Security in IoT Ecosystem

Module 5

Smart Bulb Pentest example

Prof.: Joaquín Recas

Table of Contents

First Step

- Attack Surface Mappin
  - Finding as much **information** as possible about the **device.**
  - Focus on the following categories
    1. Embedded device.
    2. Firmware, software, and applications.
    3. **Radio communications.**

HEKKE AC85-265V

- General specs:
  - Color Changing LEDs.
  - RGB+ Cool White
  - 12 fixed color: Red, Green, Blue, Orange, etc.
  - Dual Memory: help you preset what you set last time.
  - **Bluetooth connectivity**
  - **Android APP:** HappyLighting

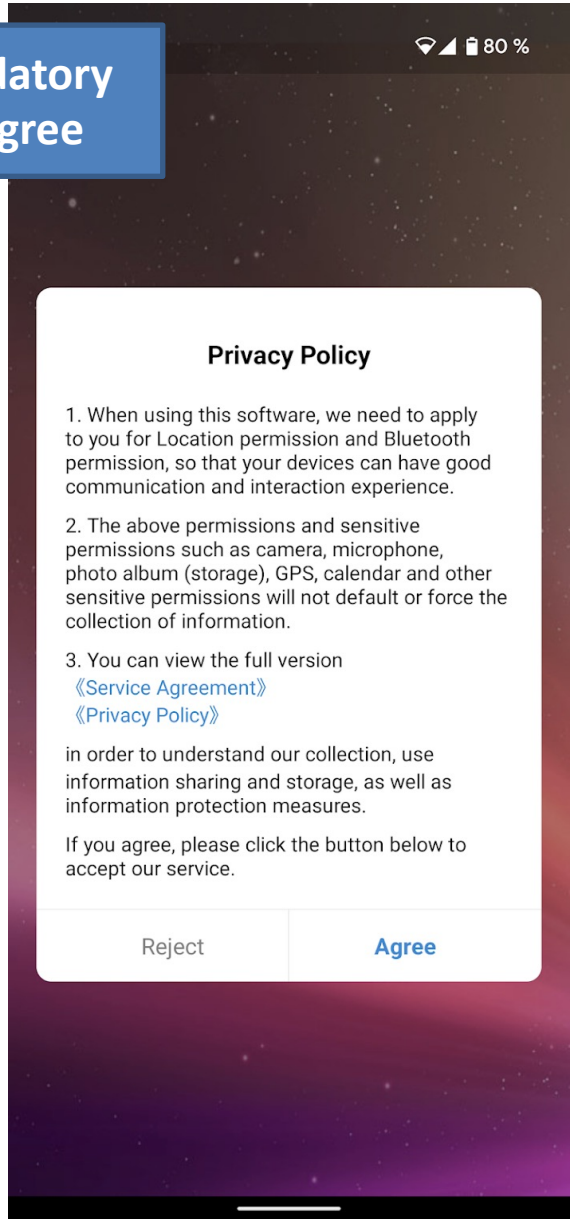
HappyLighting

- HappyLighting is a Bluetooth lamp control software:

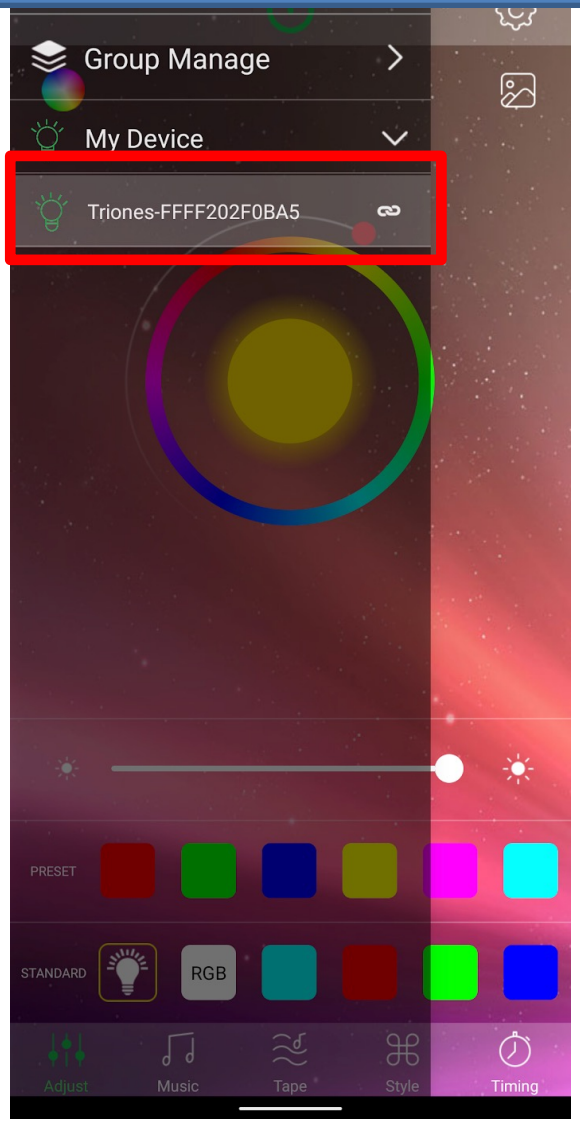
  1. You can control the HappyLighting Bluetooth lights for color matching.
  2. You can control the timing of HappyLighting Bluetooth lights.
  3. Can control the HappyLighting Bluetooth lamp to set the lighting mode.
  4. You can change the color of the light according to the music.

