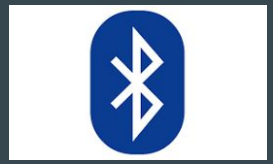
Bluetooth



...

It’s all pairing things of devices

hcitool leinfo (Mr-IoT)

...

iotpentest.com

Bluetooth History

. What is bluetooth ?

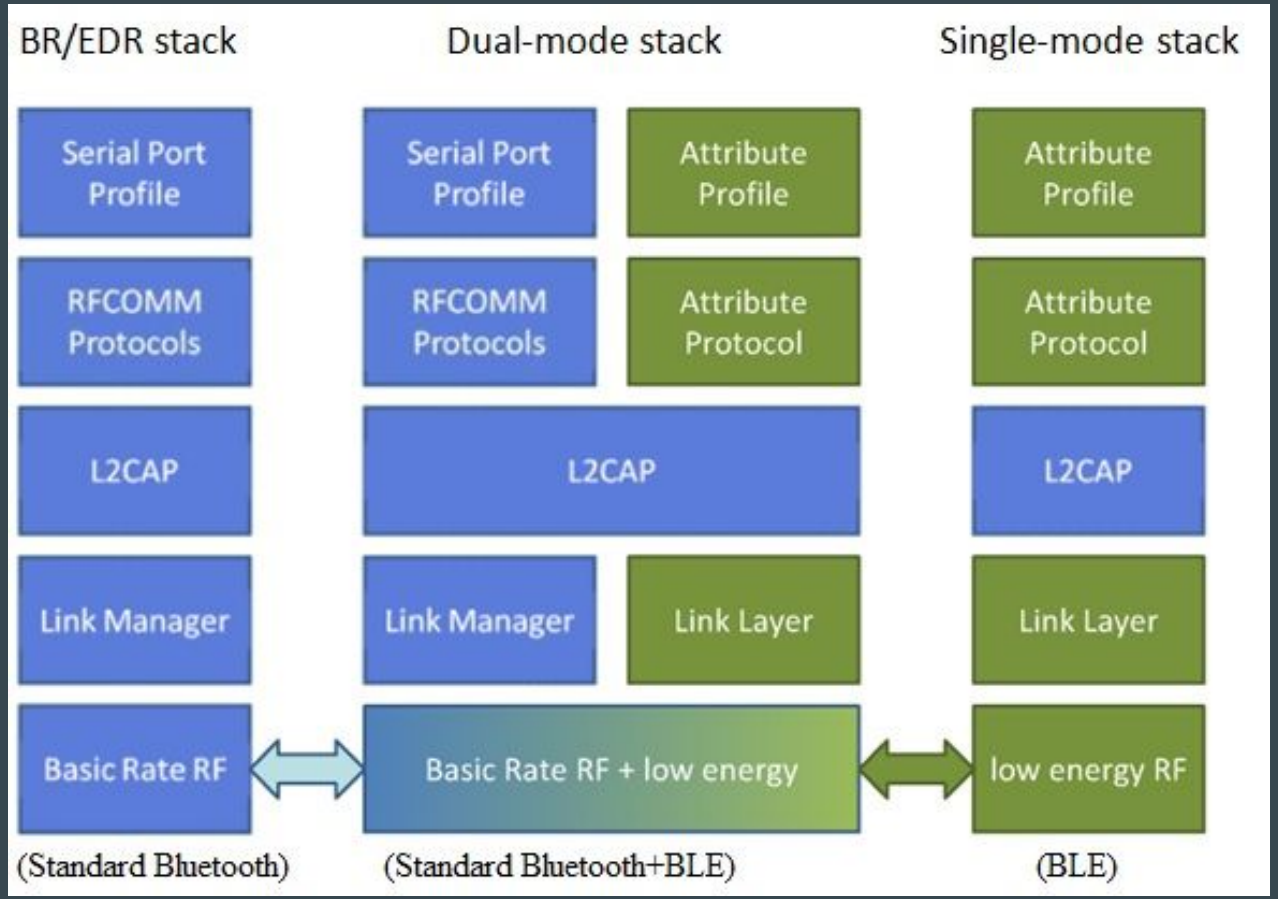
Bluetooth is a wireless technology standard for exchanging data between fixed and mobile devices over short distances using short-wavelength UHF radio waves in the industrial, scientific and medical radio bands, from 2.400 to 2.485 GHz, and building personal area networks (PANs). It was originally conceived as a wireless alternative to RS-232 data cables.

