Are Security And Low Energy Incompatible?
BLE Security Triangle

Can we have all three?

- Tracking /Privacy
- Confidentiality /Authenticity
- Low Energy
First a diversion...
Meet KeySweeper, the $10 USB charger that steals MS keyboard strokes
Always-on sniffer remotely uploads all input typed into Microsoft Wireless keyboards.

by Dan Goodin - Jan 13 2015, 1:10pm PST

Samy Kamkar
MS Fail 2010

- Proprietary 2.4Ghz wireless protocol
- Broadcasts HID commands to anyone listening
- Luckily it's encrypted!
  - By XOR'ing packet with 5-byte keyboard mac address
  - How nice of MS to broadcast that too!
• All MS keyboard MAC address start with 0xCD
• In HID, the keycode always aligns to that byte
OK, but this is not BLE
Bluetooth Low Energy

- Single-hop protocol
- Physical, Link and Application layers
- Optimized for small exchanges and low energy:
  - ~24 byte exchanges; infrequently
  - μA power consumption
  - Can run for years on coin battery
Who the heck cares...

- Personal devices (fitness bands)
- Mobile payments
- Door locks, bike locks
- Medical devices
Bluetooth Low Energy

Unconnected

Advertising Packets

Connected (Paired?)

Generic Attribute Protocol (GATT)

L2CAP

Link Layers
Terms

- “Piconet” - star topology
- Peripheral (fitness band, watch, dead-bolt, etc)
  - Advertises and responds to connection requests
  - One central at a time
- Central (smart phone, laptop, gateway, etc)
  - Scans for advertisements and initiates connections
  - Many peripherals
Confidentiality/Authenticity
Advertisements in the Clear

• Advertiser's MAC address
• Optionally:
  – Available services
  – Human readable name
  – Security preferences
  – Connection preferences
  – Etc...
Establishing a Connection

- Advertise
- Connection Request
- Connection Response
- Advertise
- Advertise
Establishing a Connection

Piconet

Insecure*

*Unless previously paired
“Security Features”

- **Pairing**
  - Generating/exchanging shared secrets in a connection
- **Device authentication**
  - Verifying that two devices have the same shared key
- **Bonding**
  - Storing long term keys for use in future connections
  - “Trusted Device Pair”
Pairing – Two Phases

- Phase 1 – Selecting a key generation method
  - “Just Works”
  - “Passkey Entry”
  - “Numeric Comparison”
  - “Out of band”
- Phase 2 – Establishing a session key
Pairing Protocols

- **LE Legacy Pairing**
  - Obsolete as of December
  - No protection against passive eavesdropping
  - What everything uses

- **LE Secure Connections**
  - ECDH key generation (protects against passive eavesdropper)

- No pairing
  - What everything *actually* uses
LE Legacy Pairing

- Just Works
  - Temporary key = 0

- Passkey Entry
  - Temporary key = 6 digit passkey (< 20 bits of entropy)

- “...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.”

- “If the pairing information is distributed without an eavesdropper being present then all the pairing methods provide confidentiality.”
LE Legacy Pairing

“If the pairing information is distributed without an eavesdropper being present then all the pairing methods provide confidentiality.”
LE Secure Connections

• ECDH to derive a shared key

• Separate authentication step:
  – Just Works
  – Passkey Entry
    • User inputs passkey into both devices
    • Confirmation values generated independently (AES-CMAC)
  – Numeric Comparison
    • derive 6 digit independently from random commitments (AES-CMAC)
## Pairing: I/O Capabilities

<table>
<thead>
<tr>
<th>Local input capacity</th>
<th>Local output capacity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No input</td>
<td>No output</td>
<td>NoInputNoOutput</td>
</tr>
<tr>
<td>Yes/No</td>
<td>NoOutputNoOutput¹</td>
<td>DisplayOnly</td>
</tr>
<tr>
<td>Keyboard</td>
<td>KeyboardOnly</td>
<td>KeyboardDisplay</td>
</tr>
</tbody>
</table>

¹ This category is not applicable in the current context.

*Table 2.5: I/O Capabilities Mapping*
Minimum I/O Requirements

• Numeric comparison
  – Display + DisplayYesNo

• Passkey Entry
  – Keyboard + Keyboard or Keyboard + Display

• Just Works
  – Everything else
  – Unauthenticated
Bonding

- Exchange a long term key once paired
- In future connections, use LTK immediately

“Oh, I recognize you!”
Establishing a Connection

Connection Request

Connection Response

Advertise
Bonding Pro

- LE Legacy Pairing:
  - Connection only insecure the first time
  - Market for faraday cages ($$$)

- LE Secure Connections
  - ECDH expensive
  - Faster subsequent connections
  - Lower power for both peripherals and centers
Bonding Con: Everything can track you!
Tracking/Privacy: 3 Advertising Addresses

• Public:
  – Based on manufacturer, baked into device
  – Totally trackable

• Random “Static”:
  – Change as frequently as you want
  – Untrackable but can't bond

• Random “Private”
  – Change as frequently as you want
  – Bonded devices can recognize you, but no one else
Random Private Addresses

- Peripheral generates an IRK (Identity Resolving Key)
  - Provides to all bonded centrals
- Composed of:
  - Random part
  - “Hash” of Random Part:
    - AES(key, random part) mod 24

---

Figure 1.5: Format of resolvable private address
Random Private Addresses

- When central sees Random Private address
  1) Iterates through all stored IRKs
  2) $\text{AES}(\text{key}_i, \text{random part}) \mod 24 = \text{hash part}$

**Figure 1.5: Format of resolvable private address**
Summary

• Tradeoffs between
  – Confidentiality/Authenticity
  – Privacy
  – Low Energy

• Feasible in new spec, but is it realistic?

• What do actual systems do?