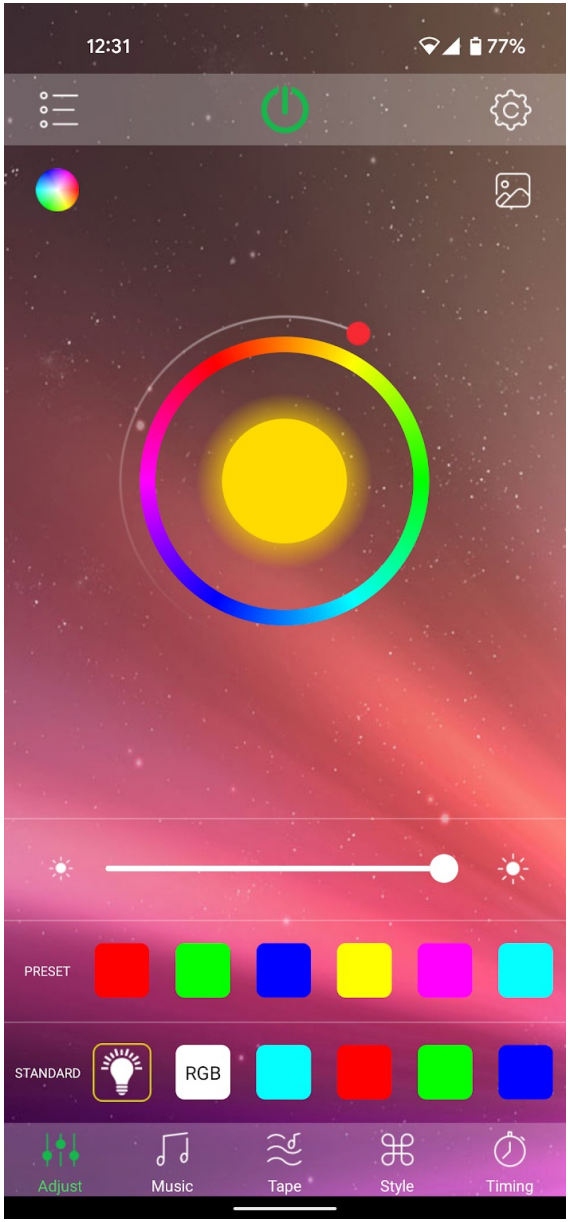
- If your Bluetooth lamp doesn't start with "***Triones***, *BRGlight, Dream, Light*" in the Bluetooth list, don't download this app, because this app only controls these Bluetooth lamp devices.

Mandatory to agree

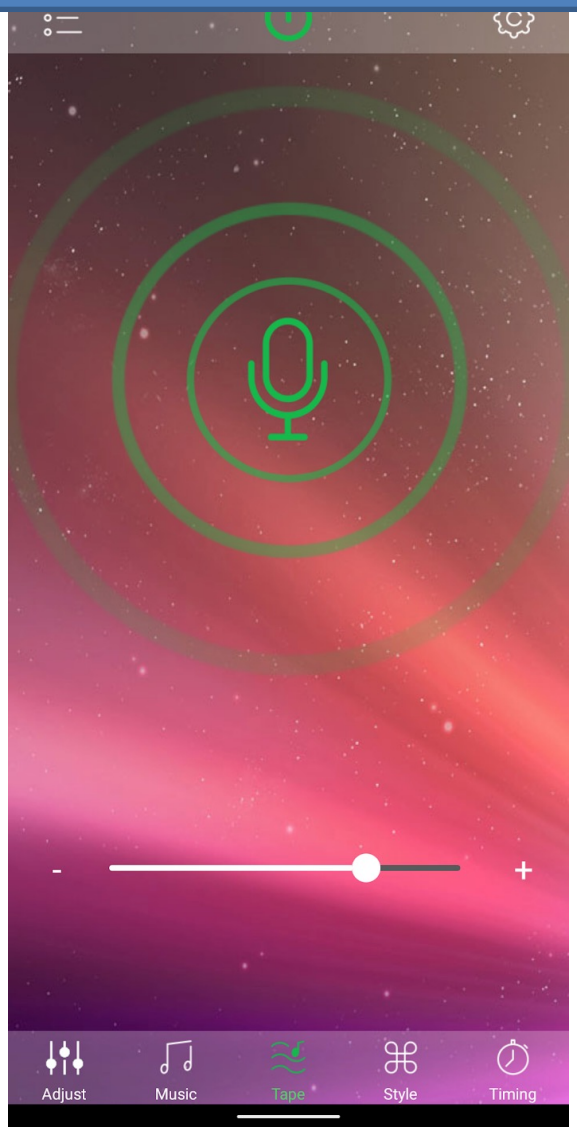


Device ID: Triones-FFFF202F0BA5





Record audio and access to files



HappyLighting

qh-tek

Showing permissions for all versions of this app

This app has access to:

Location

- approximate location (network-based)
- precise location (GPS and network-based)

Photos/Media/Files

- read the contents of your USB storage
- modify or delete the contents of your USB storage

Updates to HappyLighting may automatically add additional capabilities within each group. Learn more

Cancel

ADDITIONAL INFORMATION

Updated	Size	Installs
December 16, 2021	35M	1,000,000+
Current Version	**Requires Android**	**Content Rating**
1.6.1.7	4.3 and up	Everyone Learn more
Permissions	**Report**	**Offered By**
View details	Flag as inappropriate	qh-tek

