched up high from the start.
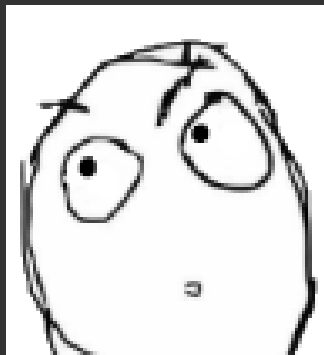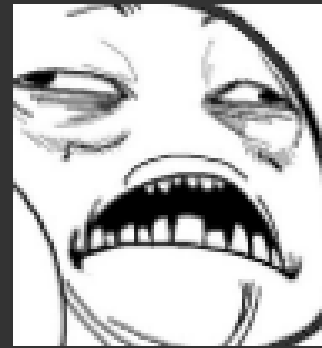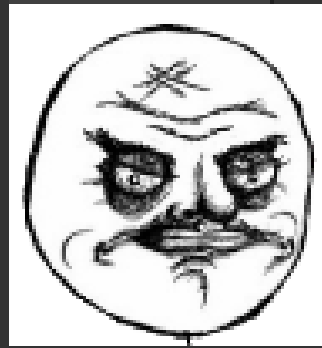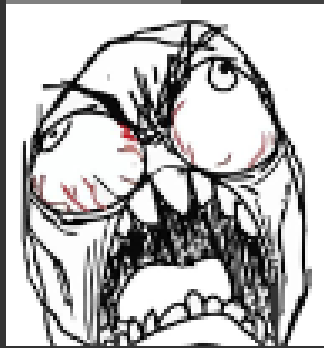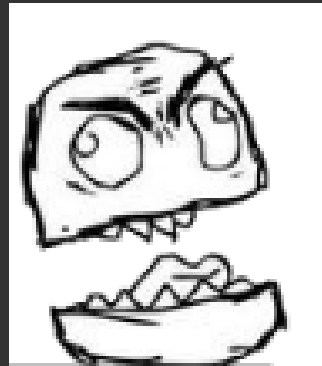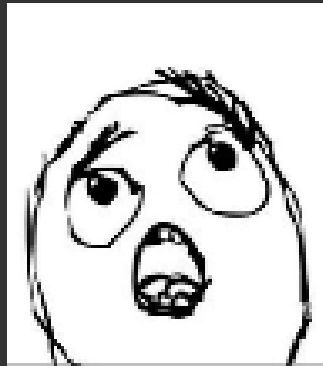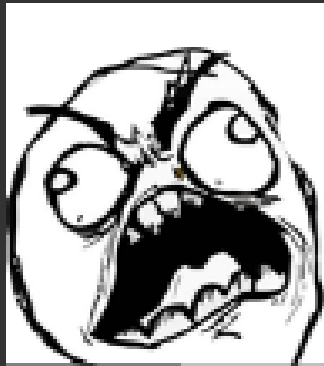
Quick Intro
(I ♥ Auto-Config/Discovery)

- To make device connect to known networks quickly, they tend to send out probes asking if 802.11 networks are available. If an answer comes back, it tries to connect.

- Unfortunately, people can use this design to try to Man-In-The-Middle attack those devices and steal credentials or other valuable information.

We all know that, right?

Everyone else does, too.

Five ways to protect yourself from Wi-Fi honeypots

If you come into range of the WiFi Pineapple Mark IV, every Web page on the Internet may be replaced by the Nyan Cat kitten, or, in the hands of someone malicious, something far worse. Here's how to protect yourself.

WiFi Pineapple Penetration-Testing Tool Sparks Interest at DEF CON

By Sean Michael Kerner | Posted 2013-08-05 🖶 Print

Hacker hunts and pwns WiFi Pineapples with zero-day at Def Con

LOL, @McGrewSecurity
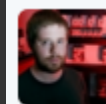
Wesley McGrew @McGrewSecurity · Feb 2
I ❤️ Attack Surface

Businesses watch for these attacks now.

Many vendors have created products, or extended their current security products, to detect WiFi MITM attacks.

FLUKE *networks.*

KASPERSKY

WiFi Pineapple Detected:

Greg Rayburn

2014-10-28

aruba
a Hewlett Packard Enterprise company

tripwire

Well this sure puts a damper on the fun...

CISCO

A honeypot AP may attempt to spoof the BSSID of a valid AP. In that case, it is detected by AP Impersonation Detection.