Nokia originally developed BLE for an in-house project called ‘WIBREE,’ which was later on, taken over by the Bluetooth SIG. BLE was conceived with an emphasis on better pairing speed and energy efficiency.

Bluetooth Versions

LMP	Bluetooth Version
0	Bluetooth 1.0b
1	Bluetooth 1.1
2	Bluetooth 1.2
3	Bluetooth 2.0 + EDR
4	Bluetooth 2.1 + EDR
5	Bluetooth 3.0 + HS
6	Bluetooth 4.0
7	Bluetooth 4.1
8	Bluetooth 4.2
9	Bluetooth 5
10	Bluetooth 5.1

Bluetooth Stack



- https://en.wikipedia.org/wiki/Bluetooth#Bluetooth_5.1
- Stadard Bluetooth 1 , 2, 3
- BLE 4,5
- Standard + BLE = Both Supports

BLE - Protocols

- HCI - Host Controller Interface
- L2CAP - Logical Link Control And Adaptation Protocol
- RFCOMM - Radio Frequency communication protocol
- SDP - Service Discovery Protocol
- BNEP - Bluetooth Network Encapsulation Protocol
- ATT - Attribute Protool
- SMP - Security Manager Protocol

BLE Profiles

- GAP - Generic Access Profile
- SPP - Serial Port Profile
- PAN - Personal Area Network
- HSP - HeadSet Profile
- HFP - Hands Free Profile
- GAP LE - Generic Access Protocol Low Energy
- GATT -- Generic Attribute Profile

Core concepts in BLE

Core concepts in BLE

There are two basic concepts in BLE.

- GAP - Generic Access Profile
- GATT - Generic Attribute Protocol

Core concepts ...

Generic Access Profile (GAP)

This is responsible for the connections and advertising in BLE. GAP is responsible for the visibility of a device to the external world and also plays a major role in determining how the device interacts with other devices.
The following two concepts are integral to GAP:

Peripheral devices : These are small and low energy devices that can connect with complex, more powerful central devices. Heart rate monitor is an example of a peripheral device.

Central devices : These devices are mostly cell phones or gadgets that have an increased memory and processing power.

Generic Attribute Protocol

Making use of a generic data protocol known as Attribute Protocol, GATT determines how two BLE devices exchange data with each other using concepts -

- Characteristics
- Services

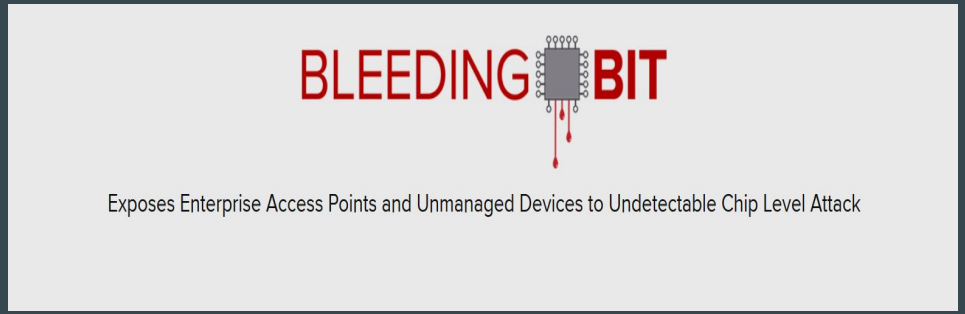
Services
A service can have many characteristics. Each service is unique in itself with a universally unique identifier (UUID) that could either be 16 bit in size for official adapted services or 128 bit for custom services.

Characteristics: Characteristics are the most fundamental concept within a GATT transaction. Characteristics contain a single data point and akin to services, each characteristic has a unique ID or UUID that distinguishes itself from the other characteristic. For example HRM sensor data from health bands etc.

