RSAConference2019
San Francisco | March 4–8 | Moscone Center

SESSION ID: SBX3-R2

Bluetooth Reverse Engineering: Tools and Techniques

Mike Ryan

Founder
ICE9 Consulting
@mpeg4codec

ICE9 CONSULTING


ellisys
Explorer
Bluetooth® All-in-One Protocol Analyzer
Power
Operating
Trigger
2.4G
5G
BR/EDR
LE
HCI
Primary

$20,000

Ubertooth One

$120

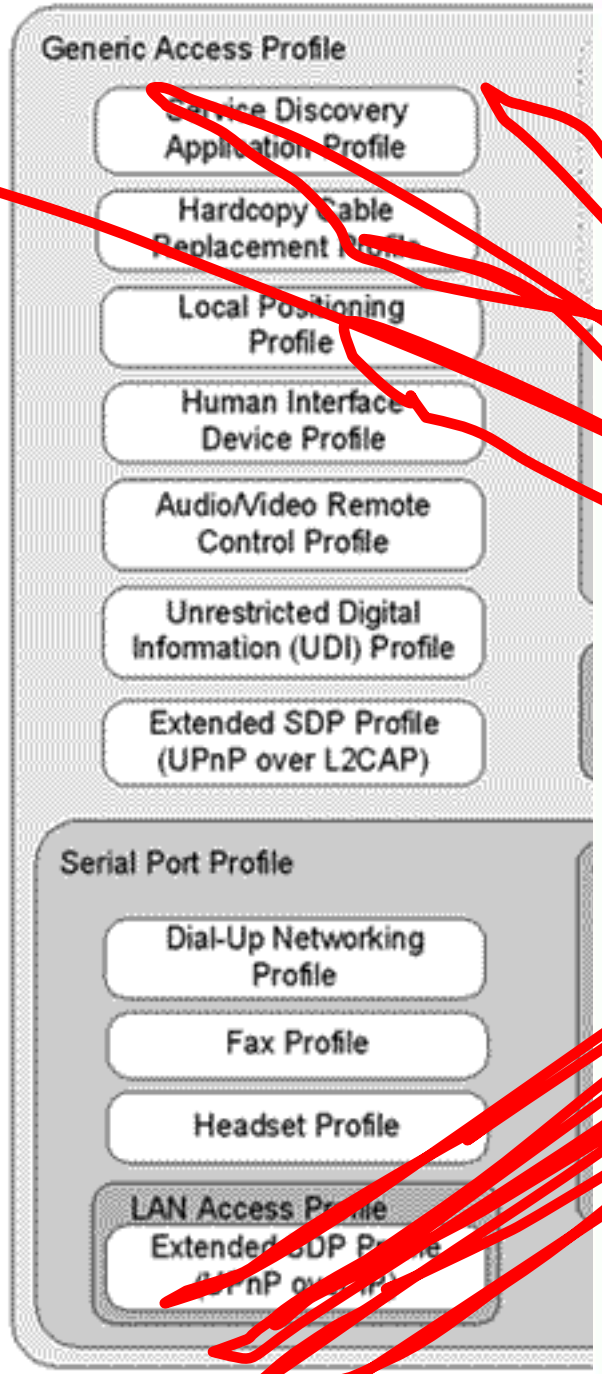
There is a Better Way!

+


WIRESHARK

Outline

1. Overview of Bluetooth
2. Approach 1: Android
   - Case studies 1-3
3. Approach 2: Ubertooth
   - Case study 4
4. Questions



