Security Considerations of Bluetooth Low Energy

(Sniffing and Understanding BLE Device)

Cyber Security Village Talk – Feb 2020

Dan Dumitrescu

NORTHERN TRUST

CPE provided by Grant Thornton NASBA #108891

OBJECTIVE

- Summarize what Bluetooth is and it's history
- Explain the workings of Bluetooth and its numerous uses
- Understand Bluetooth's security capabilities

AGENDA

- Overview
- *GAP – Generic Access Profile*
- *GATT – Generic ATTRIBUTE Profile*
- *BLUETOOTH lE - USED in A Variety of Small Devices*
- *Security Considerations*
- *Q&A*

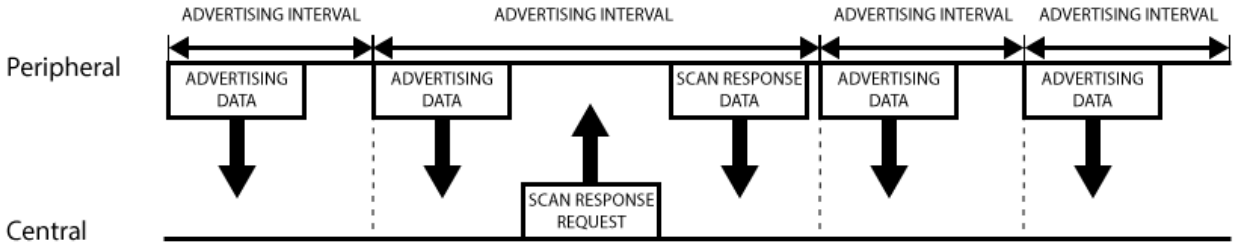
OVERVIEW

This is about Bluetooth but has nothing to do with teeth

- Bluetooth Low Energy (BLE or Bluetooth Smart) is a light subset of classic Bluetooth
- BLE is part of Bluetooth 4.0 core specifications
- There is some overlap with Bluetooth but has a different lineage
- Started by Nokia as Wibee before adopted by Bluetooth SIG
- Support for Bluetooth 4.0 and BLE (a subset of BT 4.0) was introduced in:
  - Android 4.3+ (many bug fixes in 4.4+)
  - iOS 5+ (iOS 7+ is better)
  - Apple OS X 10.6+
  - Windows 8+ (not supported in Win 7, XP or Vista)

GAP – GENERIC ACCESS PROFILE

GAP	Controls connections and advertising in Bluetooth
Peripheral	• small, low power, resource constrained devices that can connect to a much more powerful central device. Peripheral devices are things like a heart rate monitor, a BLE enabled proximity tag, etc.
Central Device	• usually the mobile phone or tablet that peripherals connect to with far more processing power and memory
Advertising and Scan Response Data	There are two ways to send advertising out with GAP. The *Advertising Data* payload and the *Scan Response* payload.



GATT – GENERIC ATTRIBUTE PROFILE

GATT	Defines the way that two Bluetooth Low Energy devices transfer data
Services	• used to break data up into logic entities, contain specific chunks of data called characteristics. A service can have one or more characteristics, and each service distinguishes itself from other services by means of a unique numeric ID called a UUID, which can be either 16-bit (for officially adopted BLE Services) or 128-bit (for custom services).
Characteristics	• lowest level concept in GATT transactions, encapsulates a single data point (though it may contain an array of related data, such as R/G/B values for an LED).
Attribute Protocol	used to store Services, Characteristics and related data in a simple lookup table using 16-bit IDs for each entry in the table
Exclusive Connections	BLE peripheral can only be connected to one central device (a mobile phone, etc.) at a time

BLUETOOTH LE - USED IN A VARIETY OF SMALL DEVICES

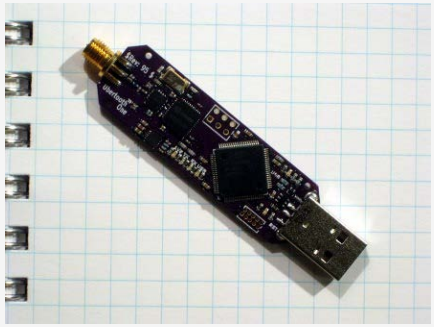
Home Automation
(Smart Light Bulbs, Smart Locks, etc)

Commercial and Industrial Devices
Door Controllers, URL and Location Beacons

Wearable Devices
Smart Watches/Bracelets, Heart Rate Monitors

Medical Devices and Toys

SECURITY CONSIDERATIONS

Capabilities	***BLE Supports encryption but is not always implemented or enabled***
Security	Low energy may equal low security (devices are resource and power constrained, may not implement all security specs)
Discoverability	• BLE devices - designed to broadcast MAC, UUID and service information at a predefined interval. Due to continuous advertisement, attacker can track the device and decode the broadcasting information using sniffers or smart phone.
MitM Attack	• attacker secretly reads and interprets the messages from the sender and delivers the message to the reader after interpreting/changing it
Passive Eavesdropping	Sniffers used to capture the communications between BLE “master” (i.e. phone) and BLE Devices, some are inexpensive UbertoothOne (~ $130) BlueFruit Sniffer (~25$)  

