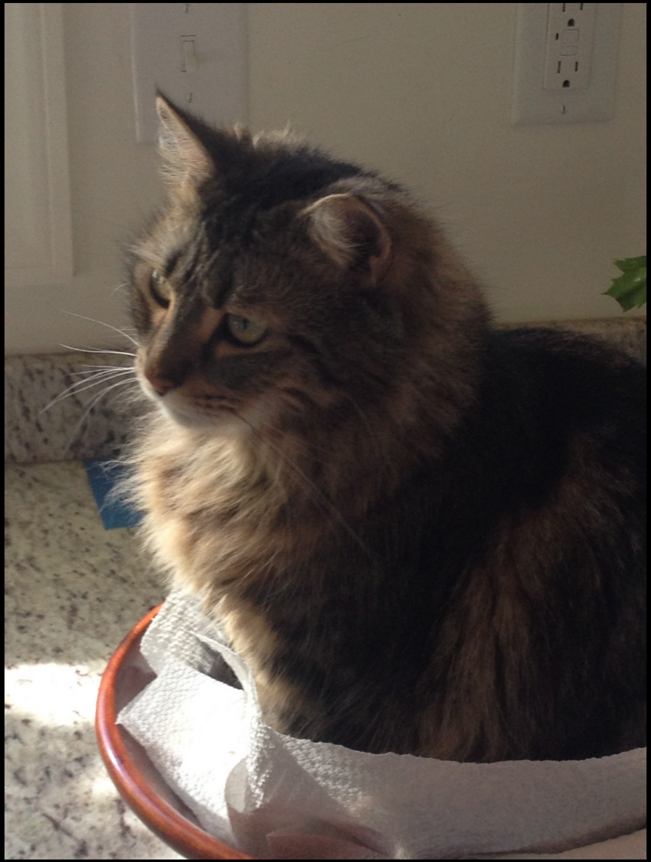
Bluetooth Sniffing and Spoofing

with ‘Punzel our Cat (PoC)



Steve.pote@protonmail.com

Bluetooth Sniffing and Spoofing

with ‘Punzel our Cat (PoC)

- Intended Audience – anyone with interest in Bluetooth, Wifi, R Pi, Cats.
- Many examples are Linux based. ~Most apply equally well to Windows.
- A familiarity with Wireshark and Metasploit is a head start but this is a good chance to try them out for the first time as well (both can be used for ~hands on~ if installed)
- No bluetooth device needed, but if you have an ~ubertooth...bring it.
- No cats were harmed in the production of this presentation, though many were annoyed.

Steve.pote@protonmail.com

└──╼ $cat punzel.txt

- Rapunzel had a sad story with a happy ending.
- She was adopted off the street when tiny.
- She is a very chill kitty, the kind that sits in Doll cloths for a tea party. More on this in a minute.
- PoC is the acronym for Proof of Concept. ‘Punzel makes a fun literal PoC, but an even better metaphor.
- She will walk us through a couple ideas later...

└──╼ $whoami
Steve Pote

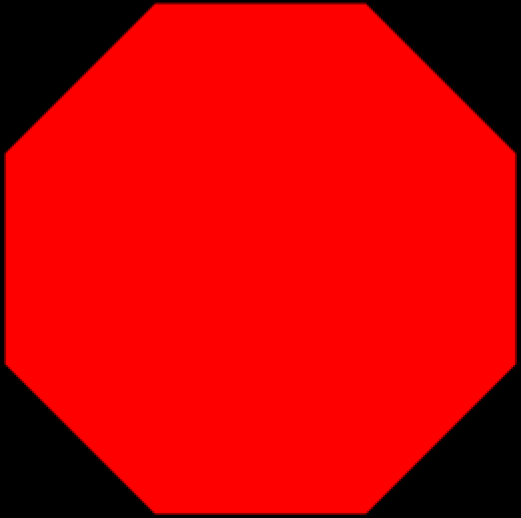
- Chaos Muppet
- Bluetooth & Wifi
- Programming and Development (Especially Naughty)

- MS in Information

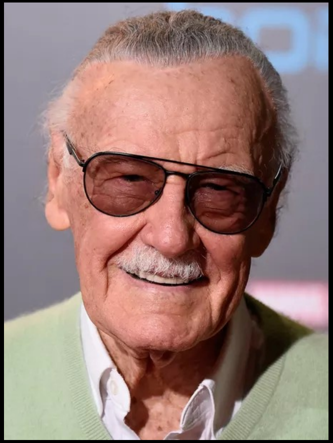
-
- Bartender & Chef...

Steve.pote@protonmail.com

A couple things first...

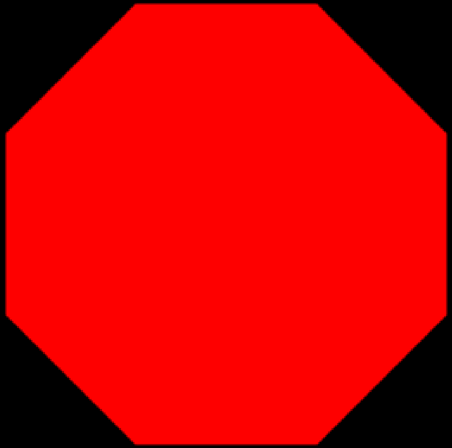


- Not without Permission
- Assume you will break it
- mens rea
- Stan



Steve.pote@protonmail.com

A couple more things ...