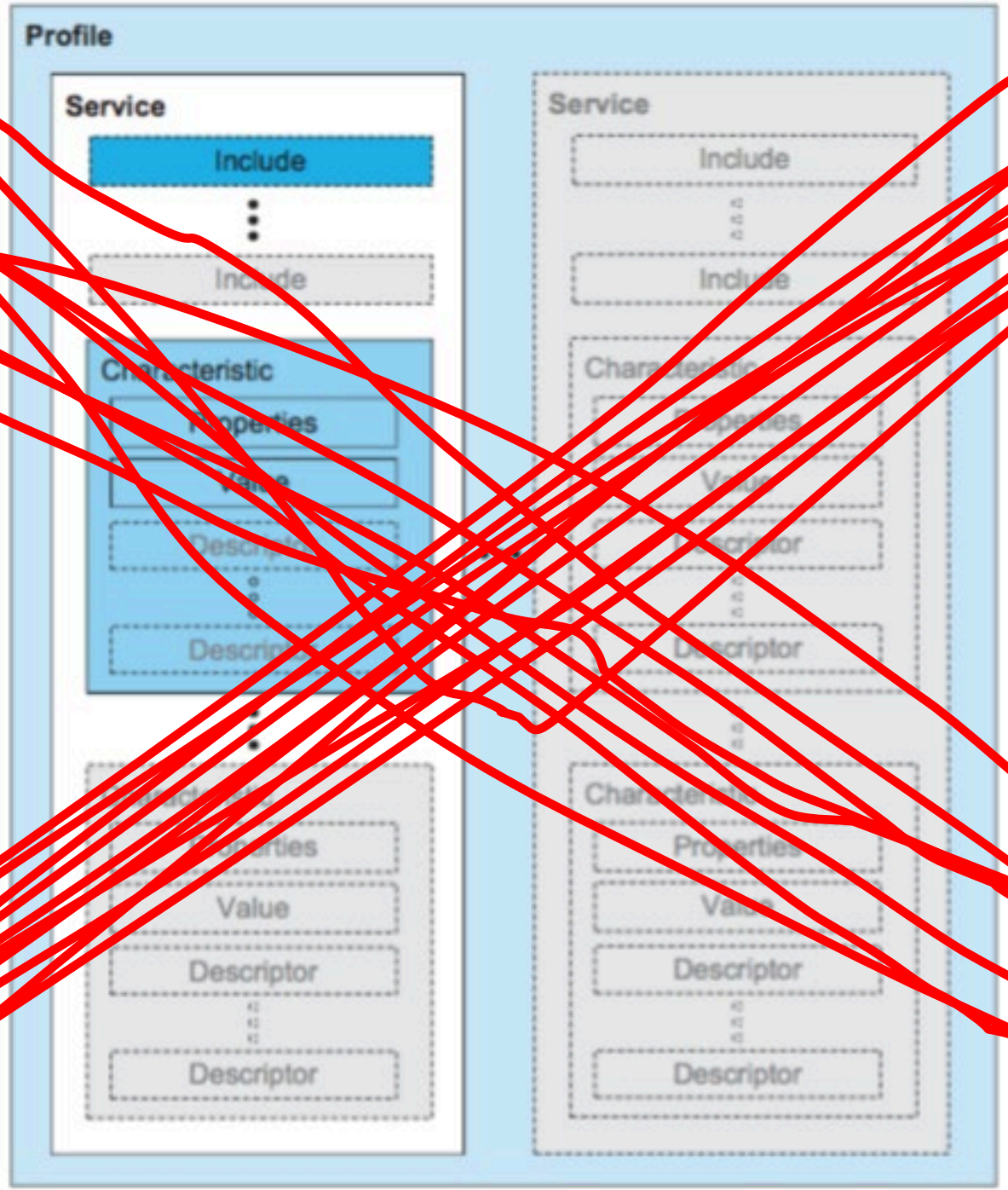
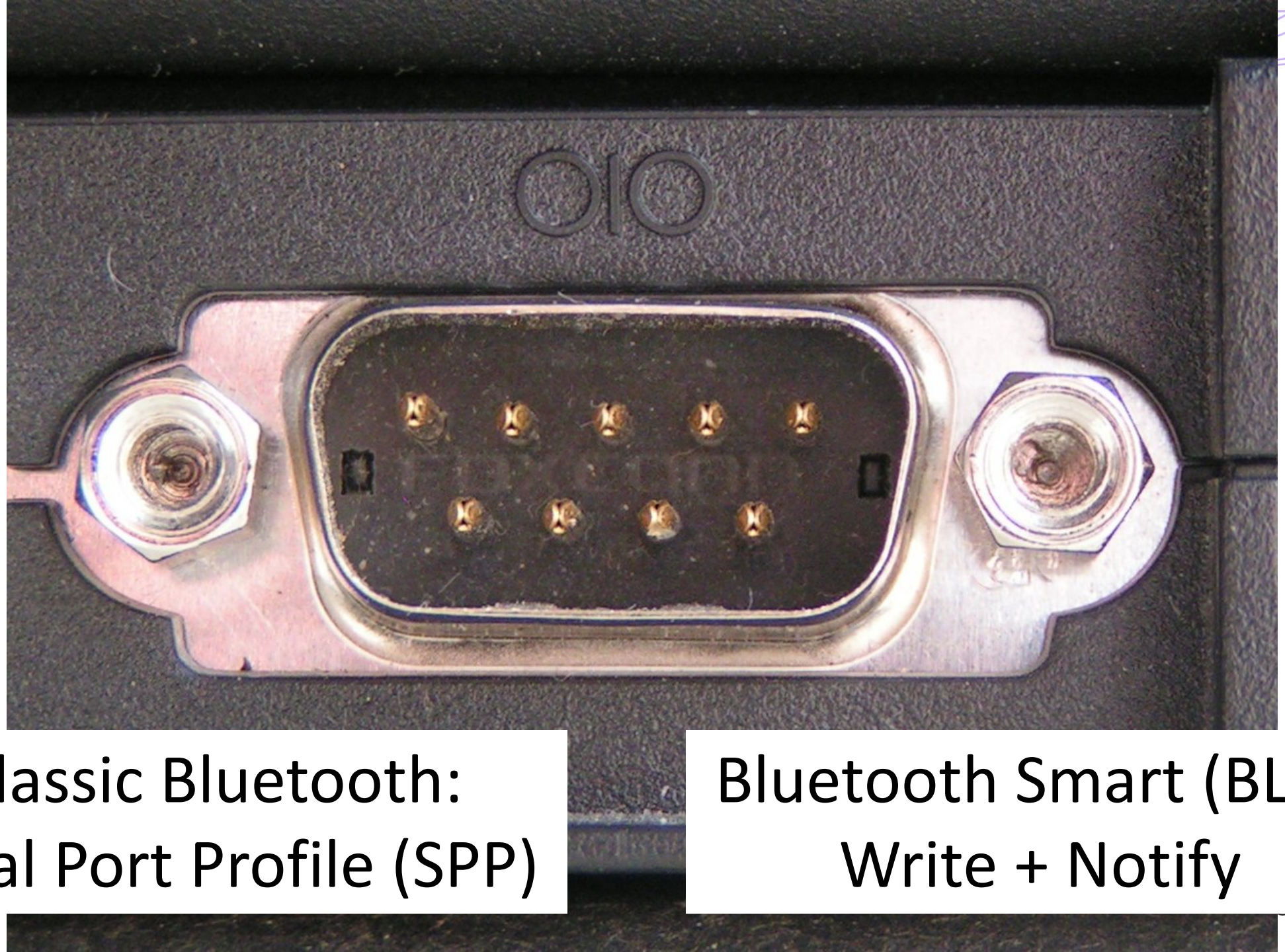


Figure 3: Bluetooth pro

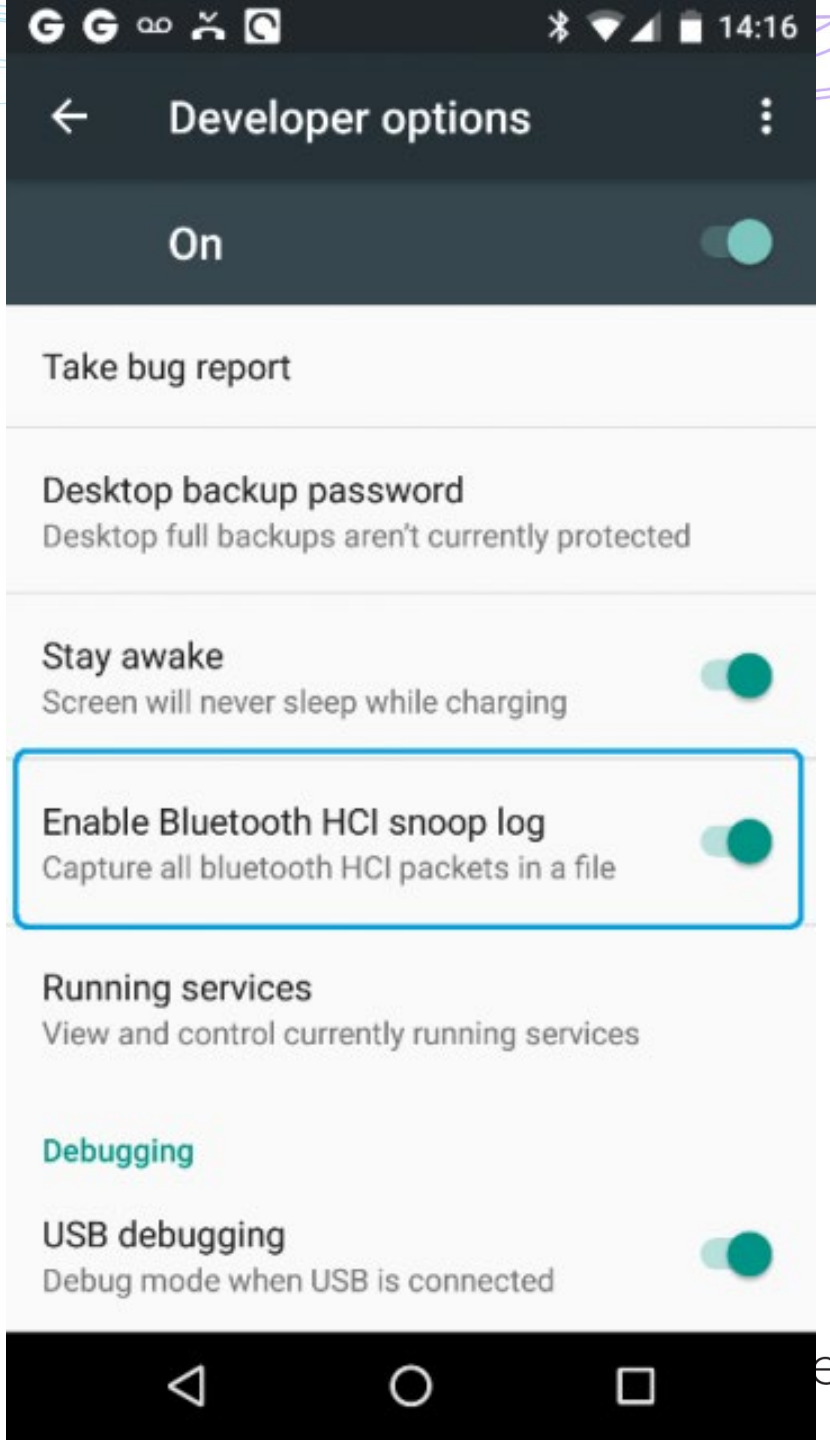
nced Audio Distribution Profile (A2DP)
ute Profile (ATT)
o/Video Remote Control Profile (AVRCP)
Imaging Profile (BIP)
Printing Profile (BPP)
mon ISDN Access Profile (CIP)
Telephony Profile (CTP)
ce ID Profile (DIP)
up Networking Profile (DUN)
Profile (FAX)
Transfer Profile (FTP)
neric Audio/Video Distribution Profile (GAVDP)
neric Access Profile (GAP)
neric Attribute Profile (GATT)
neric Object Exchange Profile (GOEP)
d Copy Cable Replacement Profile (HCRP)
lth Device Profile (HDP)
ds-Free Profile (HFP)
nan Interface Device Profile (HID)
dset Profile (HSP)
rce Profile (ICP)
Access Profile (LAP)
sh Profile (MESH)
ssage Access Profile (MAP)
ect EXchange (OBEX)
ect Push Profile (OPP)
sonal Area Networking Profile (PAN)