However, the typical honeypot attack simply duplicates the ESSID without impersonating the BSSID. When Aruba detects an unrecognized AP using a reserved ESSID, it will disable the unrecognized AP and prevent clients from associating to it.

First off – this stuff is simple.

- Nothing here is new.

- Nothing here is complex.

- Nearly all of this could be done with just a web browser and a bit more time.

- I just took the time to connect the dots, and automate some of this.

And now, story time!

Quick Sidebar: A story of potato chips and spies.

- In 2005, Italian courts issued arrest warrants for 26 American nationals for extraordinary rendition of a Muslim cleric from the streets of Milan.

- Kidnapping is much easier to say than "extraordinary rendition".

- CIA Operatives is much easier to say than "American nationals".

- Complete cellphone-related opsec fail is much easier to say than.. Wait, I didn't mention that yet.

Oops.

(Matthew Cole's Blackhat 2013 talk (youtube watch?v=BwGsr3SzCZc) goes over this story in hilarious depth!)

Maybe not so quick:
Potato chips and spies.

- Post kidnapping (er, post extraordinary-renditioning), cell phone metadata was collected in the area and analysed.

- Eventually a pattern of 18 people using about 30 phones was noticed. Their regular interconnects screamed out of the data.

- Now, they had the phone numbers and names associated. Mapping time and locational data, they created a profile of the agents days, including where their day ended.

- The agents, when back at their hotel/home, would "protect" their phones by… tucking them into snack packets, believing they would act as a faraday cage.

But we're talking about Nation State [tm] budgets, we can't do that.

But what can we do?

What can we see?

SSIDs that the device wants to connect to

MAC address/Brand of device

WiFi Modes/Speeds supported

Wireless chipset

Sifting through the data.

While not as easy as cell tower records, or having a spare Stingray or two, we can still look for patterns that may show professional "agents":

- Patterns in device brands.

- MAC addresses that are close, indicating devices possibly purchased at the same time.

- Multiple devices probing to reconnect to the same odd/unknown networks

But instead, let's decide on a specific target and see what we can do.

Fingerprinting a Target

You are near the target.

- Step One: Collect their MAC address and the SSIDs in the WiFi probes they're exposing. (We'll go into some options later.)

Fingerprinting a Target

- Step Two: Figure out which MAC address is your target. Here are a few options:

- Narrow down by SSID

  - **Names** – If you know your target's name, many people still include their name or nickname in their SSID at home.

  - **Office/business name** – if you're collecting away from the office, you may see your target exposing a request for their work place.

  - **Device type** – Do you know what device they use? Narrowing down the device manufacturer by OUI can help.

  - **General OSINT** – If you know the target's name, search their social media for favorite places/"check-ins". If you don't know their name, try to listen in to conversation for clues.

Next: Figuring out where they Exist

- We're down to a MAC, now let's pick a unique SSID.

- **Office SSIDs** – Depending on size/company, could be unique.

- **Restaurants** – SSIDs are often based on restaurant names, or potentially location. (Ex: "Dick's Last Resort", "ApplebeesUCF").

- **Home** – Is it unique in any way?... Weird phrase, nick name, or... ISP based?
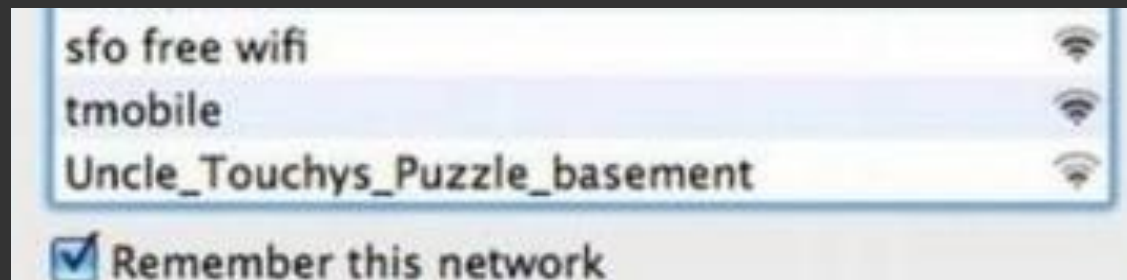
Quick Sidebar: I ♥ ISPs.