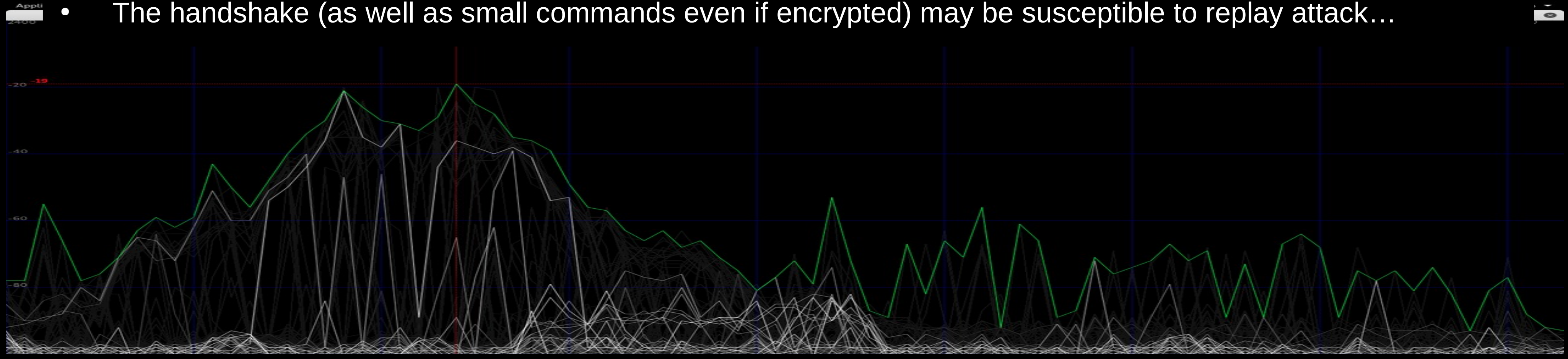


- The R Pi is a great platform to explore with. It can even server some non critical uses and the price is right… ...but, if the need is critical talk to (or become) an embedded device specialist
- Windows?
- Mac??
- Hardware (Capture and Playback, 2 radios…)
- The Wifi Pineapple (same frequency, OpenWRT, Radios!, HackRF)

- Great software (Kismet) exists to explore more deeply (and better) than silly python samples...

BTLE and the ~Cat Scan

- Tip of the Iceberg. BTLE is just a segment of the Bluetooth tech in use (@LibertyUnix). It happens to be a cooperative segment.
- The Link Layer - *Everything topical today. Data about the quietly mumbling devices around us.
- Devices use BTLE to open locks, switch lights, monitor pet movements and heart rates. Traffic flow sensors used by several Departments of Transportation at the State level detect passing BT signals and match them as they pass the next host to calculate traffic speeds. Stores use BTLE to obtain analytics on customer movement
- The handshake (as well as small commands even if encrypted) may be susceptible to replay attack…

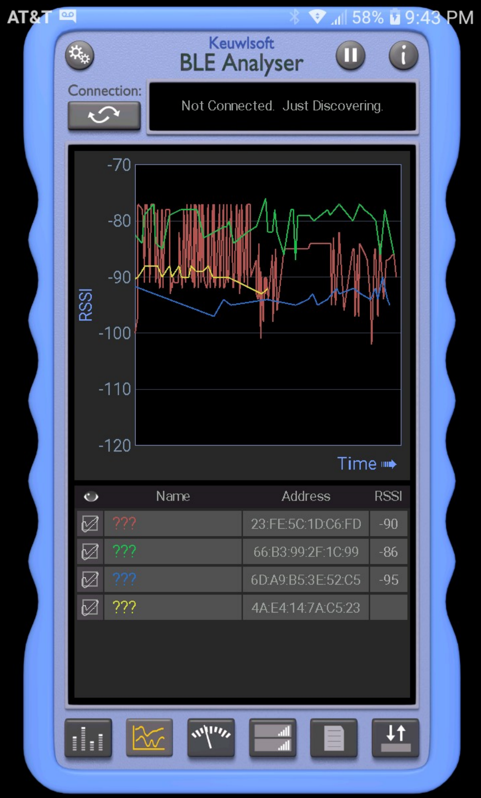


To make absolutely sure you aren’t paying attention...