Developer

service@qh-tek.com

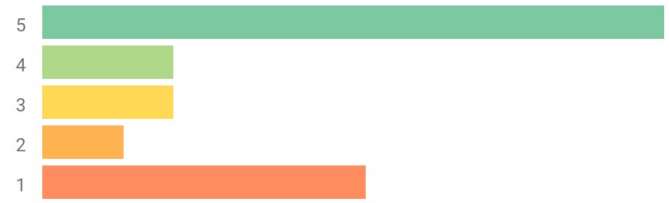
Privacy Policy

REVIEWS

Review policy and info

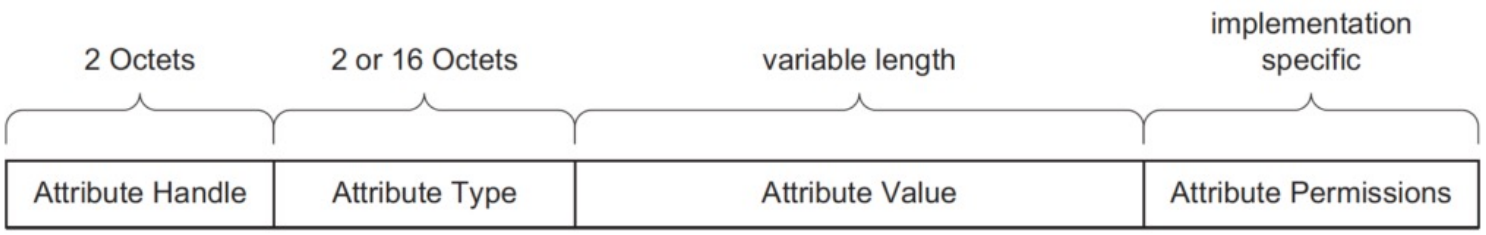
3.5

8,672 total



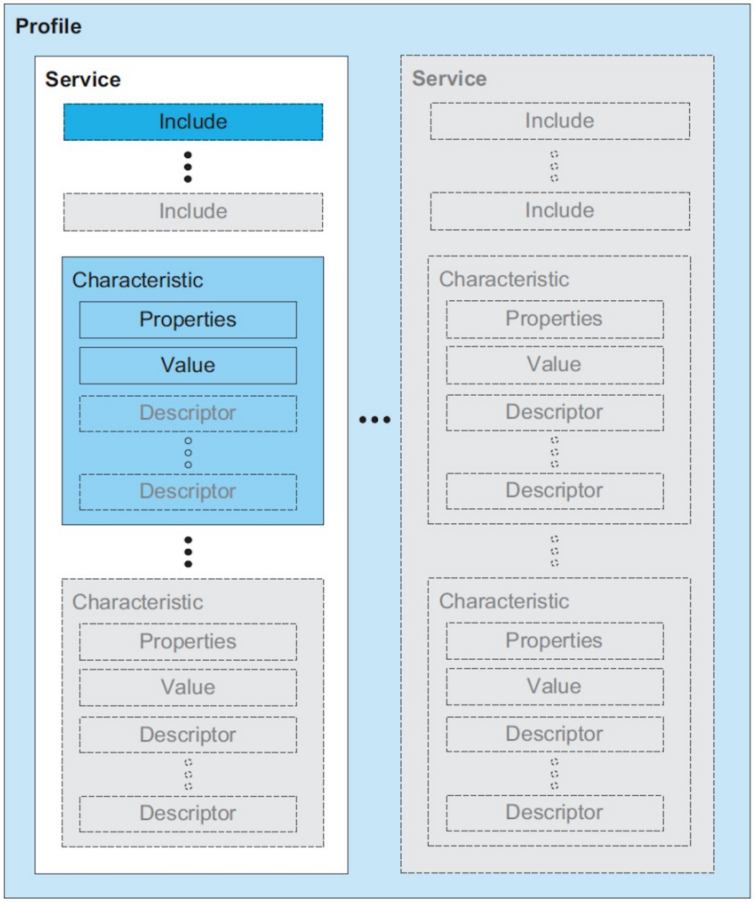
Basics of Bluetooth Low Energy (BLE)

- Attribute Protocol (**ATT**): defines how a server exposes its data to a client and how this data is structured:
  - **Server**: This is the device that exposes the data
  - **Client**: This is the device that interfaces with the server with the purpose of reading the server's exposed data and/or controlling the server's behaviour
- The data that the server exposes is structured as **attributes:**
  - **Attribute type (Universally Unique Identifier or UUID):** a 16/128b
  - **Attribute Handle:** a 16-bit value that the server assigns to each of its attributes
  - **Attribute Permissions:** Permissions determine whether an attribute can be **read** or **written** to, whether it can be **notified** or **indicated**, and what **security levels** are required for each of these operations.



Basics of Bluetooth Low Energy (BLE)

- The Generic Attribute Profile (GATT) is used after a connection has been established between the two devices:
  - A **service** is a grouping of one or more attributes, some of which are characteristics
  - A **characteristic** is always part of a service and it represents a piece of information/data that a server wants to expose to a client.
  - **Profiles** are much broader in definition than services. They are concerned with defining the behavior of both the client and server when it comes to services, characteristics, and even connections and security requirements.



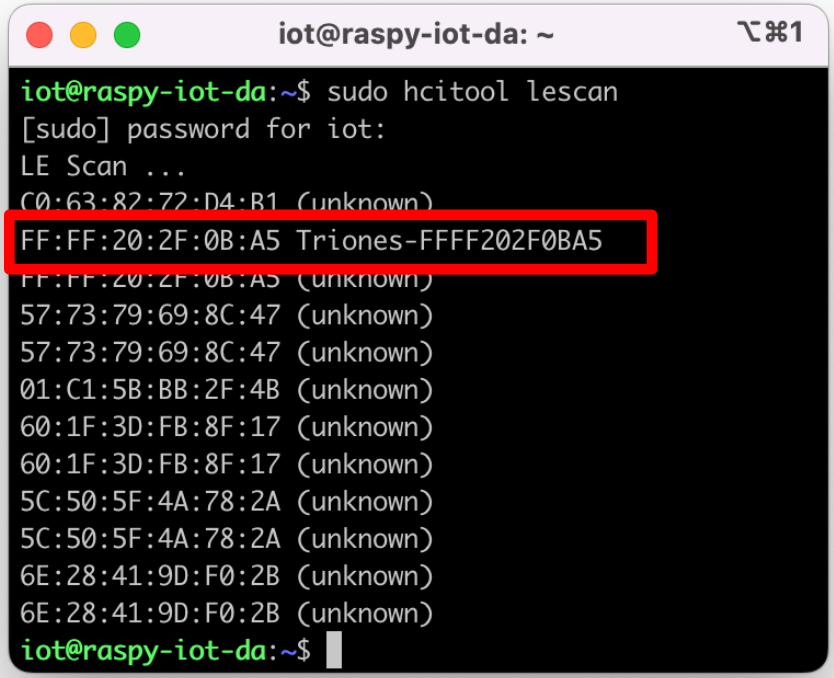
Video BLE Sec1

Video BLE Sec2

Bluetooth analysis: hcitool

- hcitool is used to configure Bluetooth connections and send some special command to Bluetooth devices:

  `scan`: Inquire remote devices. For each discovered device, device name are printed.

