Security review of proximity technologies: beacons and physical web

Renaud Lifchitz (renaud.lifchitz@digitalsecurity.fr)

BlackAlps– Switzerland – November 15-16, 2017

digital security | econocom

Outline

- Introduction to proximity technologies
- iBeacon security
- Physical Web security
- Web Bluetooth security

Speaker's bio

- French senior security engineer
- Main activities:
  - Penetration testing & security audits
  - Security research
  - Security trainings

- Significant security studies about:
  contactless debit cards, GSM geolocation, blockchain, RSA signatures, ZigBee, Sigfox, LoRaWAN, Vigik access control and quantum computation

https://speakerdeck.com/rlifchitz

About Digital Security

- Company founded in 2015 by a group of experts with the support of Econocom Group
- Provides advanced services in security audit, consulting and support
- Our expertise combine traditional security for infrastructure and application, and skills oriented to the ecosystem of **connected objects**
- Has created the CERT-UBIK, first European CERT™ specialized on **IoT security** (OSIDO monitoring service)
- Has a laboratory for studying new technologies, protocols and specific operating systems

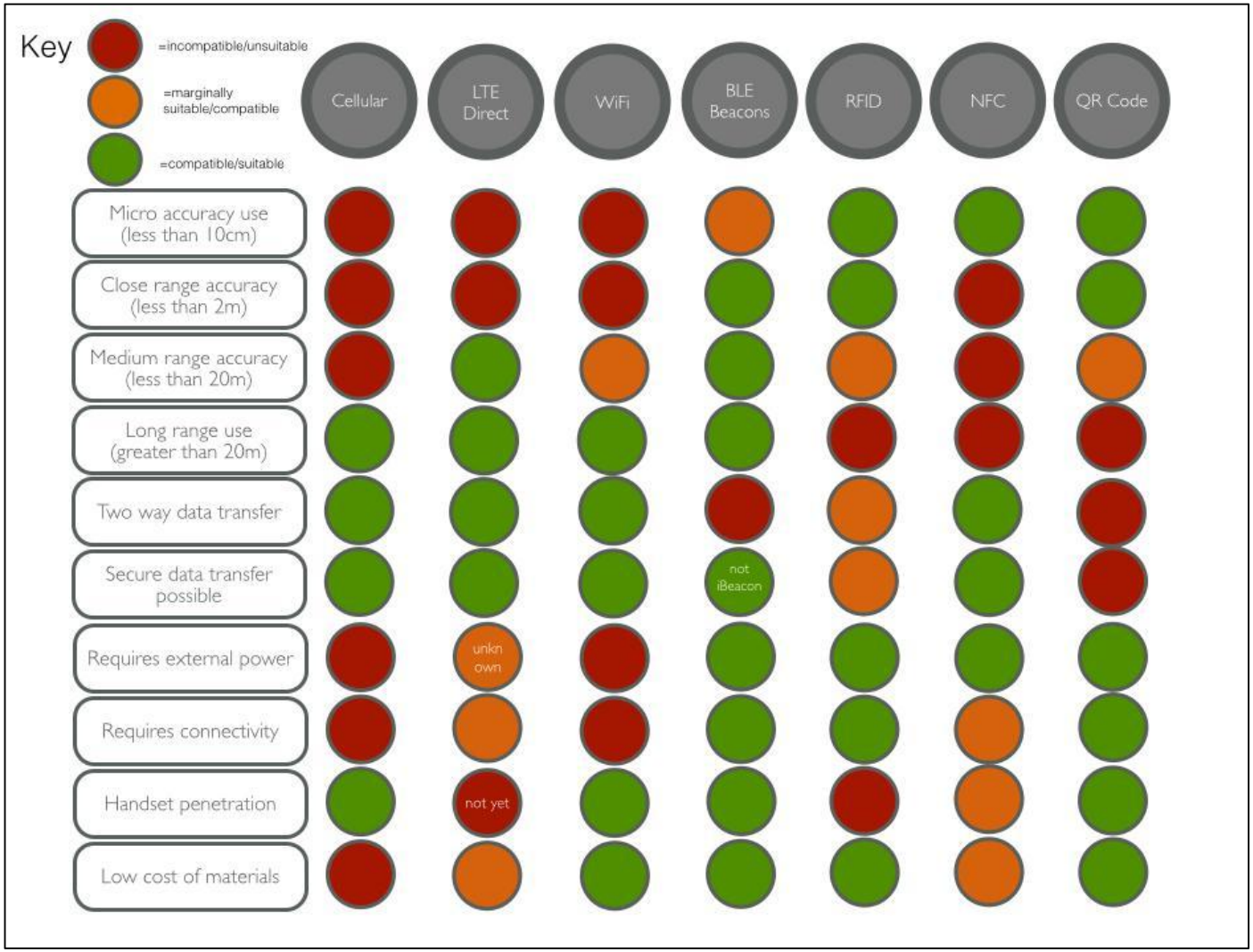
Introduction to proximity technologies

digital security

Use cases (1/2)

- Indoor location
- Proximity marketing
- Check-in coupons
