I Am Jack's Heart Monitor

Mike Ryan
Toorcon 2012 baby

WE TOORCON NOW

Oct 21, 2012

Overview

- Three parts
  - what is LE
  - how do we sniff it → demo!
  - security analysis

What is Bluetooth LE?

→ Introduced in Bluetooth 4.0

→ AKA Bluetooth Smart

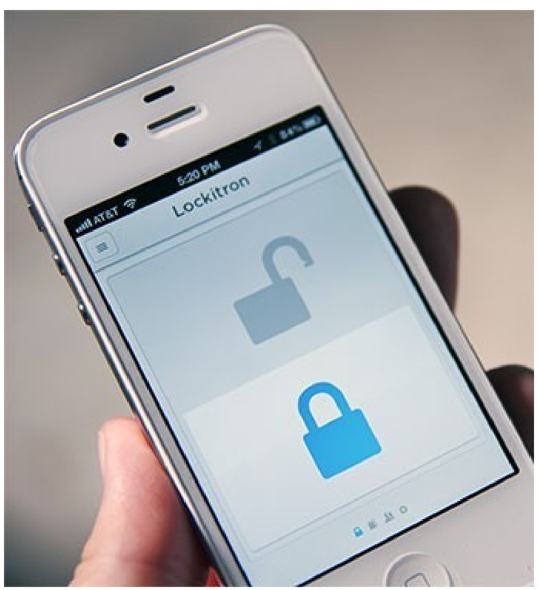
→ Almost completely different from classic Bluetooth

→ Designed to operate for a long time off a coin cell

Where is LE used?

- Sports devices (heart monitor, pedal cadence)
- Sensors (e.g., thermometer)
- Wireless door locks
- Upcoming medical devices

epic foreshadowing



How does LE compare to Classic BT?

- → Master/slave architecture
- → Different modulation parameters
- → Different channels (still in 2.4 GHz ISM)
- → Different channel hopping scheme
- → Different packet format
- → Different whitening

sniffing
Bluetooth
is
hard

sniffing

Bluetooth LE

is slightly

less hard

How do we sniff it?

Start at the bottom and work our way up:

GATT
ATT
L2CAP
Link Layer
PHY

PHY Layer

- GFSK modulation
- 40 x 1 MHz channels spaced 2 MHz apart
- Handled entirely by CC2400

RF → bits

Link Layer

LSB			MSB
Preamble (1 octet)	Access Address (4 octets)	PDU (2 to 39 octets)	CRC (3 octets)

Figure 2.1: Link Layer packet format

octets you say?

Link Layer

LSB			MSB
Preamble (1 octet)	Access Address (4 octets)	PDU (2 to 39 octets)	CRC (3 octets)

Figure 2.1: Link Layer packet format

What we have: Sea of bits

What we want: Start of PDU

What we know: AA

L2CAP and Beyond

```
06 0b 07 00 04 00 1b 11
00 16 58 b8 02 62 fb b2
```

→ RTFM

→ It's actually quite readable!

Example Packet

`06 0b`	
`07 00`	L2CAP length: 7
`04 00`	channel 4: LE Attribute Protocol
`1b`	Handle Value Notification
`11 00`	Attribute Handle
`16`	flags
`58`	**heart rate: 88 bpm**
`b8 02`	**RR-interval: 696 ms**
`62 fb b2`	

So we can turn RF into packets

→ Now what?

Let's follow connections!

How Connections Work

- Hop along data channels
- One data packet per timeslot

$3 \rightarrow 10 \rightarrow 17 \rightarrow 24 \rightarrow 31 \rightarrow 1 \rightarrow 8 \rightarrow 15 \rightarrow \ldots$
hop amount = 7

Following Connections

The four things you need to follow a connection are:

1. AA
2. CrcInit
3. Time slot length
4. Hop increment

How do I get these values?

Finding AA

- Sit on data channel waiting for empty data packets
- Collect candidate AA's and pick one when it's been observed enough

10001110111101010101
10011100000100011001
10000000000000001101
10100011000110000101

Not depicted: whitening!