- WiGLE (stumblers)

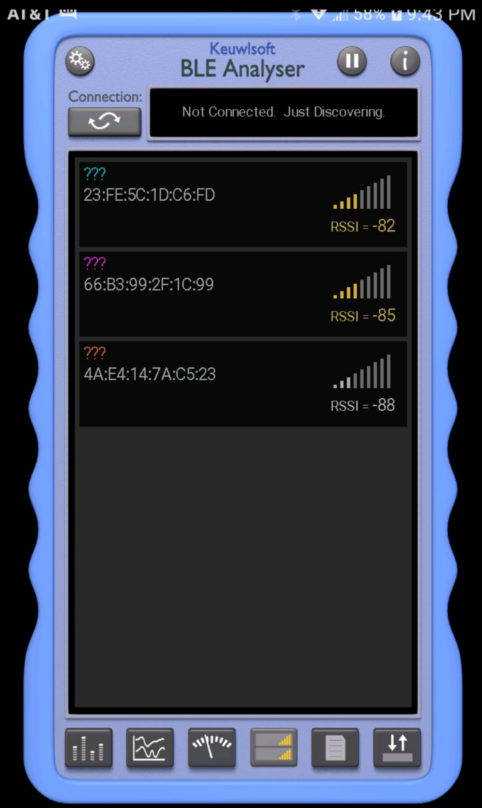
  https://wigle.net/

- ...other bluetooth scanners...



Not Connected. Just Discovering.

Name	Address	RSSI
☑ ???	23:FE:5C:1D:C6:FD	-90
☑ ???	66:B3:99:2F:1C:99	-86
☑ ???	6D:A9:B5:3E:52:C5	-95
☑ ???	4A:E4:14:7A:C5:23	



Not Connected. Just Discovering.

???
23:FE:5C:1D:C6:FD
RSSI = -82

???
66:B3:99:2F:1C:99
RSSI = -85

???
4A:E4:14:7A:C5:23
RSSI = -88

Steve.pote@protonmail.com

PoC

...allow me to demonstrate...we need a few things...

- ‘Punzel our Cat
- Raspberry Pi Zero W (actually a swarm of them)
- Ubertooth1
  (X2 so, ...Uberteeth?)
- Wireshark (tshark, tcpdump, tcpstat)
- Metasploit
- Python (Scapy)

To try this at home...you’ll need to:

- Create a pipe for the Ubertooth to communicate with wireshark ...

  ```
  mkfifo /tmp/pipe
  ```

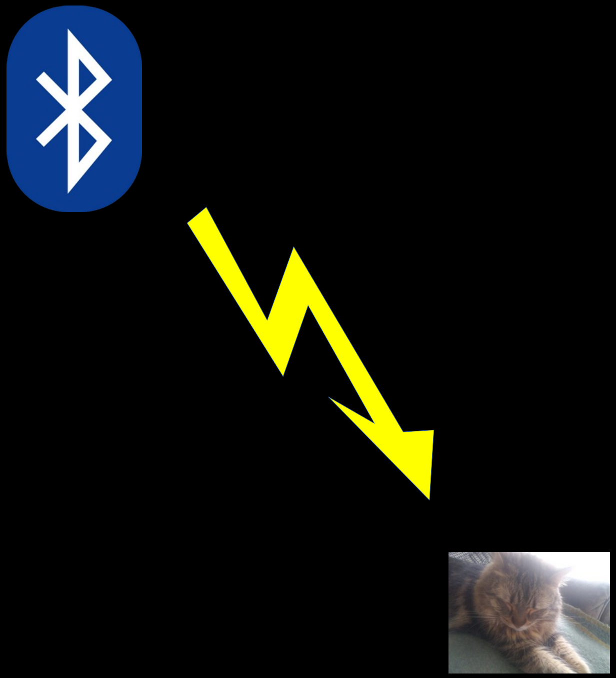
  ```
  ubertooth-btle -f -c /tmp/pipe
  ```

  ...add the pipe to the capture interfaces in Wireshark (and limit capture to that pipe)
- Capture either the complete handshake and command representing a BT transaction or transitory BTLE (LAP)
- Metasploit

  ```
  msf5> use auxiliary/spoof/replay/pcap_replay
  ```