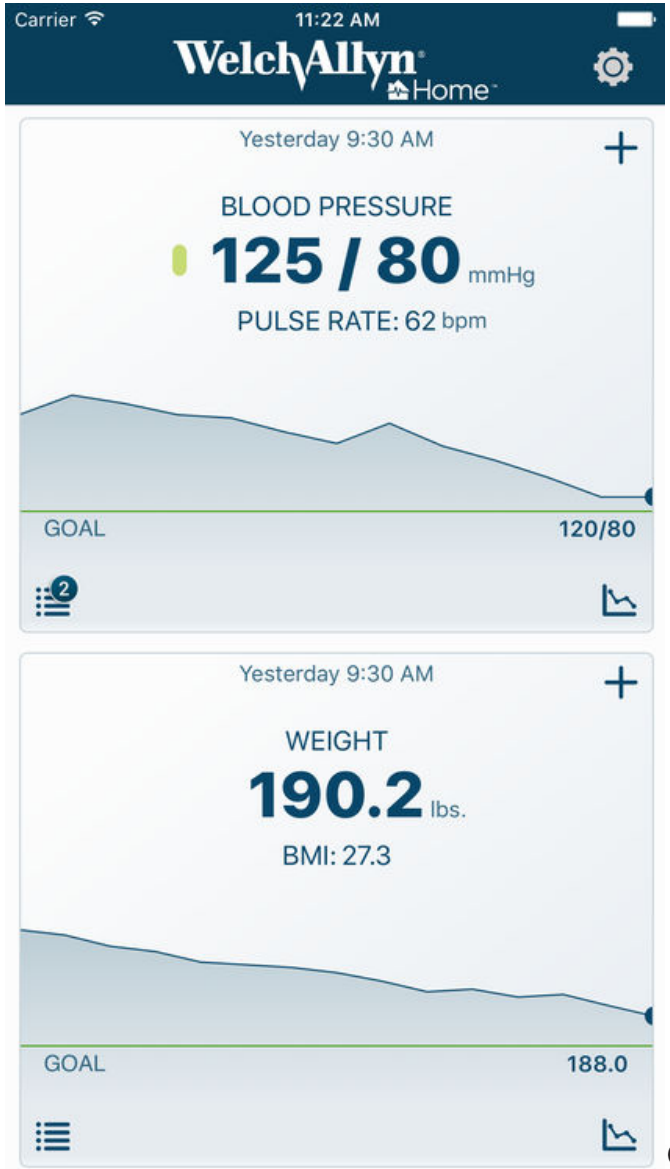
Classic Bluetooth:
Serial Port Profile (SPP)