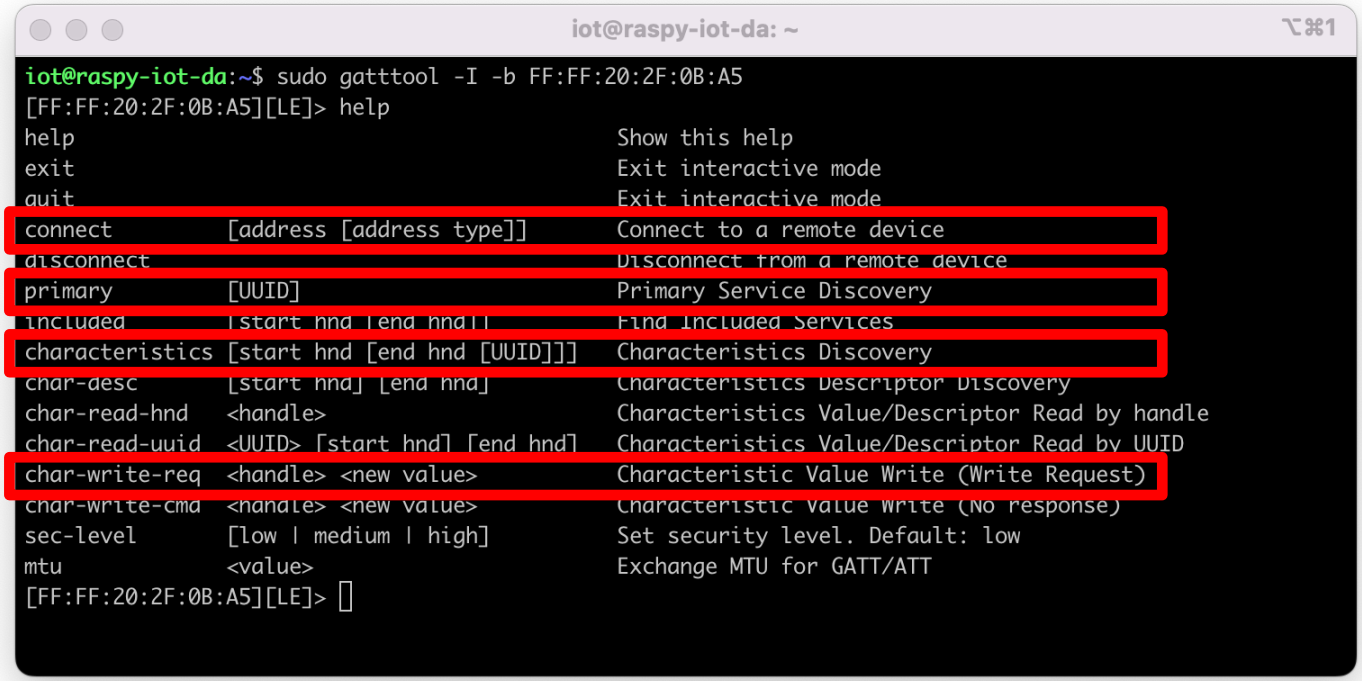


Bluetooth analysis: gatttool

- gatttool is tool that can be used to manipulate these attributes with a Bluetooth Low Energy device:

  `-b, --device=MAC` Specify remote Bluetooth address

  `-I, ---interactive` Use interactive mode



Bluetooth analysis: gatttool

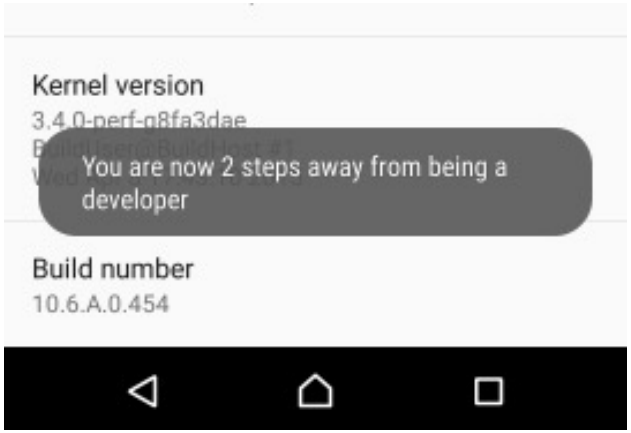
- Lets try a *Reply Attack*:
  - Reply attack: a form of network attack in which valid data transmission is maliciously or fraudulently repeated.
  - This is carried out either by an adversary who intercepts the data and re-transmits it.