- Some ISPs use unique SSIDs on the devices they put into customer homes. Two examples for the Central Florida area:

  - Brighthouse Networks (BHN) (May change w/ pending sale to Charter)
    - BHNTG1462AC812
    - BHNmodel_of_deviceLAST_FOUR_OF_MAC
    - (anyone here currently at BHN?.. Come say HI ;))

  - Verizon FiOS:

    - 7AHL7
    - FiOS-32P9V
    - I_HAVE_NO_CLUE

This unique naming scheme makes it Very easy to find the actual address/location.

So, we know some SSIDs.

How easy is it to turn this information into something useful, like a location?

```
./ssid_to_latlon_to_address_googleapi.sh BHNTG1672GD2F2...
SSID: BHNTG1672GD2F2 , last seen: 1452124364000
Seen at: Latitude: 27.95812945 / Longitude: -82.45951555
Street address: "1428 FL-685, Tampa, FL 33602, USA"
switzerd@fire:/$
```

Turns out, it's pretty easy.

Well, not that easy.. But..

```
switzerd@fire:/$ ./ssid_to_latlon_to_address_googleapi.sh BHNTG1672GD2F2
SSID: BHNTG1672GD2F2 , last seen: 1452124364000
Seen at: Latitude: 27.95812945 / Longitude: -82.45951555
Street address: "1428 FL-685, Tampa, FL 33602, USA"
switzerd@fire:/$
```

The script above just automates a sequence of metadata resolutions:

1st: SSID → Latitude Longitude (DB / remote DB)

2nd: Latitude Longitude → Street Address

People sometimes say this is creepy.

The question shouldn't be "Is this creepy?"

It should be ..

"Can we get creepier?"

What if we are here, but don't know their name?

We've gone from

MAC → SSID → Lat/Lon → Street Address

How do we get to a name?

Property Records in the USA are fairly easy to find, but that only reflects the property owner, leaving out people who live with them, or rents an apartment.

How about Voters Records?

Finding a Voter.

Voting Information is public record in the United States.
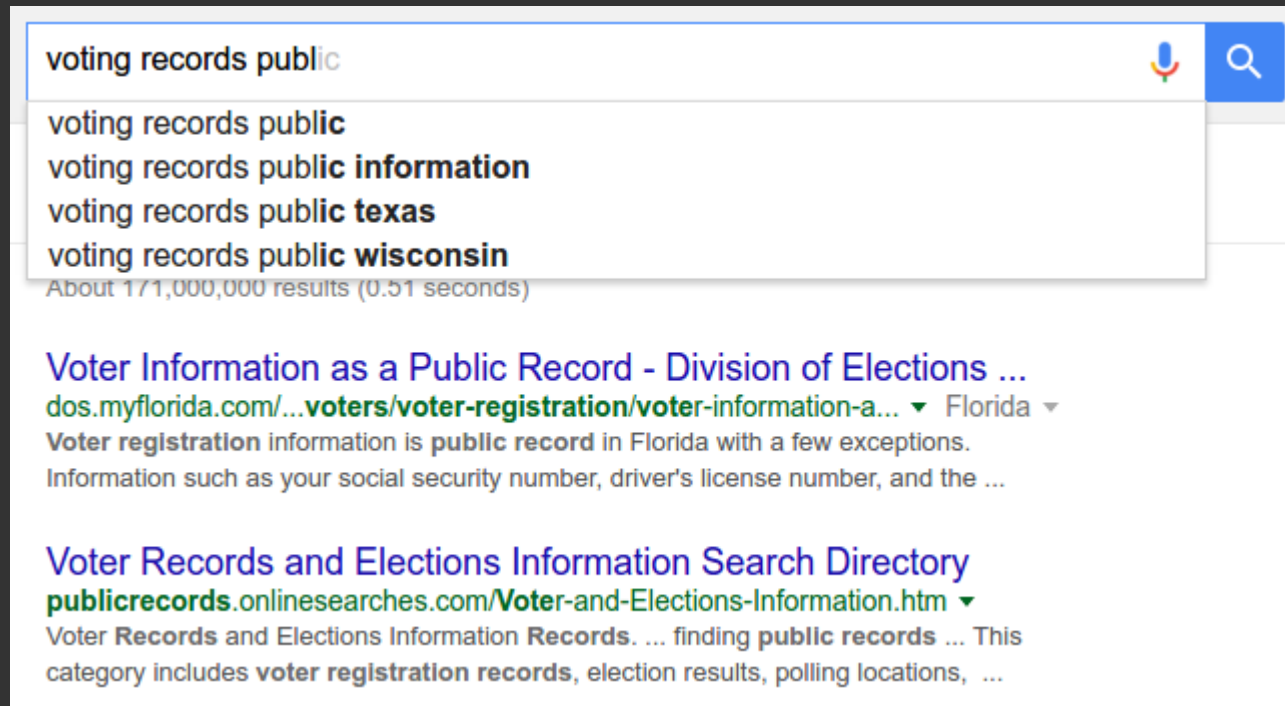
And you can typically download it all in CSVs broken up by county, so you can have it locally.