Bluetooth Smart (BLE):
Write + Notify

Approach 1: Android



Case Study 1: BLE Blood Pressure Monitor

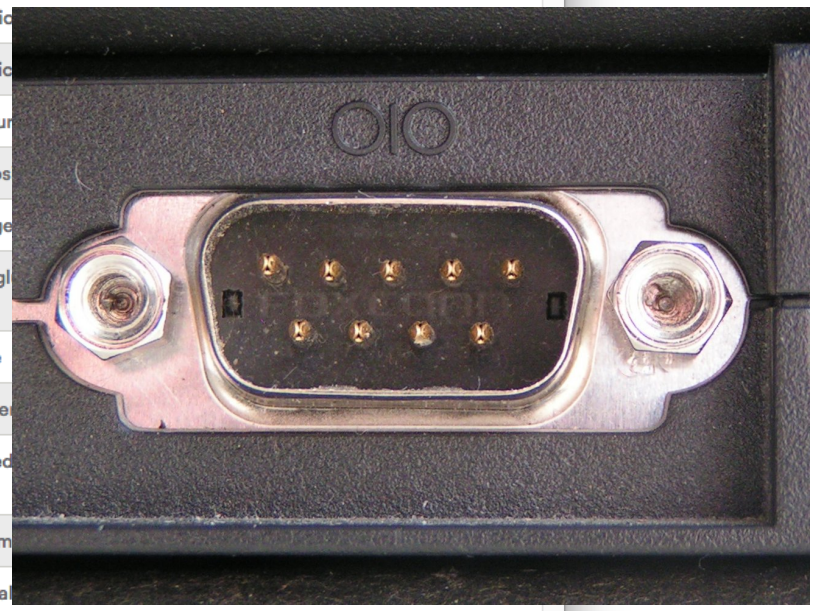


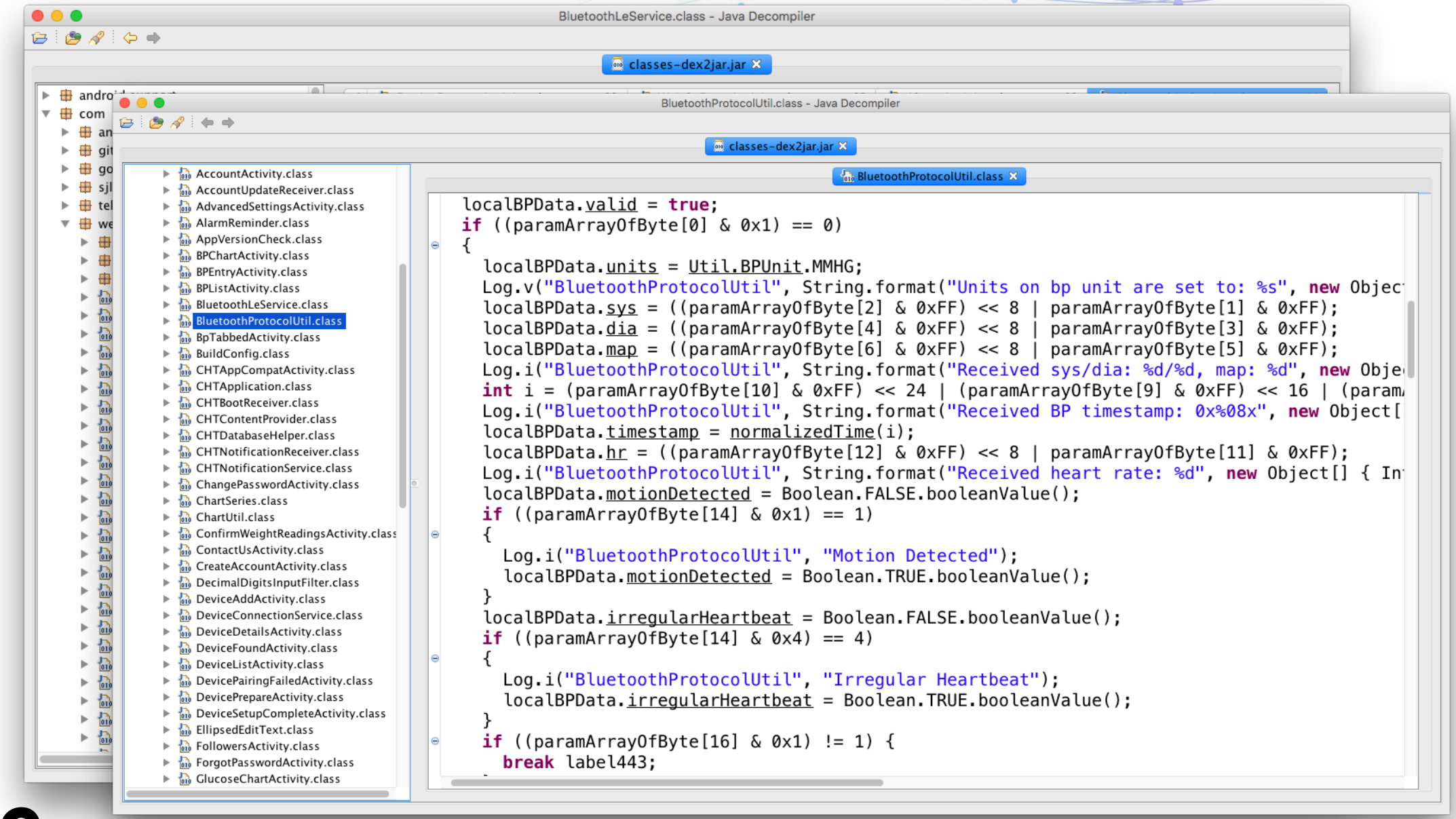
GATT Services | Bluet

https://www.bluetooth.com/specifications/gatt/services

- Protocol Specifications
- **GATT Specifications**
  - GATT Overview
  - GATT Characteristics
  - GATT Declarations
  - GATT Descriptors
  - GATT Services
  - Mesh GATT Services XML
  - Available Schemas
- Errata Service Releases
- Qualification Test Requirements
- Assigned Numbers
- Specifications in Development
- Interoperable Prototype Test Events (IOP)
- Submit an Idea For a Specification
- Archived Specifications

Name	Uniform Type Identifier	Assigned Number	Specification
Generic Access	org.bluetooth.service.generic_access	0x1800	GSS
Alert Notification Service	org.bluetooth.service.alert_notification	0x1811	GSS
Automation IO	org.bluetooth.service.automation_io		
Battery Service	org.bluetooth.service.battery_service		
Blood Pressure	org.bluetooth.service.blood_pressure		
Body Composition	org.bluetooth.service.body_composition		
Bond Management Service	org.bluetooth.service.bond_management		
Continuous Glucose Monitoring	org.bluetooth.service.continuous_glucose_monitoring		
Current Time Service	org.bluetooth.service.current_time		
Cycling Power	org.bluetooth.service.cycling_power		
Cycling Speed and Cadence	org.bluetooth.service.cycling_speed_and_cadence		
Device Information	org.bluetooth.service.device_information		
Environmental Sensing	org.bluetooth.service.environmental_sensing		
Fitness Machine	org.bluetooth.service.fitness_machine	0x1826	GSS
Generic Attribute	org.bluetooth.service.generic_attribute	0x1801	GSS
Glucose	org.bluetooth.service.glucose	0x1808	GSS
Health Thermometer	org.bluetooth.service.health_thermometer	0x1809	GSS
Heart Rate	org.bluetooth.service.heart_rate	0x180D	GSS
HTTP Proxy	org.bluetooth.service.http_proxy	0x1823	GSS




BluetoothLeService.class - Java Decompiler
classes-dex2jar.jar
BluetoothProtocolUtil.class - Java Decompiler
classes-dex2jar.jar
BluetoothProtocolUtil.class
AccountActivity.class
AccountUpdateReceiver.class
AdvancedSettingsActivity.class
AlarmReminder.class
AppVersionCheck.class
BPChartActivity.class
BPEntryActivity.class
BPListActivity.class
BluetoothLeService.class
BluetoothProtocolUtil.class
BpTabbedActivity.class
BuildConfig.class
CHTAppCompatActivity.class
CHTApplication.class
CHTBootReceiver.class
CHTContentProvider.class
CHTDatabaseHelper.class
CHTNotificationReceiver.class
CHTNotificationService.class
ChangePasswordActivity.class
ChartSeries.class
ChartUtil.class
ConfirmWeightReadingsActivity.class
ContactUsActivity.class
CreateAccountActivity.class
DecimalDigitsInputFilter.class
DeviceAddActivity.class
DeviceConnectionService.class
DeviceDetailsActivity.class
DeviceFoundActivity.class
DeviceListActivity.class
DevicePairingFailedActivity.class
DevicePrepareActivity.class
DeviceSetupCompleteActivity.class
EllipsedEditText.class
FollowersActivity.class
ForgotPasswordActivity.class
GlucoseChartActivity.class
localBPData.valid = true;
if ((paramArrayOfByte[0] & 0x1) == 0)
{
localBPData.units = Util.BPUnit.MMHG;
Log.v("BluetoothProtocolUtil", String.format("Units on bp unit are set to: %s", new Objec
localBPData.sys = ((paramArrayOfByte[2] & 0xFF) << 8 | paramArrayOfByte[1] & 0xFF);
localBPData.dia = ((paramArrayOfByte[4] & 0xFF) << 8 | paramArrayOfByte[3] & 0xFF);
localBPData.map = ((paramArrayOfByte[6] & 0xFF) << 8 | paramArrayOfByte[5] & 0xFF);
Log.i("BluetoothProtocolUtil", String.format("Received sys/dia: %d/%d, map: %d", new Obje
int i = (paramArrayOfByte[10] & 0xFF) << 24 | (paramArrayOfByte[9] & 0xFF) << 16 | (param
Log.i("BluetoothProtocolUtil", String.format("Received BP timestamp: 0x%08x", new Object[
localBPData.timestamp = normalizedTime(i);
localBPData.hr = ((paramArrayOfByte[12] & 0xFF) << 8 | paramArrayOfByte[11] & 0xFF);
Log.i("BluetoothProtocolUtil", String.format("Received heart rate: %d", new Object[] { In
localBPData.motionDetected = Boolean.FALSE.booleanValue();
if ((paramArrayOfByte[14] & 0x1) == 1)
{
Log.i("BluetoothProtocolUtil", "Motion Detected");
localBPData.motionDetected = Boolean.TRUE.booleanValue();
}
localBPData.irregularHeartbeat = Boolean.FALSE.booleanValue();
if ((paramArrayOfByte[14] & 0x4) == 4)
{
Log.i("BluetoothProtocolUtil", "Irregular Heartbeat");
localBPData.irregularHeartbeat = Boolean.TRUE.booleanValue();
}
if ((paramArrayOfByte[16] & 0x1) != 1) {
break label443;