Reply attack

- We need valid packets to send to the device
- We can obtain these packets by:
  - Sniffing the air transmission
    - Ubertooth One: open source 2.4 GHz wireless development platform suitable for Bluetooth experimentation.
  - **Record a valid transmission between a controlled android device and the Light Bulb**

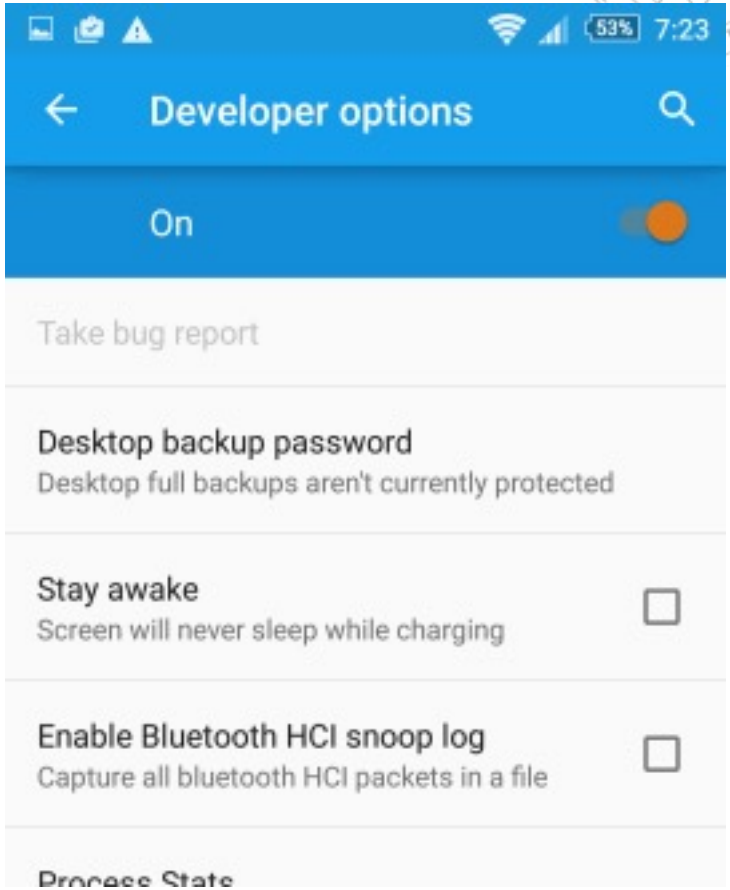
Bluetooth analysis

- Many Android phones are capable to log the full Bluetooth communication.
- Starting with Android 4.4 the log option is available on all phones in the Android developer settings.
- Enable Developer Settings
  - Usually the developer settings are invisible.
  - You can simply enable them by tapping 7 times on the build number in Android settings.



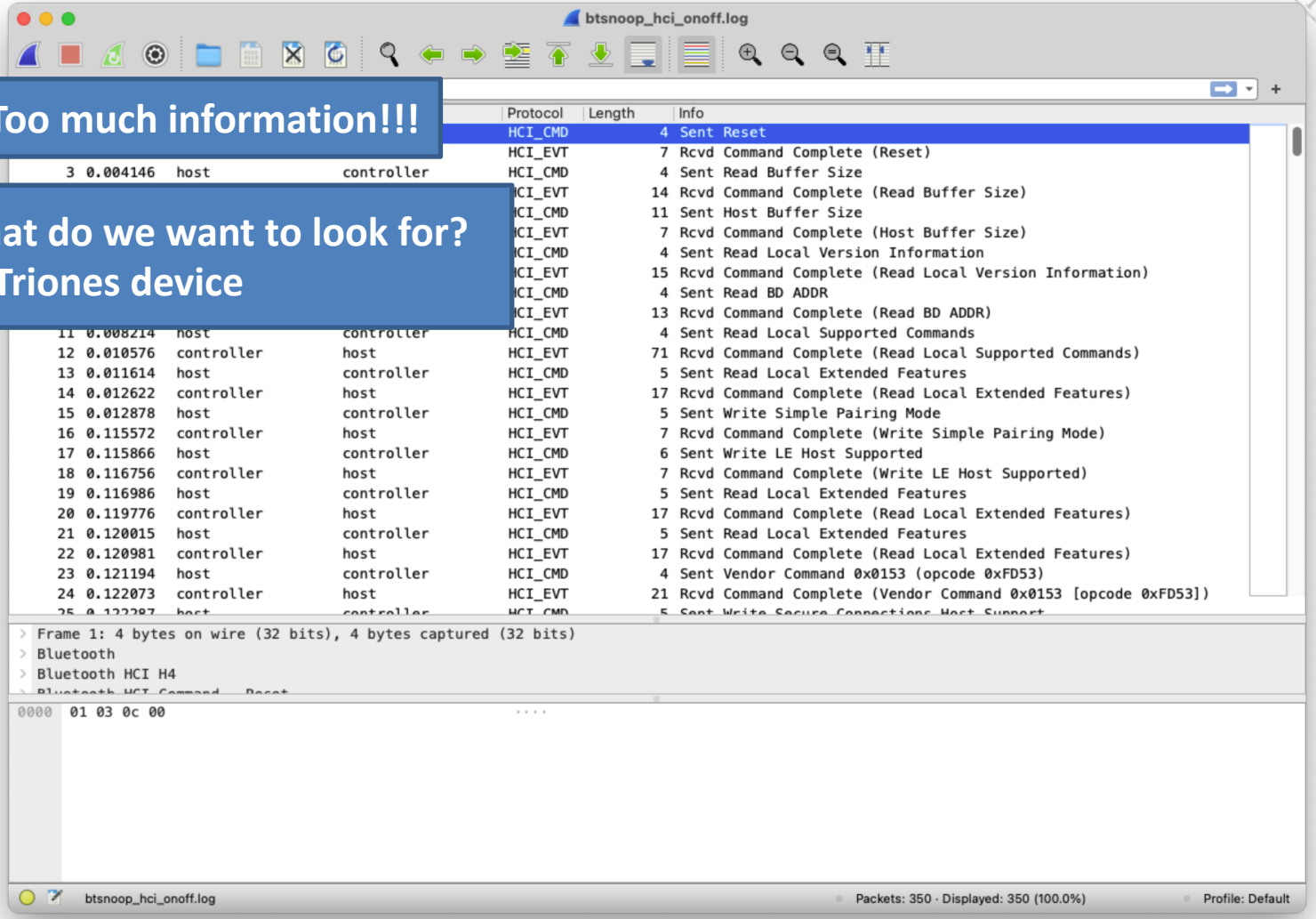
Bluetooth analysis

- Create Log:
  - Open the developer menu in Android settings.
  - You see a checkbox labelled "*Enable Bluetooth HCI Snoop Log*"
  - Start the log before you power on the car and stop the log before you send the file.
  - The log file is called *btsnoop_hci.log* and is usually stored in the root of the USB/SD storage.
- If you want to peek into the file you can use **WireShark**.