Finding a Voter.

This is a search for one of the SSIDs at my house.

```
switzerd@fire:~$ ./ssid_to_name.sh ██████████
SSID: ████████ , last seen: 1454989084000
Seen at: Latitude: 28.████████ / Longitude: -82███████████
Street address: 83██████████████ Dr, Tampa, FL 336██ , USA
Your name here: W.██████████████
switzerd@fire:~$
```

It matched the LAT/LON to the house next door to mine,
and the name of the guy renting it.

We know who they are and where they live. Now what?

- If your goal was to MITM the target, their home is the perfect place.

- Their office may have cool WiFi-attack detection toys. Their house probably doesn't.

- People feel safe in their homes – Who's going to mess with their WiFi there, right?

- Also, home usage is mostly going to be outbound to the internet (social media, checking email, etc), not internal network traffic, so they're less likely to notice that you're now in the middle.

…. Creepier ?

So you know where they live or work. You know what their phone is. You know the MAC address.

* See last slide

Bluetooth it is!

- So you know where they live or work. You can easily sniff for interesting Bluetooth devices at their home or business.

- Want to listen in or watch? **There's plenty of Bluetooth devices in the home** - "Smart" Televisions, Gaming headsets, and phone headsets are very common.

- But how can you be sure which device is your target's? Gotta figure out the Bluetooth address on your device!

Bluetooth Rabbit Hole

- Due to cell phone vendors using chipsets where the WiFi and Bluetooth are on the same chip and use sequential addresses, finding the Bluetooth ADDR is usually easy.

- **"Off By One"** – just add and subtract 1 (in hex) from the WiFi address and you probably have the Bluetooth address.

- (Quick script on the Repo to do this and test the results, just so I can even automate this step.. Did I mention I'm lazy?)

- Breaking into an actual connection may not be so easy – but connecting to the devices later and listening/watching should be MUCH easier.

… more on this at a later date. I ran out of time.

The Moving Parts

- And now.. All the stuff that can be used to make this even easier than Googling.

  This info isn't 100%, you have to use sanity check it.

- The locational stuff works better if you've collected a bunch of networks in the area (quicker, more reliable info – you captured it!)

- The scripts I've created try to be Unix-Nice [tm] – simple command line input/output for piping, attempts to do proper exit codes (0/1, etc).

Collection: WUDS

- "Wi-Fi User Detection System"

- Created by: Tim Tomes (@LaNMaSteR53, of recon-ng fame)

- https://bitbucket.org/LaNMaSteR53/wuds/

- Python/Pcapy based system for Unix that records WiFi probes, and optionally alerts to email or "Pushover" (a push app for Android).

Collection: WUDS

- Slight modification: Log the local hostname of the system running WUDS. This allows for "Sensors" feeding a central database, potentially providing location services.

- Additions: Multiple scripts to chop up and use the data further.

  - **SSIDscan.sh** – Either show the unique SSIDs requested, or search for a string.

  - **detect_target.sh / check_all_targets.sh** – check to see if a target has been seen, based any string (SSID, mac address, etc).

  - **Find_Bluetooth_addr.sh** – try to confirm Bluetooth addy by pinging one up and one down on the mac address

  - and others ..

Collection: Hardware

- If you're using WUDS, it's a tiny "heavy" so use this on real-ish systems.