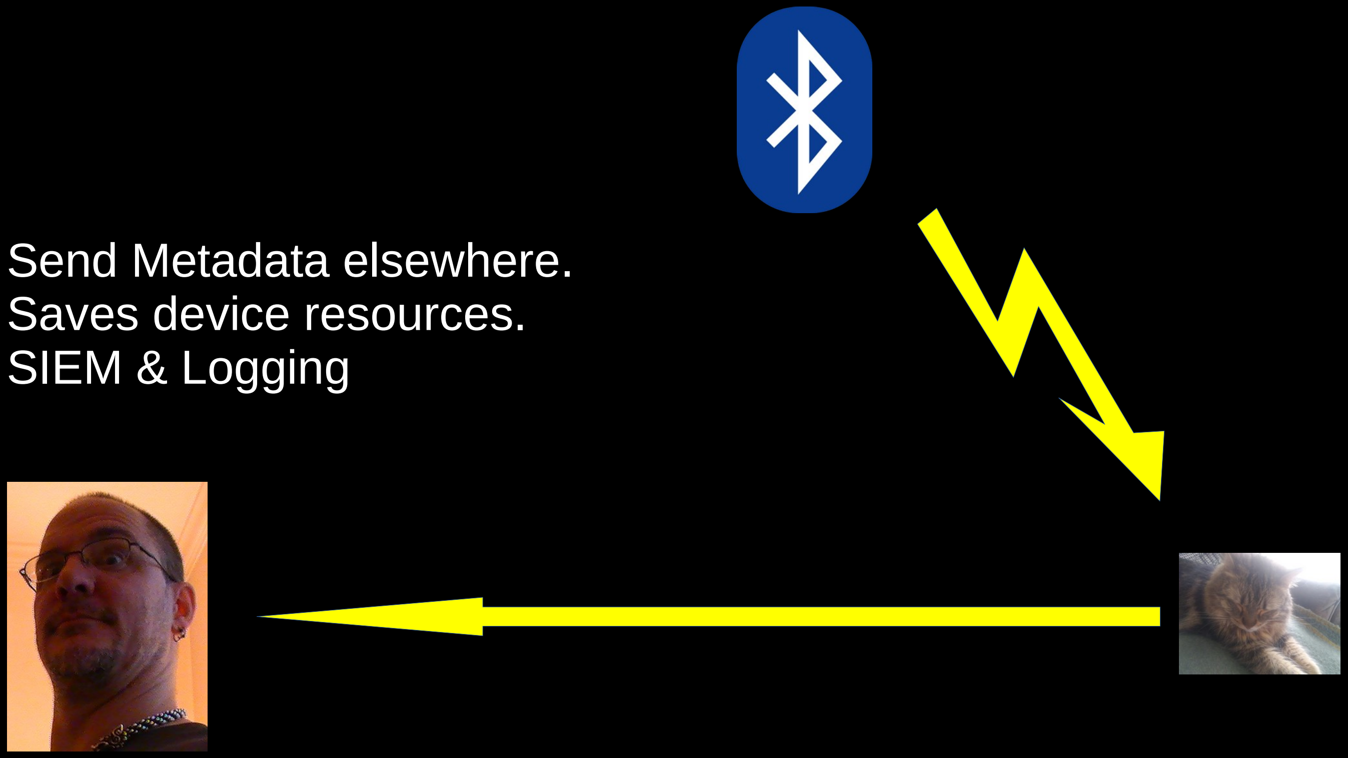
- Other ways to edit and replay (Tcpreplay, Scapy)

An example…Basic Sniffing.



LAP cat.
PoC captures BTLE Requests and responses.
Specifically the Lower Address Part (LAP) of a particular Bluetooth Device Address (BD_ADDR...the MAC address)

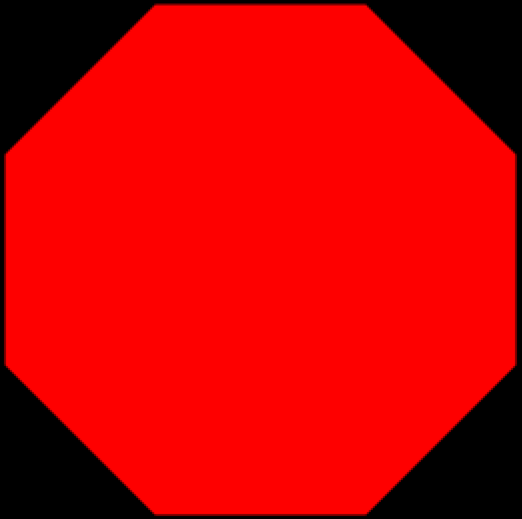
Command and Control



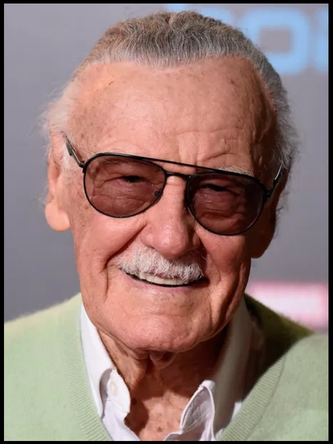
Send Metadata elsewhere.
Saves device resources.
SIEM & Logging

Steve.pote@protonmail.com

Remember this?