Case Study 2: Mystery Classic BT Device

?

1. Transmit & play back audio
2. Misc configuration

~~A2DP? HFP?~~

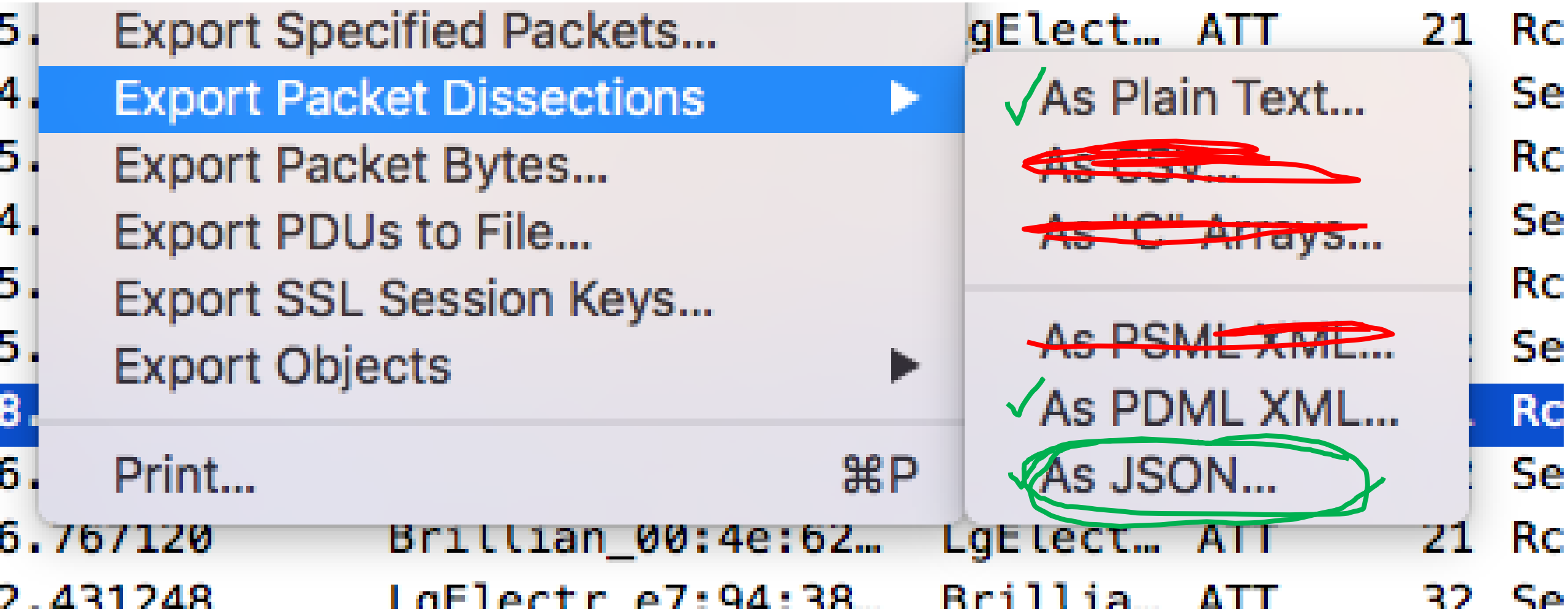
29 Proximity Profile (PXP)