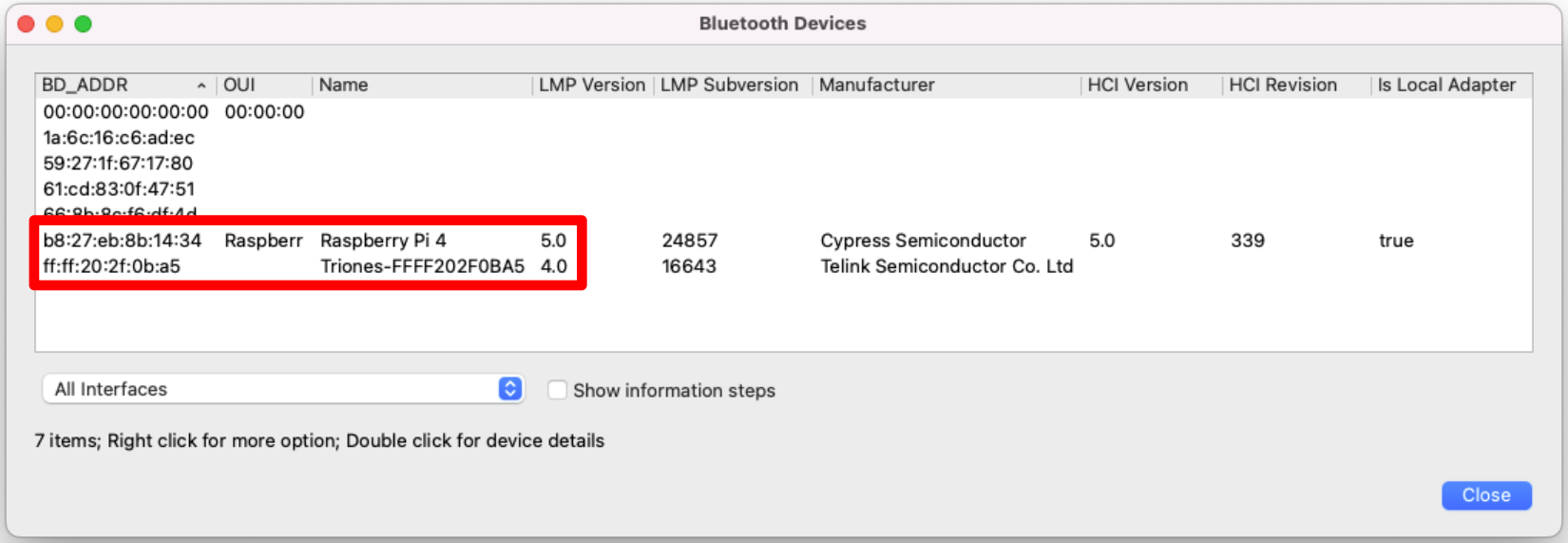
- **Wireshark** is the world's foremost and widely-used network protocol analyzer: It lets you see what's happening on your network.
- **Wireshark** has a rich feature set which includes the following:
  - Deep inspection of hundreds of protocols
  - Live capture and offline analysis
  - Multi-platform: Runs on Windows, Linux, macOS, Solaris, FreeBSD, NetBSD, and many others
  - Captured network data can be browsed via a GUI, or via the TTY-mode TShark utility
  - The most powerful display filters in the industry
  - Rich VoIP analysis
  - Live data can be read from Ethernet, IEEE 802.11, PPP/HDLC, ATM, **Bluetooth**, USB, Token Ring, Frame Relay, FDDI, and others (depending on your platform)
  - **Decryption support** for many protocols, including IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP, and WPA/WPA2
  - Etc.

WireShark: *btsnoop_hci.log*



WireShark: *BLE devices*

- Wireless -> Bluetooth Devices



WireShark filters

- Useful filters (manual):