- Contactless payments
- Contextual information
- Access control

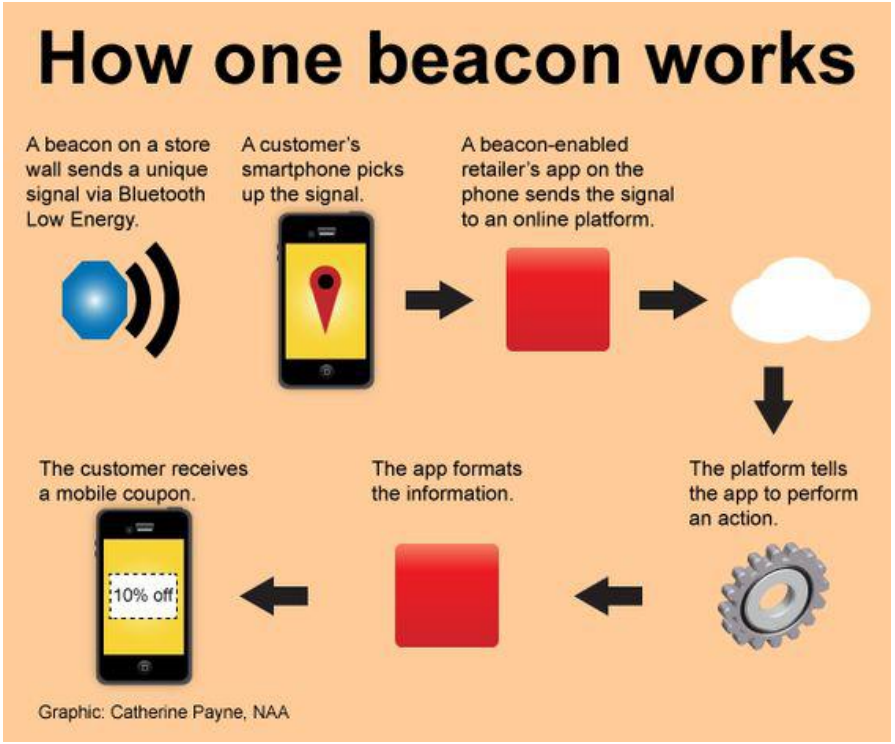
Use cases (2/2)



"A Guide to Bluetooth Beacons", september 2014, GSMA

iBeacon

- Apple technology
- Based on Bluetooth Low Energy (Bluetooth >= 4.0)
- Broadcasts
- Applications can recognize the broadcasted UUID and react accordingly



EddyStone

- Google open source format, Apache v2.0 license: https://github.com/google/eddystone
- Also based on BLE broadcasts
- Unlike iBeacon, 4 different frame formats:
  - UID: a unique 16-byte Beacon ID composed of a 10-byte namespace and a 6-byte instance
  - URL: a URL using a compressed encoding format
  - TLM: telemetry information about the beacon itself such as battery voltage, device temperature, and counts of broadcast packets.
  - EID: an encrypted ephemeral identifier that changes periodically for use in security and privacy-enhanced devices

Physical Web

- 2014 project from Google's Chrome team
- Uses Eddystone beacon protocol
- Open source approach
- Replaces the QR code
- Allow physical devices to **broadcast a URL** around:
  - to provide an access to information
  - to interact or remote control the device
  - standard: no need for a different app each time
- Apps: Google Chrome, "Nearby Notifications", compatible Android & iPhone apps
- Official web site: https://google.github.io/physical-web/