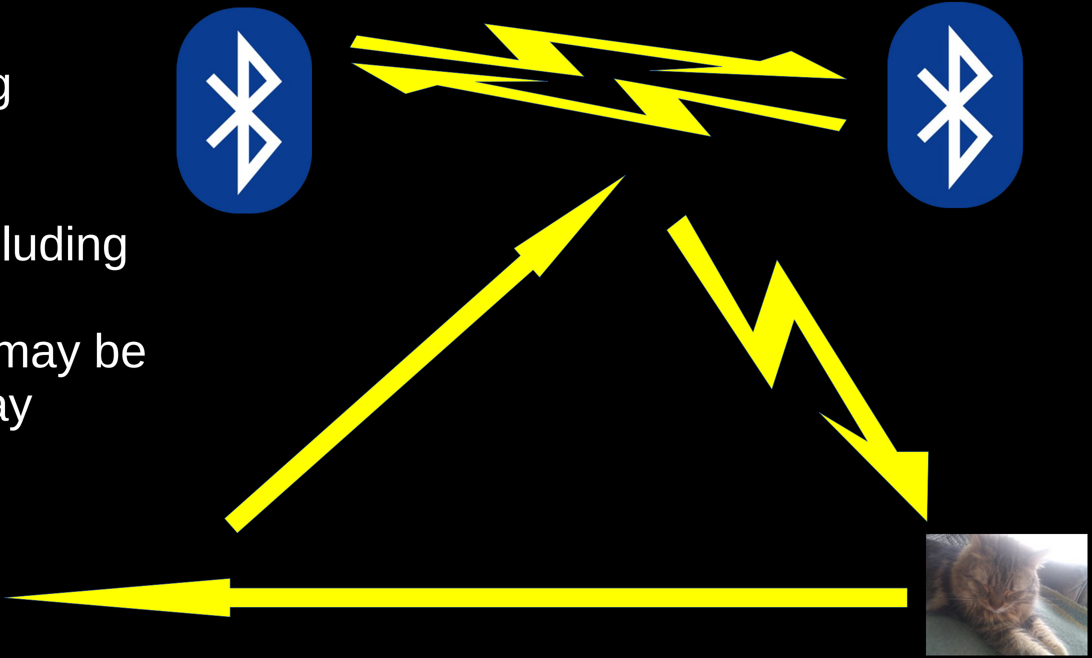


- Not without Permission
- Assume you will break it
- mens rea
- Stan



Playback 1

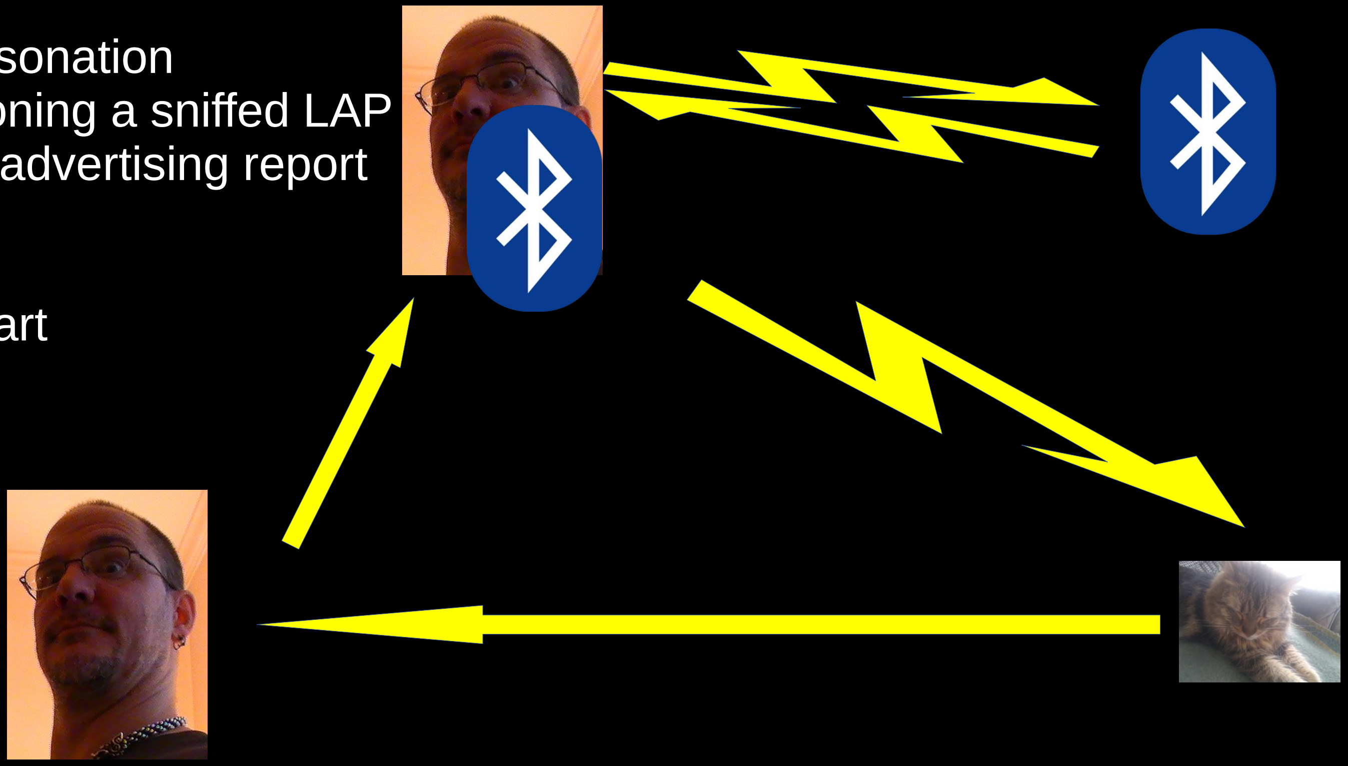
Basic spoof
Record Everything (follow)
Playback whole conversations, including handshake
Lights and Locks may be exploitable this way