Display Filter Reference: Bluetooth

Protocol field name: bluetooth

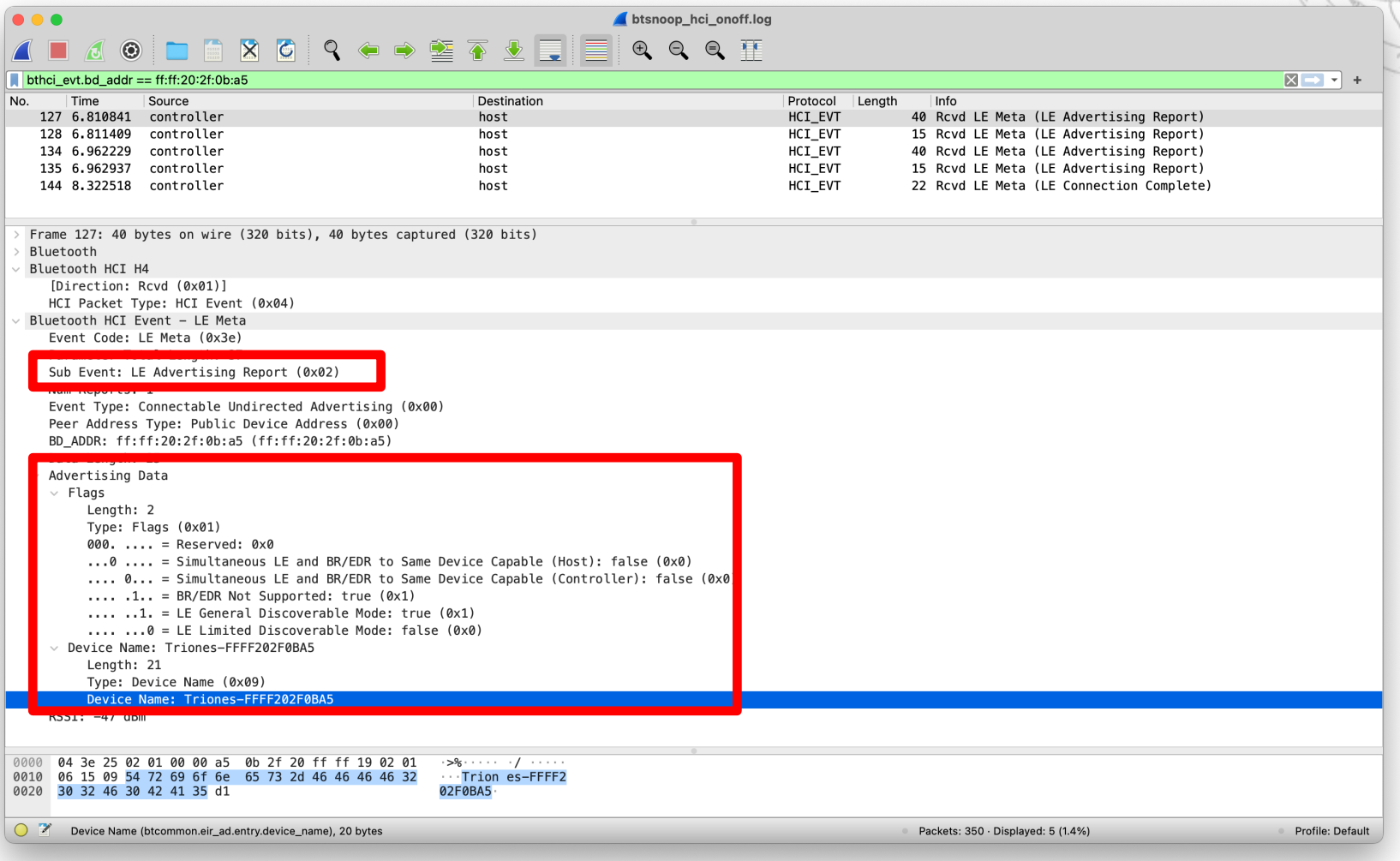
Versions: 2.0.0 to 3.6.3

Back to Display Filter Reference

FIELD NAME	DESCRIPTION	TYPE	VERSIONS
bluetooth.addr	Source or Destination	Ethernet or other MAC address	2.0.0 to 3.6.3
bluetooth.addr_str	Source or Destination	Character string	2.2.0 to 3.6.3
bluetooth.dst	Destination	Ethernet or other MAC address	2.0.0 to 3.6.3
bluetooth.dst_str	Destination	Character string	2.2.0 to 3.6.3
bluetooth.src	Source	Ethernet or other MAC address	2.0.0 to 3.6.3
bluetooth.src_str	Source	Character string	2.2.0 to 3.6.3

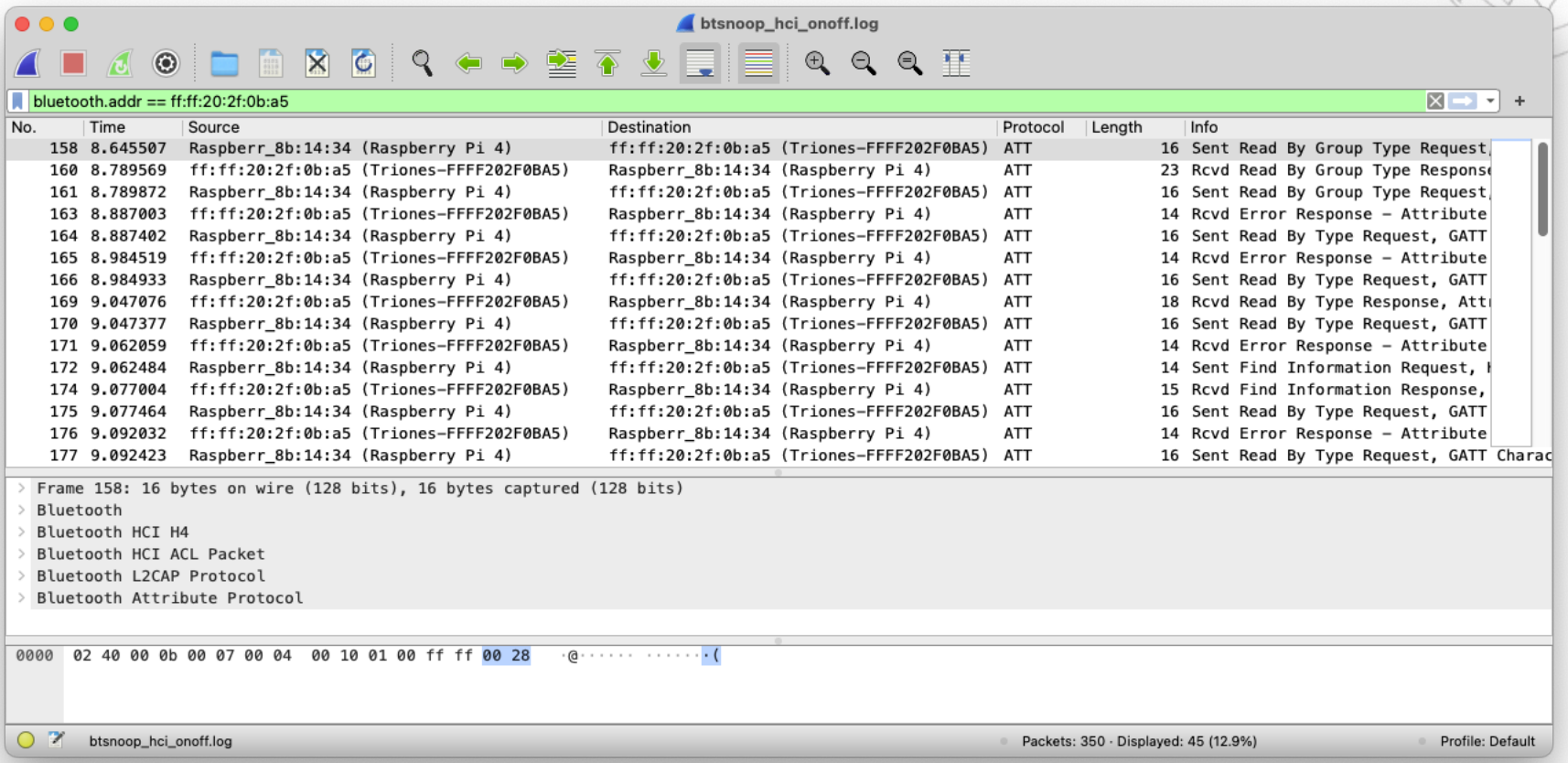
- More filters filters (Bluetooth Attribute Protocol):
  - `btatt.handle == <num>` # Interactions with handle
  - `btatt.opcode == 0x12` # Write request