An interesting hacking device: the RuuviTag beacon

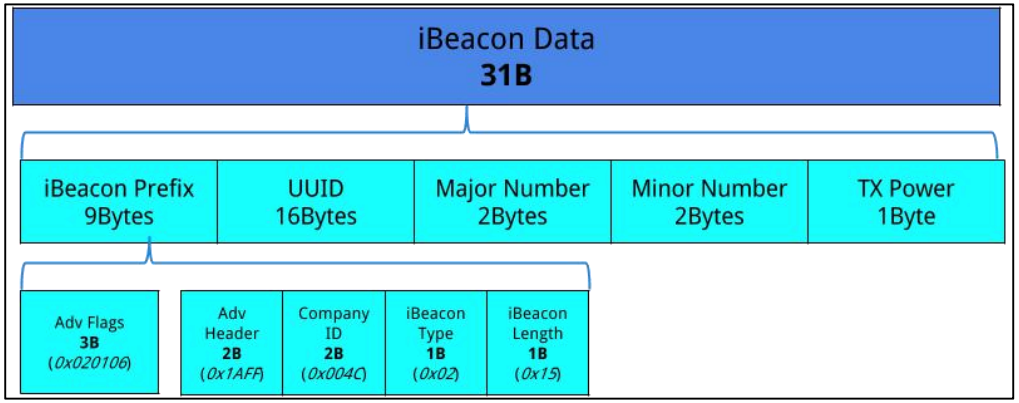
- Nordic nRF52832 SoC
- Sensors: temperature, humidity, air pressure, accelerometer
- 2 buttons, 2 LEDs, NFC-A tag, SWD debugging, FOTA programming
- 45 mm diameter PCB, IP67 enclosure
- 1000mAh battery
- BLE compatibility: iBeacon & Eddystone
- C & JavaScript programming (Espruino)
- Long range RF antenna (500-1000m!)

iBeacon security

digital security

iBeacon basics & frame format

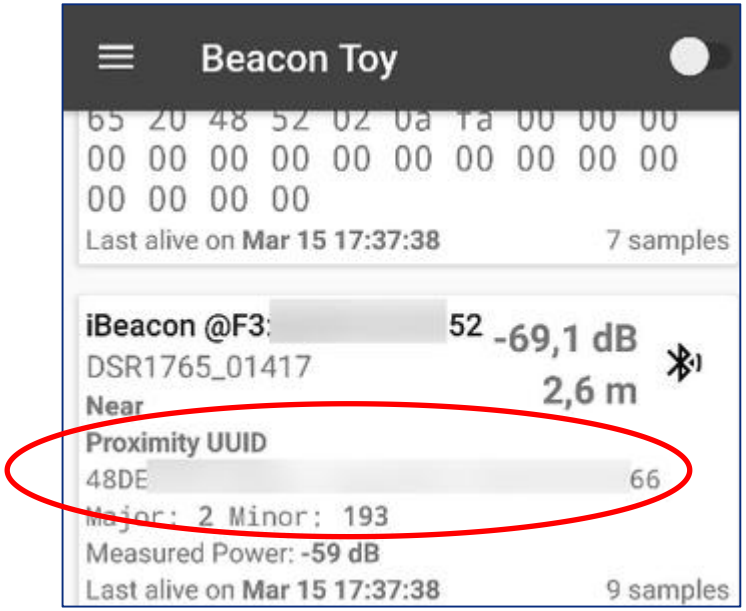
- iBeacon frames are sent **in plaintext**
- Important data for apps:
  UUID, major number & minor number
- Sniffing, replaying and cloning is easy...



Byte offset	Default value	Description	Properties
0	0x02	Data length – 2 bytes	constant preamble
1	0x01	Data type – flags	constant preamble
2	0x06	LE and BR/EDR flag	constant preamble
3	0x1a	Data length – 26 bytes	constant preamble
4	0xff	Data type - manufacturer specific data	constant preamble
5	0x4c	Manufacturer data	constant preamble
6	0x00	Manufacturer data	constant preamble
7	0x02	Manufacturer data	constant preamble
8	0x15	Manufacturer data	constant preamble

Beacons & iBeacon sniffing (1/2)

- Sniffing broadcast traffic is easy!
- Apple restricts arbitrary UUID listening...
- Using a smartphone:
  - Android tools: Beacon Toy, nRF Connect, Locate Beacon, ...



