Blackhat Vegas, 2011

PPI-Geolocation: The next generation of 802.11 visualization and geo-location
• Jon “Johnny Cache” Ellch
  – Co-Author: Hacking Exposed: Wireless
  – Many 802.11 cracking utilities
  – Aspiring Atari 2600 programmer
  – Wireless Engineer, Harris Corp.
  – Youngest obsolete guy around.
802.1 visualization now:

HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)

SITUATION:
THERE ARE 14 COMPETING STANDARDS.

14?! RIDICULOUS!
WE NEED TO DEVELOP
ONE UNIVERSAL STANDARD
THAT COVERS EVERYONE'S
USE CASES. YEAH!

[Soon:]

SITUATION:
THERE ARE 15 COMPETING STANDARDS.
Motivation

The world has too many file formats for surveys: .ns1, .ns2, .netxml, .gpx, .csv, .wU, Blaarhg...

Let's clean up this mess. Why can't all that data be stored inline with the packet?
Per-Packet Information

• PPI for short.
• Developed by CACE in 2008
• Allows applications to store *Per-Packet Information* in standard pcap file **without** breaking compatibility with tools.
What PPI-GEOLOCATION gets us

- Ability to look at a single pcap file and tell when, where, and what captured a given wireless packet.
- Ability to create/modify this data across a wide variety of tools.
- Most obvious use: Universal visualizer
Per-Packet Information

```
pcap_pkthdr
  cap_len  len
  time-stamp

ppi_packetheader
  version flags pph_len

ppi_fieldheader
  pfh_datalen

ppi_fieldheader
  tag-specific data

pfh_dlt_specific_data
  (802.11, radiotap, etc)
```

```markdown
pcap_pkthdr.cap_len

pcap_pkthdr.len (possibly > cap_len)
```
What do we store?

Well, let's start with the GPS:

\[ G = \text{GPS()} \]

\[ G.\text{Latitude} = 40.787743 \]

\[ G.\text{Longitude} = -73.971210 \]

g = \text{GPS}(\text{str}(G)) \quad \# \text{Cause the packet to be built}

g.show2()
Well that’s a start, but..

• What direction were we travelling?
• Where was the antenna pointed?
• Which way was the car pointing?
• We need a Vector
Vector details

• Vectors specify arbitrary 3-dimensional orientation.
• Vectors can be relative to each other, or relative to earth.
• Vector Characteristic bitmask is used to denote what vector represents. Most likely uses are Direction of Travel, Front of Vehicle, and Antenna.
• \( V = \text{Vector}() \)
• \( V.\text{VectorFlags} = \text{RelativeToEarth} \)
• \( V.\text{VectorChars} = \text{Direction of Travel} \)
• \( V.\text{Pitch} = 10.0 \)
• \( V.\text{Heading} = 22.5 \)
But where is the antenna pointing?
Okay, that was cool, But..

• What good is knowing the orientation if we don’t know what sort of antenna is attached?
Antenna()

- Gain = 16
-Horizbw = 30
- ModelName = "SA24-120-9"

GPS: Lat: 40.787743  Lon: -73.971210
Vector: (Forward) (DOT) (Front_of_veh) Pitch: 10.000000  Heading: 22.500000 (VehicleVec)
Sensor: Velocity 20.000000 Meters/sec
Vector: (Antenna) Heading: 90.000000 (AntennaVec) RelativeTo: RelativeTo: Forward

Antenna: Gain: 9 HorizBw: 120.000000 (SA24-120-9)

Header revision: 2
Header pad: 0
Header length: 49
Present: 0x08000007
Antenna flags: 0x00000002
Gain (dBi): 9
HorizBw: 120
ModelName: SA24-120-9
802.11-Common
IEEE 802.11 Beacon frame, Flags: ........
IEEE 802.11 wireless LAN management frame
Okay that was cool, But..

• What about how fast we were going?
• Did I say velocity? I meant Acceleration.
• And temperature.
• And Humidity (?)
S = Sensor()
S.SensorType = "Velocity"
S.Val_T = 20.0

PPI version 0, 239 bytes
Version: 0
Flags: 0x00
Header length: 239
DLT: 105
GPS: Lat: 40.787743  Lon: -73.971210
Vector: (Forward) (DOT) (Front_of_veh) Pitch: 10.000000  Heading: 22.500000 (VehicleVec)

Sensor: Velocity 20.000000 Meters/sec
Header revision: 2
Header pad: 0
Header length: 14
Present: 0x00000021
SensorType: 1 Velocity
Val_T: 20 Meters/sec
Okay that was cool, But..

- I was told there would be pretty pictures!
Visualizing the data with giskismet
Visualizing the same data with ppi-viz
Okay that was cool, But..

• You’ve been working on this for a year and all you have are some bar graphs in GE?
Say hello to my little friend!
2008 Cobalt SS

• 260 HP Supercharged Ecotec Engine
• Ridiculous wing (+5 HP)
• Sunroof optional
Directed Perceptions Servo

- 360° Pan
- +/- 30° Tilt
- 100% software controlled
WiFi Cannon

- 8 dBi antenna
- 30° horizbw
Servo Bot rollout!
These two packets bounced off this guy's house.
These two packets:

<table>
<thead>
<tr>
<th>No.</th>
<th>Time</th>
<th>eth.src</th>
<th>eth.dst</th>
<th>Protocol</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>6973</td>
<td>13:34:33.976766</td>
<td>00:24:b2:d5:14:f2</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>802.11</td>
<td>Beacon frame, S</td>
</tr>
<tr>
<td>8005</td>
<td>13:34:57.918728</td>
<td>00:24:b2:d5:14:f2</td>
<td>ff:ff:ff:ff:ff:ff</td>
<td>802.11</td>
<td>Beacon frame, S</td>
</tr>
</tbody>
</table>

Frame 6973: 556 bytes on wire (4448 bits), 556 bytes captured (4448 bits)

PPI version 0, 249 bytes

Version: 0

Flags: 0x00

Header length: 249
DLT: 105

GPS: Lat:41.428833 Lon:-87.234888

Vector: (Forward) Pitch:0.000000 Roll:0.000000 Heading:103.530000 RelativeTo: Earth

Vector: (Antenna) Pitch:0.000000 Roll:0.000000 Heading:116.397796 RelativeTo: Forward

Antenna: Gain: 16 HorizBw: 26.000000 (PA-2416)

IEEE 802.11 Beacon frame, Flags: ...........

IEEE 802.11 wireless LAN management frame

Fixed parameters (12 bytes)

Tagged parameters (271 bytes)

Tag: SSID parameter set: offline

.....d.1...o
ffline... ........$
...........US$
...........P.....P..
These two packets
Okay, That *was* cool

• But nothing! It *was* cool.
• But where is all this code?
  – Wireshark: merged.
  – Scapy: merged (/contrib)
  – Kismet: merged
  – [SDK](http://www.govcomm.harris.com/solutions/products/csp-white-papers.asp)
Get The SDK:

• SDK Includes:
  - C++ Library for reading/writing tags (Linux/Windows)
  - Python state machine that illustrates proper interpretation
  - Ppi-viz-dev: Developer visualizer
  - Ppi-viz: Basic signal strength visualizer
Supporting vendors

You’re Welcome.
Thanks to

- Charlie, Jody, Tyler, Megan, Manny, Dragorn, Phillipe, Gerald, R15, Pandy, Craig, shegget, Kiersten, etc.
- Nick *kind-of-a-big-deal* Petrillo + Pusscat
- My (intentionally-vague) friend Josh.
- HD, skape, rjohnson + the rest of uninformed and 219.