30 Serial Port Profile (SPP)

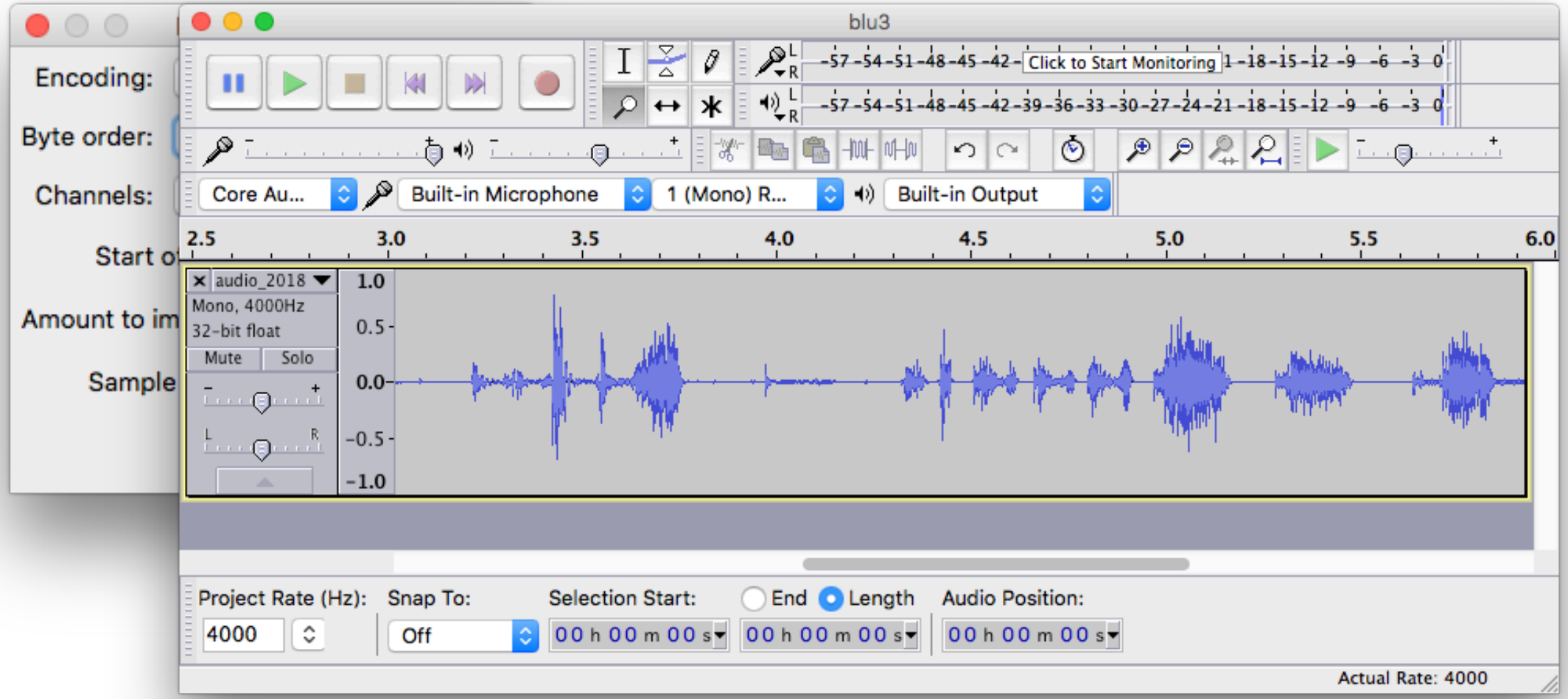
31 Service Discovery Application Pr

32 SIM Access Profile (SAP, SIM, rS

Wireshark ag


Export Specified Packets...
Export Packet Dissections
Export Packet Bytes...
Export PDUs to File...
Export SSL Session Keys...
Export Objects
Print... ⌘P
As Plain Text...
As CSV...
As "C" Arrays...
As PSML XML...
As PDML XML...
As JSON...

Audacity®


blu3
Encoding:
Byte order:
Channels:
Start o
Amount to im
Sample
Core Au...
Built-in Microphone
1 (Mono) R...
Built-in Output
Click to Start Monitoring
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
audio_2018
Mono, 4000Hz
32-bit float
Mute
Solo
1.0
0.5
0.0
-0.5
-1.0
Project Rate (Hz):
4000
Snap To:
Off
Selection Start:
End
Length
Audio Position:
00 h 00 m 00 s
Actual Rate: 4000

Aside: Talking to this device

Step 1: Pair using system dialog

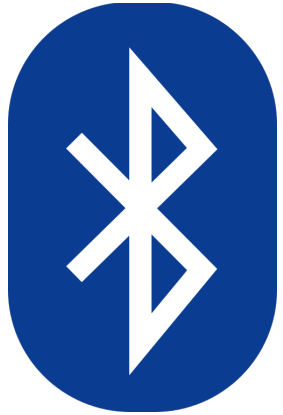
Step 2:

```
from PyBT import BDAddr, BluetoothSocket

bdaddr = BDAddr('11:22:33:44:55:66')
f = BluetoothSocket(socket.AF_BLUETOOTH, socket.SOCK_STREAM,
                    socket.BTPROTO_RFCOMM)
f.connect_rc(bdaddr, 1)

f.send(...)
f.recv(...)
```

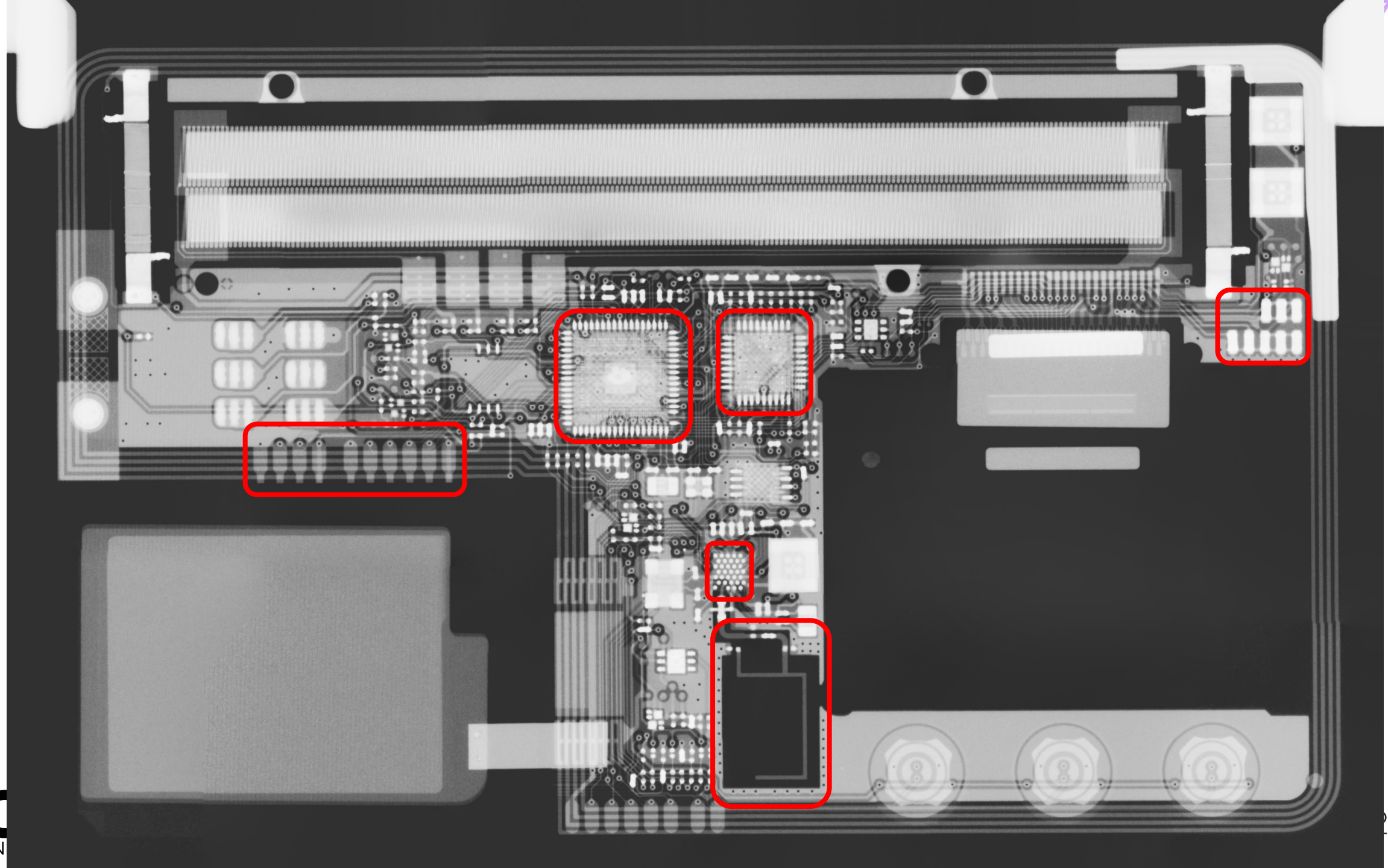
Case Study 3: Bluetooth Credit Card

What is a Bluetooth credit card?