Playback 2

Impersonation
Beaconing a sniffed LAP
or LE advertising report

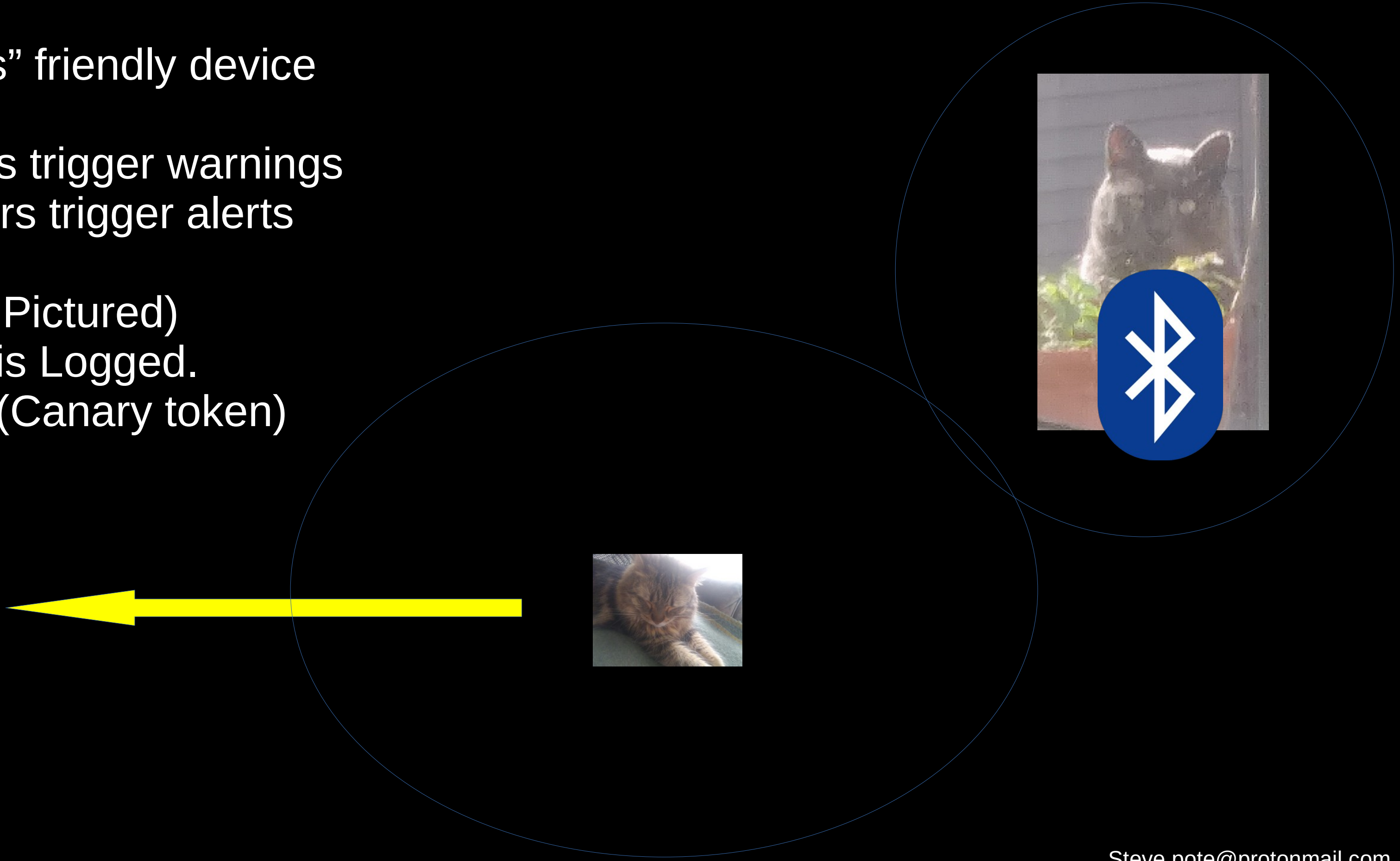
Tile
Walmart
DOT?



Steve.pote@protonmail.com

Win one for the Blue Team

PoC “learns” friendly device LAPs.
Non-Friends trigger warnings
Threat Actors trigger alerts and alarms
(Bluebeard Pictured)
Everything is Logged.
Beaconing (Canary token)



scp

- @scp15487477
- steve[dot]pote[at]protonmail[dot]com
- https://github.com/scp-localhost/DETS

I would love to hear from you but remember I am ~professionally paranoid.

Additional “Props” and Links

- LibertyUnix
- PukingMonky
- DC20
  https://www.defcon.org/images/defcon-20/dc-20-presentations/Holeman/DEFCON-20-Holeman-Scapy.pdf
- Scapy

  https://scapy.readthedocs.io/en/latest/layers/bluetooth.html
- Project Ubertooth

  http://ubertooth.sourceforge.net/usage/start/
-

Steve.pote@protonmail.com

Question…

A rhetorical statement used to test knowledge...but that’s not important right now.