BLE Vulnerabilities

- MAC Spoofing Attack
- PIN Cracking Attacks
- MiTM
- DOS
- Fuzzing
- Bruteforce

Test Cases about BLE



Xiaomi M365 Electric Scooter Hacked and Remotely Controlled

Understanding Bluetooth security

One of the best communication platform for the IoT devices to share and communicate and for operate device is Bluetooth low energy protocol

- Bluetooth standard – Non Secure one
- Bluetooth Low Energy – is Secure one
- Bluetooth 4.0 – vulnerable
- 4.1 – vulnerable
- 4.2 – vulnerable
- 5 , 5.1 – current in market (no - 5.0)

Pairing in bluetooth

Phase One:

Attribution Protocol (ATT) values. These live at layer 4 with L2CAP, and are typically not ever encrypted

Phase Two

The purpose is to generate a Short Term Key (STK). This is done with the devices agreeing on a Temporary Key (TK) mixed with some random numbers which gives them the STK.

Phase Three

If an LTK wasn't generated in phase two, one is generated in phase three. Data like the Connection Signature Resolving Key (CSRK) for data signing and the Identity Resolving Key (IRK) for private MAC address generation and lookup are generated in this phase.

Lets get hands dirty a little ... Not so Fast

Requirements to test BLE

Hardware

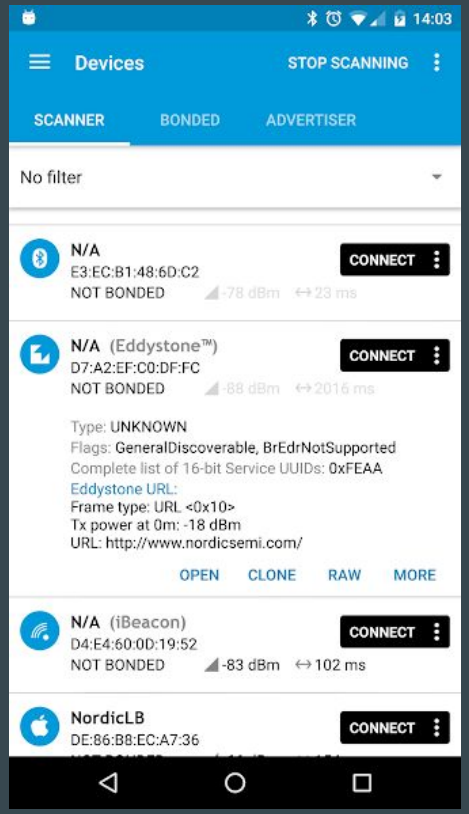
1. CSR 4.0 & Small Dongles
2. UD100
3. Ubertooth
4. Good configuration laptop
5. Any Cheap or Vulnerable device buy from the robu or banggood
6. ESP32 -- Microcontroller - Wifi and BLE

BLE FLAGS

Very Very Very Important

- 0x00 Display Only
- 0x01 Display Yes/No (both a display and a way to designate yes or no)
- 0x02 Keyboard Only
- 0x03 No Input/No Output (e.g. headphones)
- 0x04 Keyboard Display (both a keyboard and a display screen)
- 0x05-0xFF Reserved

NRF Connect APP - Android



Tools need to be installed ..

1. Bluez (hcitool)
2. Gatttool
3. Btproxy
4. Bettercap
5. Wireshark
6. Btlejack
7. Btle juice
8. NRF Connect APP
9. Etc

Depends on requirement we can install the tools

Tools which is going to use

hcitool:

It makes use of the host controller interface in a laptop to communicate and read/write changes to BLE devices. hcitool is therefore, useful in finding out the available victim BLE device that advertises, and then in changing the values after connection.
The values/data can only be changed if one knows the service and characteristic the data is coming from. In order to find out the relevant services and characteristics, one may use a gatttool.

gatttool:

As mentioned in the previous paragraph, gatttool is mainly helpful in finding out the services and characteristics of an available BLE device so that the victim's data can be read/written according to the attacker.