+



==



misc_actions.log

((btatt.opcode.method == 0x12) && (btatt.handle == 0x0018)) || ((btatt.opcode.method == 0x1b) && (btatt.handle == 0x001a))

Expression...

No.	Time	Source	Destination	Protocol	Lengt	Info
3314	316.173622	LgElectr_e7:94:38…	Brillia…	ATT	32	Sent Write Request, Handle: 0x0018 (Unknown: Unknown)
3317	316.259659	Brillian_00:4e:62…	LgElect…	ATT	25	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3318	316.306833	LgElectr_e7:94:38…	Brillia…	ATT	32	Sent Write Request, Handle: 0x0018 (Unknown: Unknown)
3321	316.397758	Brillian_00:4e:62…	LgElect…	ATT	26	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3322	316.409395	LgElectr_e7:94:38…	Brillia…	ATT	32	Sent Write Request, Handle: 0x0018 (Unknown: Unknown)
3325	316.485582	Brillian_00:4e:62…	LgElect…	ATT	32	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3326	316.486765	Brillian_00:4e:62…	LgElect…	ATT	32	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3327	316.489184	Brillian_00:4e:62…	LgElect…	ATT	32	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3328	316.528282	Brillian_00:4e:62…	LgElect…	ATT	29	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3329	316.994640	LgElectr_e7:94:38…	Brillia…	ATT	32	Sent Write Request, Handle: 0x0018 (Unknown: Unknown)
3332	317.068326	Brillian_00:4e:62…	LgElect…	ATT	29	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3371	336.015132	LgElectr_e7:94:38…	Brillia…	ATT	32	Sent Write Request, Handle: 0x0018 (Unknown: Unknown)
3374	337.221906	Brillian_00:4e:62…	LgElect…	ATT	25	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)
3407	345.883355	LgElectr_e7:94:38…	Brillia…	ATT	32	Sent Write Request, Handle: 0x0018 (Unknown: Unknown)
3410	346.843388	Brillian_00:4e:62…	LgElect…	ATT	21	Rcvd Handle Value Notification, Handle: 0x001a (Unknown: Unknown)

```
▶ Frame 3575: 32 bytes on wire (256 bits), 32 bytes captured (256 bits)
▶ Bluetooth
▶ Bluetooth HCI H4
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Write Request (0x12)
  ▶ Handle: 0x0018 (Unknown: Unknown)
    Value: 02b9010131306d69727374206e616d65000000ee
    [Response in Frame: 3577]
```

```
0000  02 0d 00 1b 00 17 00 04  00 12 18 00 02 b9 01 01   ........ ........
0010  31 30 6d 69 72 73 74 20  6e 61 6d 65 00 00 00 ee   10mirst  name....
```

Value (btatt.value), 20 bytes

Packets: 9801 · Displayed: 59 (0.6%) · Load time: 0:0.149

Profile: Default

```
tx 02be0101303031000000000000000000008f ....001.............
tx 02bf0101000000000000000000000000000bf ....................
tx 02ea0101000000000000000000000000000ea ....................
tx 02c501013030000000000000000000000000c5 ....00............
tx 02ea0101000000000000000000000000000ea ....................
tx 02b9010131304669727374206e616d650000000ee ....10First name....
tx 02b3020131303130313034666f6f6f31303b3939388f ....10104fooo10;998.
tx 02b3020238373737373f303033303000000000087 ....87777?00300.....
tx 02ea0101000000000000000000000000000ea ....................
tx 02c501013130000000000000000000000000c4 ....10............
```

```
02 XX YY ZZ .. .. .. .. .. .. .. .. .. .. .. .. .. .. WW
```

XX – opcode

YY – total number of messages

ZZ – current message

WW – checksum

BlueZ Toolbox

`bluetoothctl`

`gatttool`

Optional:

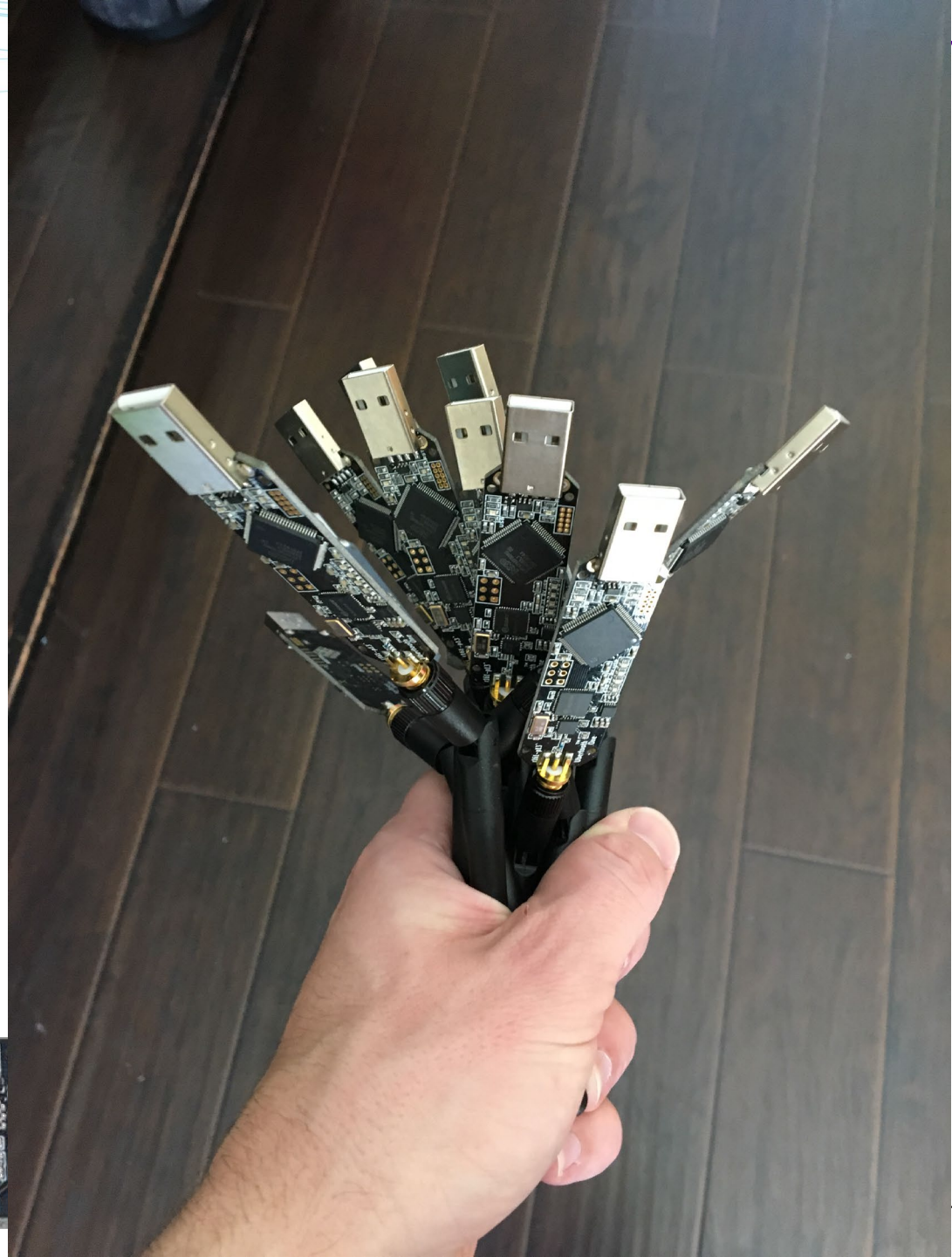
`btmon`

`btmgmt`

Active Reversing Procedure

1. Pair using bluetoothctl
2. Send commands with gatttool
3. Analyze command output
4. Fuzz anyway!

Approach 2: Ubertooth



Q: When should I use Ubertooth?