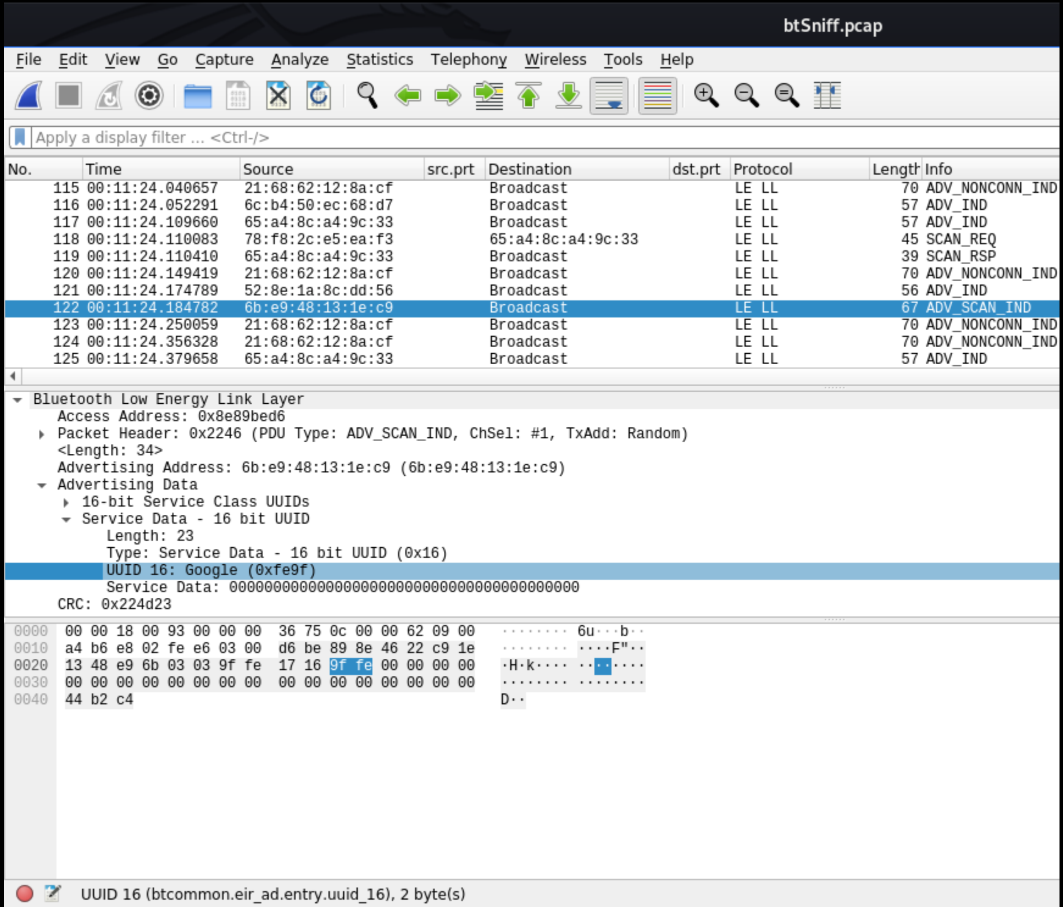
Bonus Slide...

- Wireshark
- Berkley Packet Filter
- Link Layer traffic
- Samples:

  btFragment.pcap (incomplete, fragments)

  btConvo.pcap (dialog between devices)

  btSniff.pcap (What can you find?)


btSniff.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time              Source             src.prt Destination        dst.prt Protocol  Length Info
 115 00:11:24.040657   21:68:62:12:8a:cf          Broadcast                  LE LL     70 ADV_NONCONN_IND
 116 00:11:24.052291   6c:b4:50:ec:68:d7          Broadcast                  LE LL     57 ADV_IND
 117 00:11:24.109660   65:a4:8c:a4:9c:33          Broadcast                  LE LL     57 ADV_IND
 118 00:11:24.110083   78:f8:2c:e5:ea:f3          65:a4:8c:a4:9c:33          LE LL     45 SCAN_REQ
 119 00:11:24.110410   65:a4:8c:a4:9c:33          Broadcast                  LE LL     39 SCAN_RSP
 120 00:11:24.149419   21:68:62:12:8a:cf          Broadcast                  LE LL     70 ADV_NONCONN_IND
 121 00:11:24.174789   52:8e:1a:8c:dd:56          Broadcast                  LE LL     56 ADV_IND
 122 00:11:24.184782   6b:e9:48:13:1e:c9          Broadcast                  LE LL     67 ADV_SCAN_IND
 123 00:11:24.250059   21:68:62:12:8a:cf          Broadcast                  LE LL     70 ADV_NONCONN_IND
 124 00:11:24.356328   21:68:62:12:8a:cf          Broadcast                  LE LL     70 ADV_NONCONN_IND
 125 00:11:24.379658   65:a4:8c:a4:9c:33          Broadcast                  LE LL     57 ADV_IND
```