    - - Laptops/servers

    - - Raspberry Pis, Beagle Bones, etc

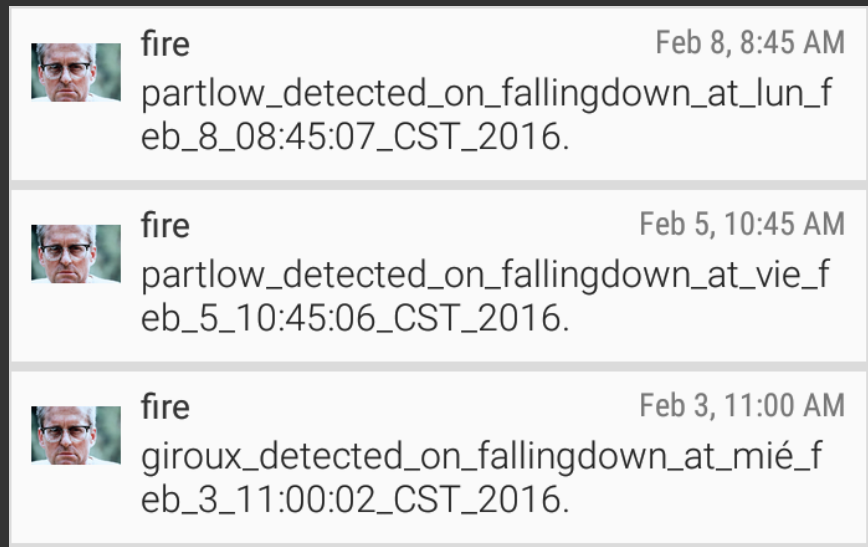- Smaller / Portable?   Use TCP-Dump, later convert the pcaps and import into your DB (scripts on repo)

$10 and needs power

$35 and has a battery

Collection: WUDS

- Another recommended use for WUDS? Keep track of people, and when they come and go.

- The scripts in repo let you set up targets/times and get important information via "PushOver."

| fire | Feb 8, 8:45 AM |
| partlow_detected_on_fallingdown_at_lun_feb_8_08:45:07_CST_2016. | |

| fire | Feb 5, 10:45 AM |
| partlow_detected_on_fallingdown_at_vie_feb_5_10:45:06_CST_2016. | |

| fire | Feb 3, 11:00 AM |
| giroux_detected_on_fallingdown_at_mié_feb_3_11:00:02_CST_2016. | |

Like when your CISO or VP arrive at the office.

Data Sources: WiGLE (Remote/Web)

- WiGLE.net

- "**Crowd Sourced**" war-driving results – SSID locations from around the world.

- Android app for data collection.

- **Pro**: Free and huge amounts of global information.

- **Con**: Slow, no real API for automation, and .. I ran out of time to script that part ;) Check back to the GIT

Data Sources: WiGLE (Local)

- WiGLE.net Android app for data collection.

- App stores data in a simple SQLite3 db.

- **Pro**: Free app, easy data collection, super fast local use.

- **Con**: If you didn't drive past it, the scripts don't know about it.

Data Sources: Google

- Google Map's web based API will provide you a JSON of information based on longitude and latitude.

- This includes breakdowns of metro area, city, neighborhood down to rooftop of the location by lat/lon (by coordinates of NorthEast and SouthWest corners of the area)

- **Pro**: It's Google. So it's free and fast.

- **Con**: It's Google. It's.. not *quite* perfect. Street addresses are often one number off, or will point to a side street not the main address (still accurate, and for tracking in person, close enough).

(Seriously – Google knows/sees all.. You can hate them, but they've got your info anyhow.. May as well use 'em!)

Data Sources: Texas A&M

- Texas A&M University's Geoservices project (geoservices.tamu.edu)

- **Pros**: Accuracy – it's frighteningly dead-on. That's only a few percentage points over Google's data… but still.

- **Cons**: It's slow, it's not completely free (though a free account has enough free queries that most regular people would need), and.. Well I feel better wasting Google's resources to amuse myself than a university's.

Data Sources: Voter Records

- **Public Record in the United States** – just Google for your state.

- Typically available in CSV format broken up by county. Florida is 3.75gigs uncompressed.

- **Pro**: Free, available for all states, small considering the data.

- **Cons**: It's CSV. Either load it into a DB, or at least narrow your searches down by county. 3.75 gigs of CSV can take a tiny bit to search.

Fail Demo

Did I do the goat-sacrifice for the Demo-Gods right?

But a quick example, time permitting.