Finding CRCInit

- → Filter packets by AA
- → Plug CRC into LFSR and run it backward

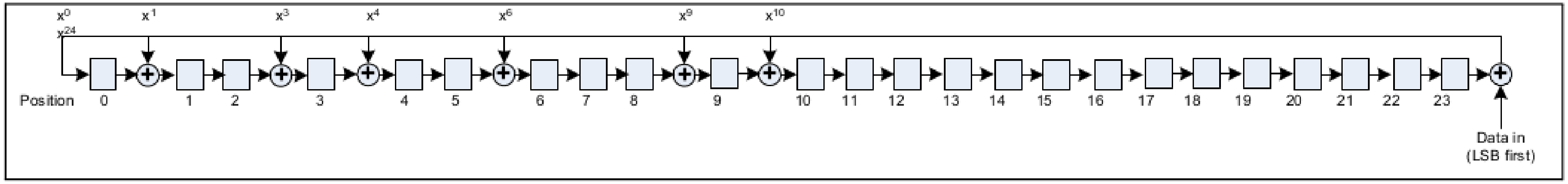


Figure 3.2: The LFSR circuit generating the CRC

See also "Bluesniff: Eve meets Alice and Bluetooth"

Finding time slot length

- Observation: 37 is prime
- Sit on data channel and wait for two consecutive packets

$$\frac{\Delta t}{37} = time\ slot\ length$$

Finding Hop Increment

- Start on data channel 0, jump to data channel 1 when a packet arrives
- We know hop interval, so we can calculate how many channels were hopped between 0 and 1
- I won't bore you with the math
- We use a LUT to convert that to a hop amount

MATH

Promiscuous: Summary

The four things you need to follow a connection are:

- ✓ AA
- ✓ CrcInit
- ✓ Hop interval
- ✓ Hop amount

Current Status

- Sniff new connections
- Sniff already-established connections (promiscuous)
- Jamming

- Grab the git!
- Available in Gentoo! (thanks Zero_Chaos)

- Everything implemented in-firmware

Demo

- demo
- demo
- demo
- demo
- demo
- demo
- demo
- demo
- demo

Security

- Good news: there is encryption
- Bad news: depending on your situation it's probably not very effective

Key Exchange

- Pairing mode determines temporary key (TK)
  - Just Works
  - 6 digit PIN
  - OOB

Not DH!

- Just works: no passive eavesdropper protection
- 6 digit PIN: easily brute forceable
- OOB provides the only meaningful security

Eavesdropping Scenario

- → Alice pairs with her brand new LE device
- → Eve observes pairing / key exchange
- → Just Works or 6 digit PIN: Eve recovers TK
- → With TK and pairing data: Eve recovers STK
- → With STK and key exchange: Eve recovers LTK

LTK = Session Key = GAME OVER

Well, not quite..

- Each connection uses a different nonce, so Eve has to witness connection setup
- The LTK is exchanged once and reused for many connections

Active Attacks

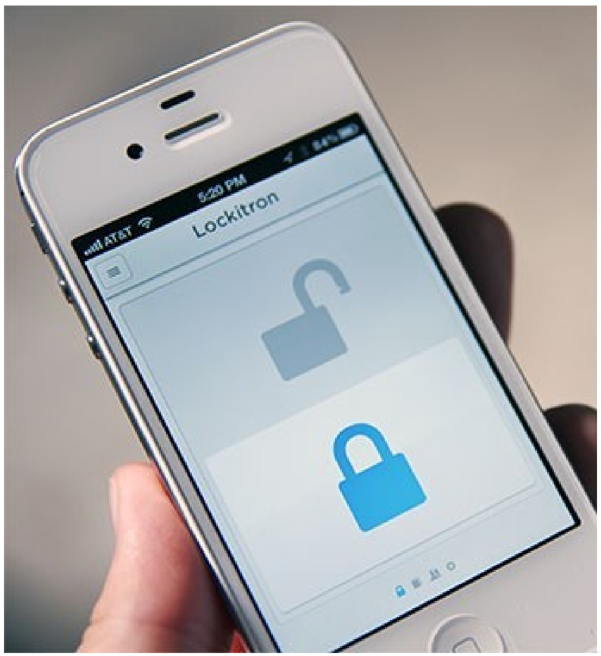
- How do you witness a connection setup?
  - Force a reconnect!
  - Should be as simple as jamming the connection
- What about connections that use a pre-shared LTK?
  - Inject message LL_REJECT_IND (reject LTK)

“My Bad”

“**None of the pairing methods provide protection against a passive eavesdropper** during the pairing process as predictable or easily established values for TK are used.”