```
▾ Bluetooth Low Energy Link Layer
    Access Address: 0x8e89bed6
  ▸ Packet Header: 0x2246 (PDU Type: ADV_SCAN_IND, ChSel: #1, TxAdd: Random)
    <Length: 34>
    Advertising Address: 6b:e9:48:13:1e:c9 (6b:e9:48:13:1e:c9)
  ▾ Advertising Data
    ▸ 16-bit Service Class UUIDs
    ▾ Service Data - 16 bit UUID
          Length: 23
          Type: Service Data - 16 bit UUID (0x16)
          UUID 16: Google (0xfe9f)
          Service Data: 0000000000000000000000000000000000000000
    CRC: 0x224d23
```

```
0000  00 00 18 00 93 00 00 00  36 75 0c 00 00 62 09 00   ········ 6u···b··
0010  a4 b6 e8 02 fe e6 03 00  d6 be 89 8e 46 22 c9 1e   ········ ····F"··
0020  13 48 e9 6b 03 03 9f fe  17 16 9f fe 00 00 00 00   ·H·k···· ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  44 b2 c4                                           D··
```

UUID 16 (btcommon.eir_ad.entry.uuid_16), 2 byte(s)

Bonus Slide...

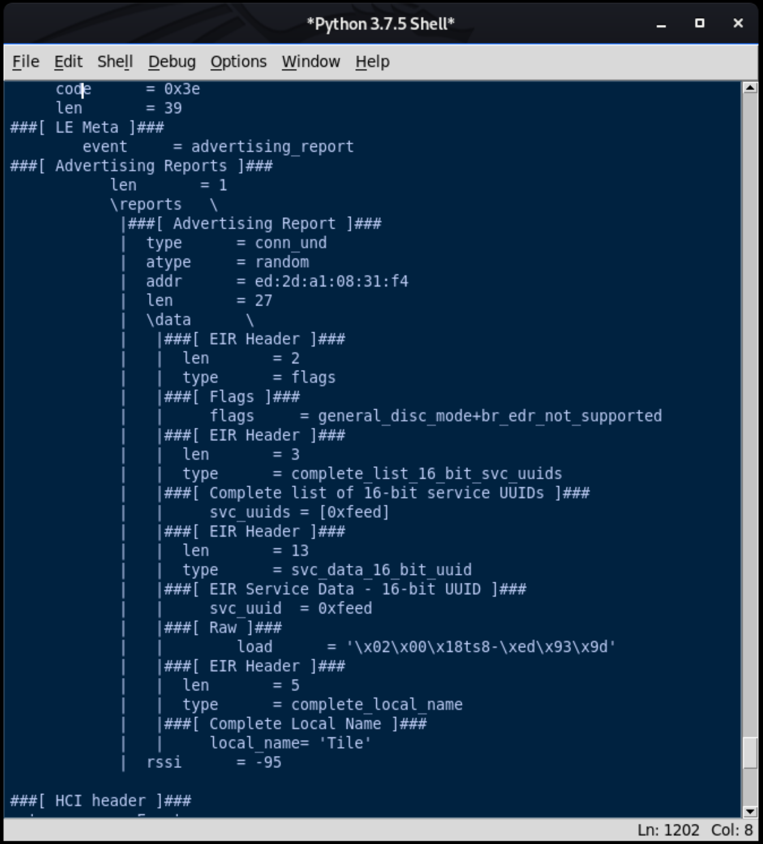
- PerimeterPig.py

  Lots of my stuff has Pig names as homage to Snort

  1) Train Db with friends

  2) Add Threat Actors list

  3) Run in Guard Mode



```
     code      = 0x3e
     len       = 39
###[ LE Meta ]###
        event     = advertising_report
###[ Advertising Reports ]###
           len       = 1
           \reports   \
            |###[ Advertising Report ]###
            |  type      = conn_und
            |  atype     = random
            |  addr      = ed:2d:a1:08:31:f4
            |  len       = 27
            |  \data      \
            |   |###[ EIR Header ]###
            |   |  len       = 2
            |   |  type      = flags
            |   |###[ Flags ]###
            |   |     flags     = general_disc_mode+br_edr_not_supported
            |   |###[ EIR Header ]###
            |   |  len       = 3
            |   |  type      = complete_list_16_bit_svc_uuids
            |   |###[ Complete list of 16-bit service UUIDs ]###
            |   |     svc_uuids = [0xfeed]
            |   |###[ EIR Header ]###
            |   |  len       = 13
            |   |  type      = svc_data_16_bit_uuid
            |   |###[ EIR Service Data - 16-bit UUID ]###
            |   |     svc_uuid  = 0xfeed
            |   |###[ Raw ]###
            |   |        load      = '\x02\x00\x18ts8-\xed\x93\x9d'
            |   |###[ EIR Header ]###
            |   |  len       = 5
            |   |  type      = complete_local_name
            |   |###[ Complete Local Name ]###
            |   |     local_name= 'Tile'
            |  rssi      = -95

###[ HCI header ]###
```

Steve.pote@protonmail.com