Not-Really-Live Demo

20 minutes of scanning around 5pm yesterday resulted in these SSIDs being seen in probes:

```
root@fire:/usr/local/src/wuds# ./ssidscan.sh
2WIRE844
AFRC CLASS 5
appletv
belkin54g
Cipher-Backup
Concourse B
deers
default
Disney World
eclubballroom
EClub_Dining
employee
FBI Surveillance Van
FLVS
Gate 10
guest
H0wD33pD03sTh3R4bb1tH0l3G0
Holiday Inn Tampa
```

```
Holiday Inn Tampa
home
linksys
mlb
muffin
<None>
NotYourWiFi 5GHz
orangemuffin
Parkbench
PEPPER
pops
Salt Fork Guest Access
Sheraton LodgeNet
striata_wifi
testdrive
tivo
UIpublicWiFi
UOPX-guest
Westin-Guestrooms
Wolfgang 5GHz
root@fire:/usr/local/src/wuds#
```

Demo

A quick glance shows.. Nothing terribly exciting, but "FLVS" sticks out the most.

```
root@fire:/usr/local/src/wuds# ./ssidscan.sh
2WIRE844
AFRC CLASS 5
appletv
belkin54g
Cipher-Backup
Concourse B
deers
default
Disney World
eclubballroom
EClub_Dining
employee
FBI Surveillance Van
FLVS                  <--
Gate 10
guest
H0wD33pD03sTh3R4bb1tH0l3G0
Holiday Inn Tampa
```

```
Holiday Inn Tampa
home
linksys
mlb
muffin
<None>
NotYourWiFi 5GHz
orangemuffin
Parkbench
PEPPER
pops
Salt Fork Guest Access
Sheraton LodgeNet
striata_wifi
testdrive
tivo
UIpublicWiFi
UOPX-guest
Westin-Guestrooms
Wolfgang 5GHz
root@fire:/usr/local/src/wuds#
```

Demo

So let's see who's asking for that.

```
root@fire:/usr/local/src/wuds# ./ssidscan.sh flvs
2016-03-12 16:22:04.721670|d6:cb:0e:2e:b1:d3|-74|FLVS|Admin OUI|fire
2016-03-12 16:23:35.860018|d6:cb:0e:2e:b1:d3|-82|FLVS|Admin OUI|fire
2016-03-12 16:25:48.351757|4a:1b:0a:b5:64:5c|-80|FLVS|Admin OUI|fire
2016-03-12 16:26:52.735973|4a:1b:0a:b5:64:5c|-85|FLVS|Admin OUI|fire
root@fire:/usr/local/src/wuds#
```

OK, two devices…

Demo

Next – see what SSIDs they have in common:

```
root@fire:/usr/local/src/wuds# ./ssidscan.sh "(d6:cb:0e:2e:b1:d3|4a:1b:0a:b5:64:5c)" | cut -d \| -f 4  | sort | uniq -c | sort -rn
     15 home
     11 <None>
      8 tivo
      8 appletv
      7 pops
      7 belkin54g
      6 testdrive
      6 employee
      5 muffin
      4 orangemuffin
      4 FLVS
      4 deers
      4 2WIRE844
      3 linksys
      3 guest
      1 default
root@fire:/usr/local/src/wuds#
```

"2WIRE844" is likely an AT&T Uverse 2wire Gateway – but the names are commonly reused.

FLVS still is the more interesting SSID.

Demo

Let's see if it's in the local DB:

```
root@fire:~/Desktop/IDEAS/breadcrumbs_chrome/code# ./ssid_to_latlon_to_address_googleapi.sh FLVS
Nothing found in the db..
root@fire:~/Desktop/IDEAS/breadcrumbs_chrome/code#
```

Unfortunately, no.   Off to the website (since I haven't written the web API stuff yet – dangit!)

Demo

The most recent times it was seen are December 11 of 2015.

Notice it was seen on multiple channels – this is probably a multi-AP network, like a business or a public wifi install.

Demo

```
./latlon_to_address_googleapi.sh 25.73454285 -80.43188477
```

```
root@fire:~/Desktop/IDEAS/breadcrumbs_chrome/code# ./latlo
Seen at: Latitude: 25.73454285 / Longitude: -80.43188477
Street address: "3601 SW 147th Ave, Miami, FL 33185, USA"
```

Welp, Google obviously knows about it. Let's see
a map.