EXAMPLE #1: SHARPER IMAGE SBT5007 SMART LIGHT BULB

What is it	Smart-phone-controlled light bulb with built-in speaker

Services Exposed	• Appears as Human Interface Device (HID) to a BLE Device/Scanner with two “unknown” (custom) service UUID’s exposed:
	  • UUID 00006666-0000-1000-8000-00805f9b34fb
	  • UUID 00007777-0000-1000-8000-00805f9b34fb

Security	• **Encryption is disabled – LE Encrypt = False** in Link Layer Feature Response (LL_FEATURE_RSP) packet
Light ON	• Write 0x01 FE 00 00 53 83 10 00 00 00 00 00 50 FF 00 00 to Handle 0x0006
Red Light	• Write 0x01 FE 00 00 53 83 10 00 00 00 **FF** 00 50 00 00 00 to Handle 0x0006
Green Light	• Write 0x01 FE 00 00 53 83 10 00 **FF** 00 00 00 50 00 00 00 to Handle 0x0006
Blue Light	• Write 0x01 FE 00 00 53 83 10 00 00 **FF** 00 00 50 00 00 00 to Handle 0x0006
Conclusion	• RGB colors are controlled by changing the 11th, 9th and 10th byte written to handle #6

EXAMPLE #2: CHINESE F1 SMART WATCH/BAND

What is it	***Health-oriented Smart band with Heart Rate (Pulse), Pedometer, Blood Pressure (?) and Pulse Oximetry***
Characteristics	• Company: Reserved ID <0xFFF0> • Tencent Holdings Limited UUID • **Apple Notification Center** Service UUD (can act as an Apple notification device, showing iPhone notifications on the bracelet)
Security	• **Encryption is disabled** – LE Encrypt = False in LL_FEATURE_RSP packet
WhatsApp Alert	• Write 0x AB 00 05 FF 72 80 0A 01 to Handle 0x0011
Twitter Alerts	• Write 0x AB 00 05 FF 72 80 0F 01 to Handle 0x0011
Heart Rate Reading	Slave Writes 0x AB 00 05 FF 31 09 **4B** 04 to Handle 0x000E **4Bh = 75 – Heart Rate Displayed by the App**
Blood Pressure Reading	Slave writes 0x AB 00 05 FF 31 21 **79 46** …to Handle 0x00E **79h / 46h = 121 / 75 Blood Pressure Displayed by the App**
Twitter Notification	Write 0x AB 00 13 FF 72 80 0F 02 **46 69 6E 64 20 79 6F 75 72 20 66** … to Handle 0x0011 **“F i n d y o u r f “**

EXAMPLE #3: BLE SMART BEACONS (ONE MAY BE NEAR YOU)

What are they	***BLE devices broadcasting identifier to nearby portable electronic devices. Beacons enable smartphones and other devices to perform actions when in close proximity to them***
iBeacon	• In mid-2013 Apple introduced iBeacons and experts wrote about how it is designed to help the retail industry by simplifying payments and enabling on-site offers.
URIBeacon	• URIBeacons are different from iBeacons - rather than broadcasting an identifier, they send an URL which can be accessed immediately
Eddystone	• Eddystone is a Google's standard for Bluetooth beacons. It supports three types of packets, Eddystone-UID, Eddystone-URL, and Eddystone-TLM.[16] Eddystone-UID functions in a very similar way to Apple's iBeacon, however, it supports additional telemetry data with Eddystone-TLM. The telemetry information (such as battery voltage, beacon temperature, number of packets sent since last startup, and beacon uptime is sent along with the UID data.

DEMO (IF DEMO GODS AND TIME ALLOWS)

	BLE Sniffer Demo
Sniffers	Ubertooth One and Adafruit Bluefruit BLE Sniffer nRF Connect App, nRF Toolbox App, Google Physical Web App (not needed on newer Android device, URL will displayed as a notification)
BLE Light Bulb	• Demo the device • Attempt to Intercept Traffic • Attempt to change color from a Raspberry Pi
F1 Smart Watch	• Demo the Device • Attempt to Intercept traffic and send/see notifications
BLE Smart Beacons	Home-made BLE Smart beacons – beacon location (for iOS Beacons) or beacons sending a “secret” URL which can be “seen” directly on Android phones, with nRF Tools App, Physical Web App or other beacon apps

Q&A