A: When you have no alternative

Case Study 4: Original Boosted Board

Dark days of 2014/2015: No app!

01_connection.pcapng

(btatt.opcode == 0x1d) || (btatt.handle == 0x0013)

Expression...

No.	Time	Source	Destination	Protocol	Lengt	Info
1830	0.015483	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1844	0.067138	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1850	0.029968	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1856	0.030261	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1862	0.029887	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1870	0.030028	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1878	0.029732	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1884	0.030005	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1891	0.030539	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1897	0.029804	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1908	0.059615	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1915	0.030316	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)
1938	0.119854	Unknown_0x6477b343	Unknown_0…	ATT	48	UnknownDirection Write Request, Handle: 0x001a (Unknown)

```
▶ Frame 1870: 48 bytes on wire (384 bits), 48 bytes captured (384 bits) on interface 2
▶ PPI version 0, 24 bytes
  DLT: 147, Payload: btle (Bluetooth Low Energy Link Layer)
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Write Request (0x12)
    Handle: 0x001a (Unknown)
    Value: 524330323332370d
    [Response in Frame: 1874]
```

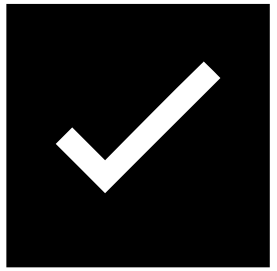
```
0000  00 00 18 00 93 00 00 00  36 75 0c 00 00 7e 09 00   ........ 6u...~..
0010  22 b9 4b 7b 80 7f 00 00  43 b3 77 64 0e 0f 0b 00   ".K{.... C.wd....
0020  04 00 12 1a 00 52 43 30  32 33 32 37 0d bd a8 be   .....RC0 2327....
```

Opcode (btatt.opcode), 1 byte

Packets: 2821 · Displayed:

Boosted board protocol

FFFF0

3FF1 

RC00000 idle

RC02002 dead man's trigger

RC02327

RC027D6

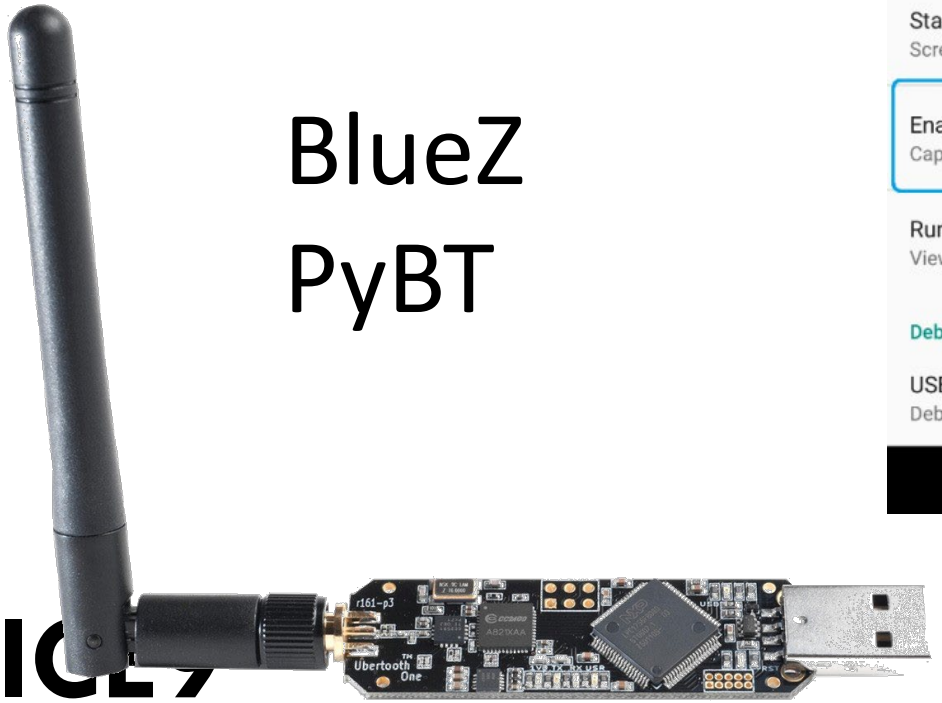
RC02AA6

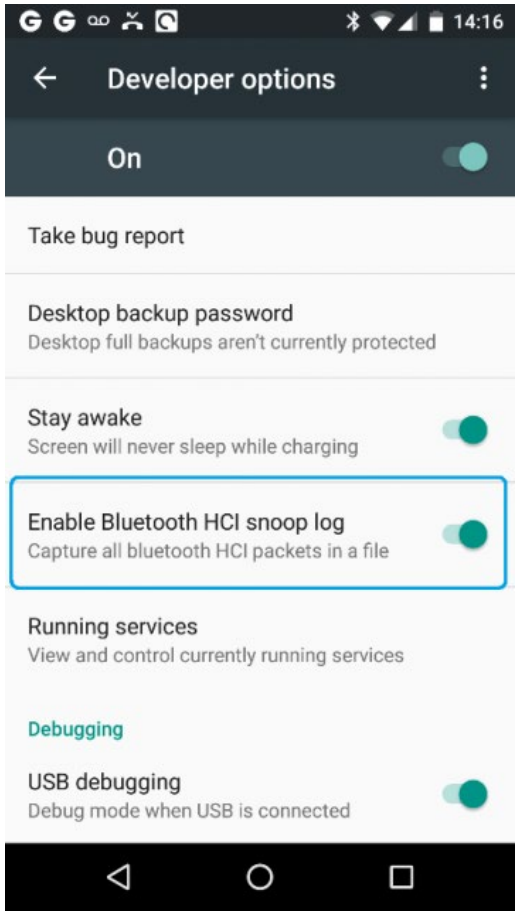
RC032F4

increasing throttle

Tools:

BlueZ
PyBT





Techniques:

- Do things in app / on device
- Export from Wireshark to JSON
- Smash face against keyboard repeatedly

“Apply” Slide

- Bluetooth devices continue to proliferate with limited security oversight
- This talk describes affordable tools and techniques you can use to analyze Bluetooth devices you currently use
- If you are developing a Bluetooth device, real attackers will use these techniques against your devices so take steps to harden your security posture