Demo
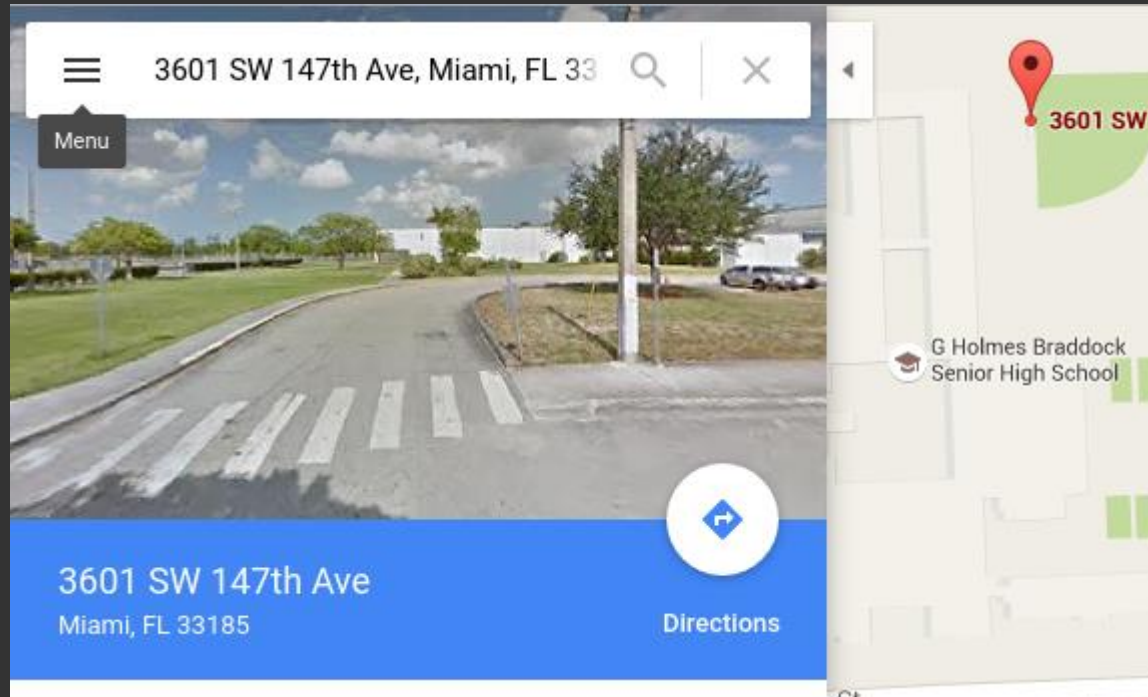
Looks like G. Holmes Braddock Senior High in Miami, Florida.

That would explain the multiple APs.

Demo
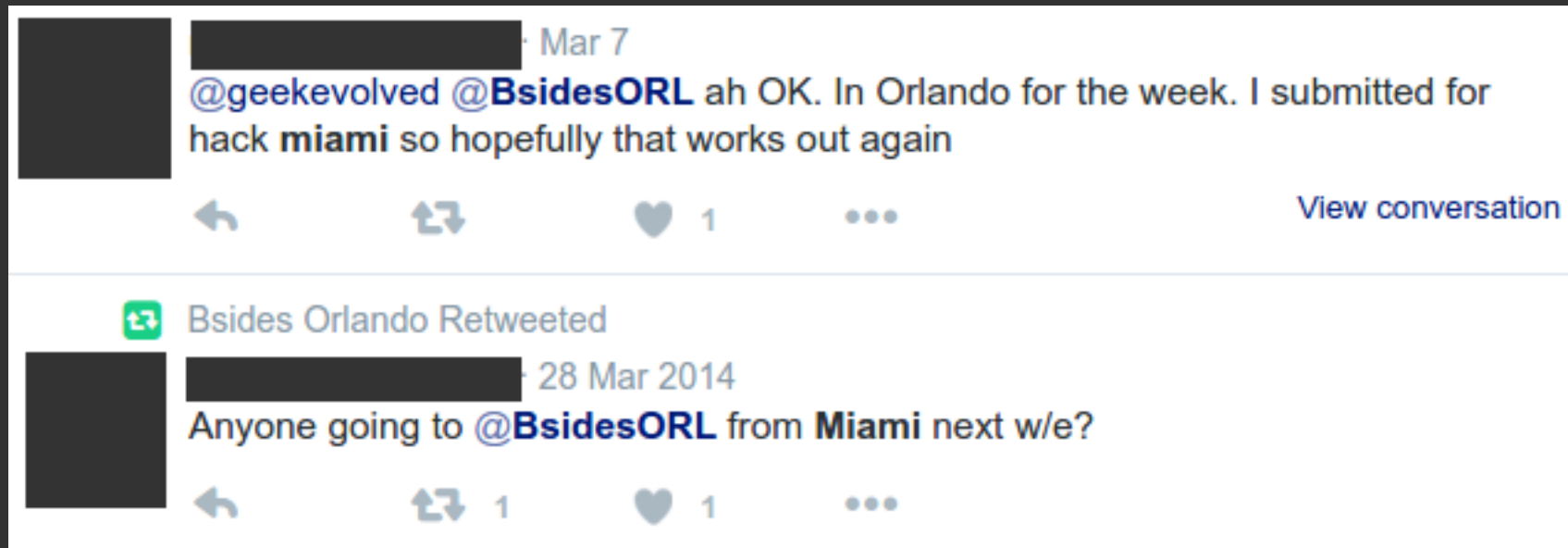
Two "Miami" and "BsidesORL" cross-references – these are two different people in the example.