WireShark: *btsnoop_hci.log*



bthci_evt.bd_addr == ff:ff:20:2f:0b:a5

Filter by address




```
bluetooth.addr == ff:ff:20:2f:0b:a5
```

Tasks

1. Download btsnoop_hci.log file for your group available in the SEC web page.
   - Multiple on/off sequences
   - Multiple color sets
2. Identify the MAC address on the Light Bulb
   - Wireless -> Bluetooth Devices
   - Identify *Triones* device
3. Filter the packets to see only “Write Request”
   - `btatt.opcode == 0x12`
   - In which handle is the Android (Raspi4) writing?
4. Identify the payload for reply attack

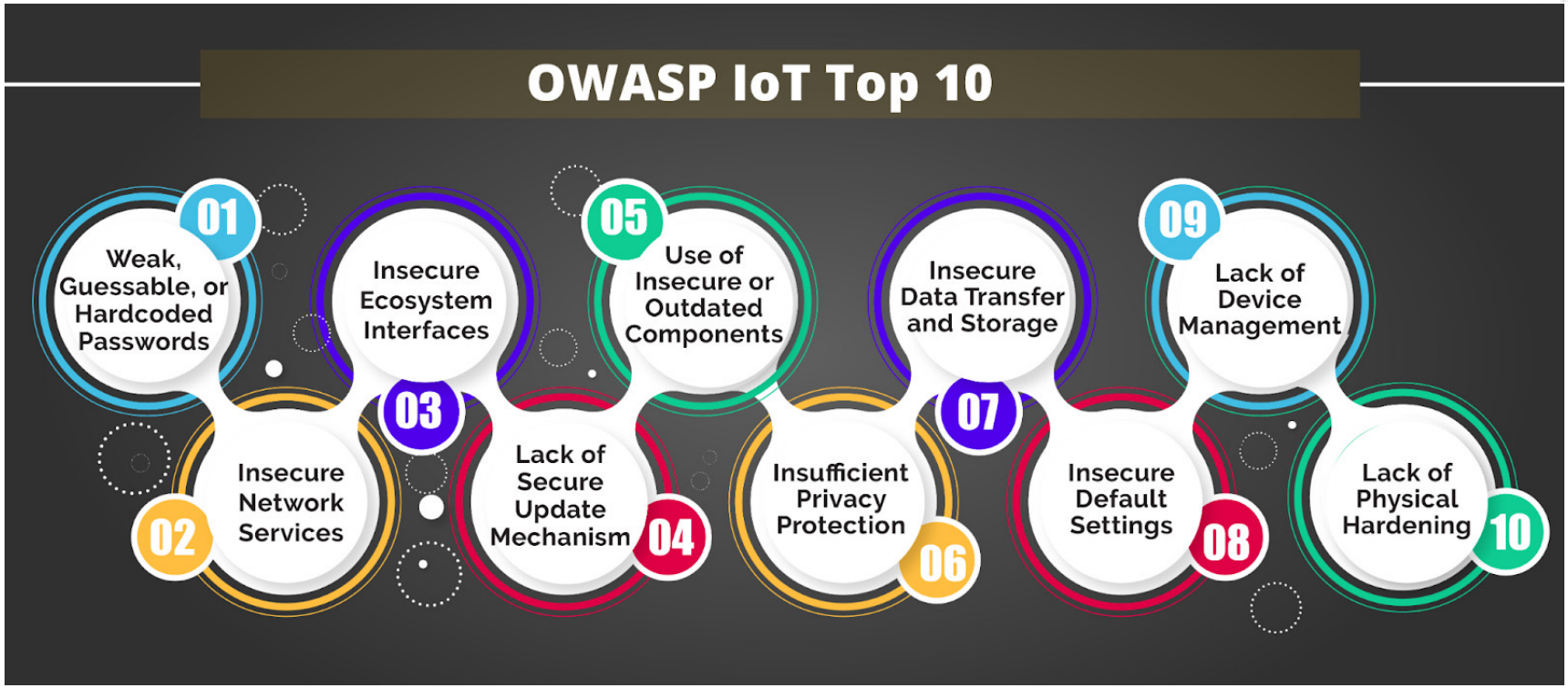
Table of Contents

OWASP Recommendations

- If we had followed the OWASP recommendations, would we have avoided these security holes?

- Task: identify in the OWASP Top 10 recommendations not followed and detected in this Pentest.

OWASP IoT Top 10



https://www.appsealing.com/owasp-iot-top-10/