Walkthrough Commands

--- hcitool -h and man hcitool

--- gatttool -h and man gatttool

Lets get little understand about the commands

Usage

hciconfig : Used to list all the attached BLE adapters.

hciconfig hciX up : Enable the BLE adapter named hciX.

hciconfig hciX down : Disable the BLE adapter named hciX.

hcitool lescan : Scan for BLE devices in the vicinity.

gatttool -I : Launches gatttool in an interactive REPL like mode where the user can various issue commands as listed below.

connect <addr> : Connect to the BLE device with the specified address.

gatttool -t random -b <addr> -I : Connect to the device using a random address.

Primary

Characteristics

Start scan devices

. turn on the vulnerable device (smart band or smart watch)

-- run the below command

##hcitool lescan

Note the MAC address of the device

BLE Exploitation

Try to connect the device

Try to get the information about the device

BLE Exploitation

Connect with gatttool

##gatttool -I connect <ble address>

##primary

##characteristics

BLE Exploitation

Identify the read/write characteristics

##char-desc

Filter displayed handles

##char-desc 01 05

Find read characteristic

##char-read-hnd <handle>

BLE Exploitation

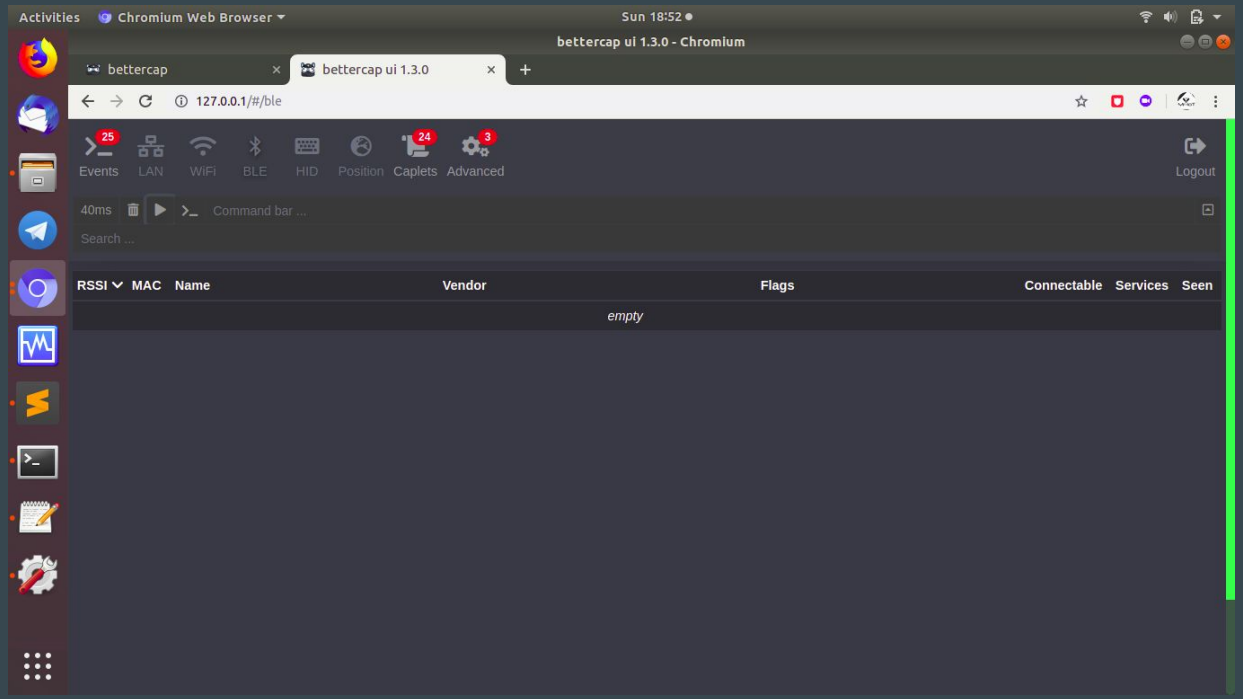
Write the data to characteristic

##char-write-req (or) char-write-cmd

A Successful write request shows hack a vulnerable device

Bettercap With UI

sudo bettercap -caplet http-ui



Thank You