Fail Demo NOTE

The point of the "fail demo" was to show a quick, non-prepped demo. It ended up being somewhat fail, but it was w/o any target/recon/anything that would help this more.

Also, during the talk someone pointed out that FLVS is "Florida Virtual School", so that AP could have been in any semi-major city in Florida, not necessarily the school in Miami.

So. Let's review what your phone may be telling us.

- Where you work.
- Where you eat.
- Where you hang out.
- … where you live.

… maybe … do you use their WiFi?

OK, maybe not *you*.

- We're all too smart for that, right?

- People are encouraged to *not* connect to public WiFi, but not everyone follows this rule.

- Even those who do still connect at home and at work. That's safe, right?

- Some cell plans are even based on offloading calls/data to WiFi whenever possible. This further encourages connecting to WiFi networks where ever possible – free, fast access to [social media] and lower phone bills! What's not to like?

- And who is *your target* more likely to be? The tech conscious/paranoid sort that's out in this audience, or…

Protecting Yourself:
Regular version

- **Don't let your devices auto-connect anywhere.**

- **Don't advertise your SSIDs at home.**

  - Sure, people can still find your SSID, but this makes it a longer/harder process.

- **Use a boring SSID – like "linksys". (* See last slide)**

  - Don't keep the ISP unique SSID (BHNTG82223, FIOS-AYXZ), don't use your last name, and don't use your 3133t nick name. Be boring.

- **Keep your "known networks" on your devices at a minimum, and clear them out regularly.**

Protecting Yourself:
Tinfoil Hat Edition

- **Regularly change your home SSID.**

    Determined people can still find your SSID even if you hide it– mix it up!

- **Your home is my home – use someone else's SSID from across town.**

    Find a unique SSID from across town, and use that for your home.

- **Visiting a location potentially tracking WiFi devices? Shield yourself, make things exciting!**

    It's easy to make small devices (Android phones, dropboxes) spew out fake WiFi probes/beacons/etc. Let your device get lost in the mix. (Check out my "poaching" repo for code examples)

Protecting Yourself:
EZ Mode

Turn off WiFi on your device when you leave your home / office.

There are applications that attempt to partially automate this – if you're using an Android device, go buy "Smarter WiFi" for $2 (USD). It's by Dragorn of Kismet fame, and you know you probably owe him a couple bucks for his work over the years, right?

And now, it's quiet.

Thanks for Listening

Questions/Suggestions/Additions/fixes?:

github.com/violentlydave // @violentlydave

Email in the code on Github.. Barely obfuscated.

- Thanks to people whom I've met, and some I haven't, who share info and tools: **@LaNMaSteR53, @digininja, Wireshark Developers, Dragorn/Kismet Developers, .. Too many more to list.**

- **Thanks to Jonathan and Kevin for help and hanging out.**

- Thanks for just dealing w/ me through the tests, interviews, papers.. and now talk prep: **Jaci**

* = Notes / Response:

1> Yes, Rainbow tables exist for "Linksys". But I mention that the suggestion is somewhat a joke in the talk, and I would hope if you're at a security conference, your WifI network isn't "password123".

2> That isn't a "kiddie" – that's Danny Glick. Go Google "salems lot danny glick" before you think this is creepy *in that way*. Young whipper snappers ;)