“A future version of this specification will include elliptic curve cryptography and Diffie-Hellman public key exchanges that will provide passive eavesdropper protection.”

Why should I care about LE security?



Pacemaker hack can deliver deadly 830-volt jolt

Pacemakers and implantable cardioverter-defibrillators could be manipulated for an anonymous assassination

By Jeremy Kirk

October 17, 2012 12:40 AM ET

+ Briefcase More

IDG News Service - Pacemakers from several manufacturers can be commanded to deliver a deadly, 830-volt shock from someone on a laptop up to 50 feet away, the result of poor software programming by medical device companies.

The new research comes from Barnaby Jack of security vendor IOActive, known for his analysis of other medical equipment such as insulin-delivering devices.

Jack, who spoke at the Breakpoint security conference in Melbourne on Wednesday, said the flaw lies with the programming of the wireless transmitters used to give instructions to pacemakers and implantable cardioverter-defibrillators (ICDs), which detect irregular heart contractions and deliver an electric shock to avert a heart attack.

>2012

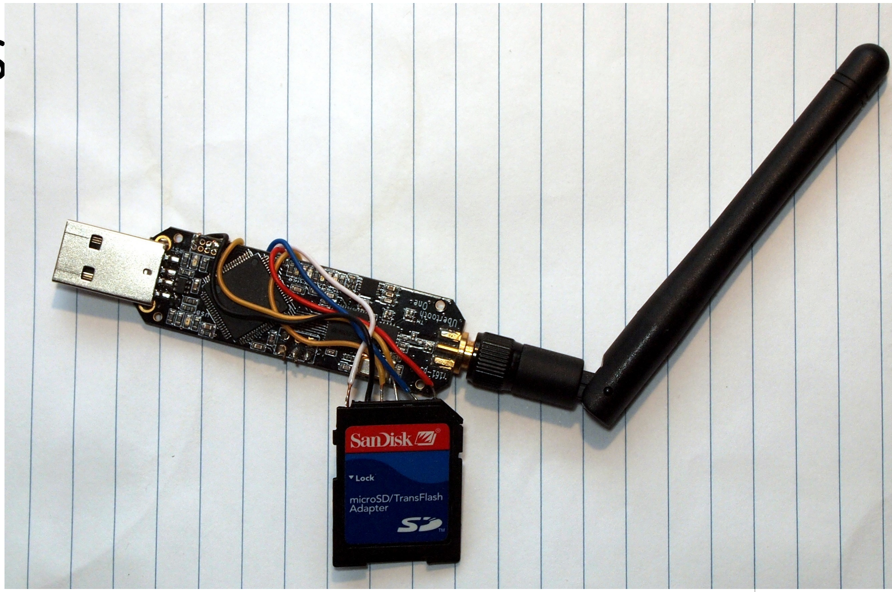
>wireless security still broken

Take-Away

- → LE security compromised by design
- → If security matters, use OOB pairing
- → Alternatively: BYOE   see also: The end-to-end principle

Future Work

- Kismet and Wireshark integration
- Demonstrate encryption attacks
- Master on dongle
  - MITM possible
- Slave on dongle
  - I really am Jack's heart monitor
- SD + battery
- Channel maps that don't use all 37 channels



Thanks

Mike Ossmann
Dominic Spill

Mike Kershaw (dragorn)
#ubertooth on freenode
bluez
Bluetooth SIG

Toorcon

Thank You

Mike Ryan

@mpeg4codec

mikeryan@lacklustre.net

http://lacklustre.net/

http://ubertooth.sf.net/

Related Work

- TI CC2540EMK-USB – $49
- BlueRadios BlueSniff™ – $249
  - "Only available to BlueRadios Clients who purchased our modules for use"
- Ellisys Bluetooth Explorer 400+LE – $N0,000

None support sniffing already-established connections!

Slave Device Lifecycle

- When connected
  - Hop along data channels
  - One data packet per timeslot
- When not connected
  - periodically announce existence on advertising channel
  - respond to requests from master

$3 \rightarrow 10 \rightarrow 17 \rightarrow 24 \rightarrow 31 \rightarrow 1 \rightarrow 8 \rightarrow 15 \rightarrow \ldots$

hop amount = 7