Beacons & iBeacon sniffing (2/2)

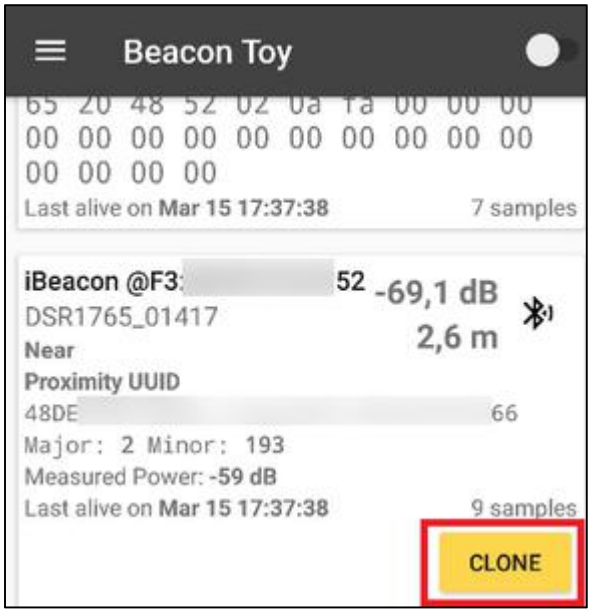
- Or using a computer:
  - Proprietary Windows tool Nordic nRF Sniffer
  - Open source Linux tool hcidump (with hcitool and optionally btmon for RSSI):

```
$ sudo hcitool lescan --duplicates &
$ sudo hcidump --raw -X -t
HCI sniffer - Bluetooth packet analyzer ver 5.37
device: hci0 snap_len: 1500 filter: 0xffffffffffffffff
2017-11-14 22:36:33.494792 > 0000: 04 3e 2b 02 01 03 01 24  b4 8c 20 46 29 1f 1e ff
.>+....$.. F)...
  0010: 06 00 01 09 20 00 09 a8  d0 5a 56 ad 2c 40 92 f5  .... ....ZV.,@..
  0020: 5d 9d f8 05 60 06 a8 9e  2e 95 6e aa 6d a7        ]...`.....n.m.
2017-11-14 22:36:36.705447 > 0000: 04 3e 1a 02 01 04 00 1f  ff 1a 6a 3b 12 0e 0d 09
.>........j;....
  0010: 61 62 65 61 63 6f 6e 5f  46 46 31 46 cc           abeacon_FF1F.
2017-11-14 22:36:36.788447 > 0000: 04 3e 29 02 01 03 01 f1  6d 1d 44 53 c7 1d 02 01
.>).....m.DS....
  0010: 06 03 03 aa fe 15 16 aa  fe 10 fb 03 62 69 74 2e  ............bit.
  0020: 6c 79 2f 53 55 72 70 72  69 73 65 c8              ly/SUrprise.
```

Sniffing BLE advertisements & iBeacons

Spoofing attacks

- hcitool and companion scripts (https://github.com/irontec/ibeacons-simple-tools.git) can easily spoof iBeacons
- BT profile and BDADDR may have to be spoofed too
- Android Beacon Toy provides easy cloning feature!




```
renaud@primelt:~/installations/ibeacons-simple-tools$ sudo ./launch-ibeacon -u DEADBEEF-DEAD-BEEF-DEAD-DEADBEEFDEAD -M 1 -m 2
· Setting up hci0 [leadv, noscan] ✓
· Setting up beacon payload ✓
    UUID: DE AD BE EF DE AD BE EF DE AD DE AD BE EF DE AD
    MAYOR: 00 01 (1)
    MINOR: 00 02 (2)
    POWER: C5 (-59)
· Activating Beacon ✓
    hcitool -i hci0 cmd 0x08 0x0008 1E 02 01 1A 1A FF 4C 00 02 15 DE AD BE EF DE AD BE EF DE AD DE AD BE EF DE AD 00 01 00 02 C5 00
```

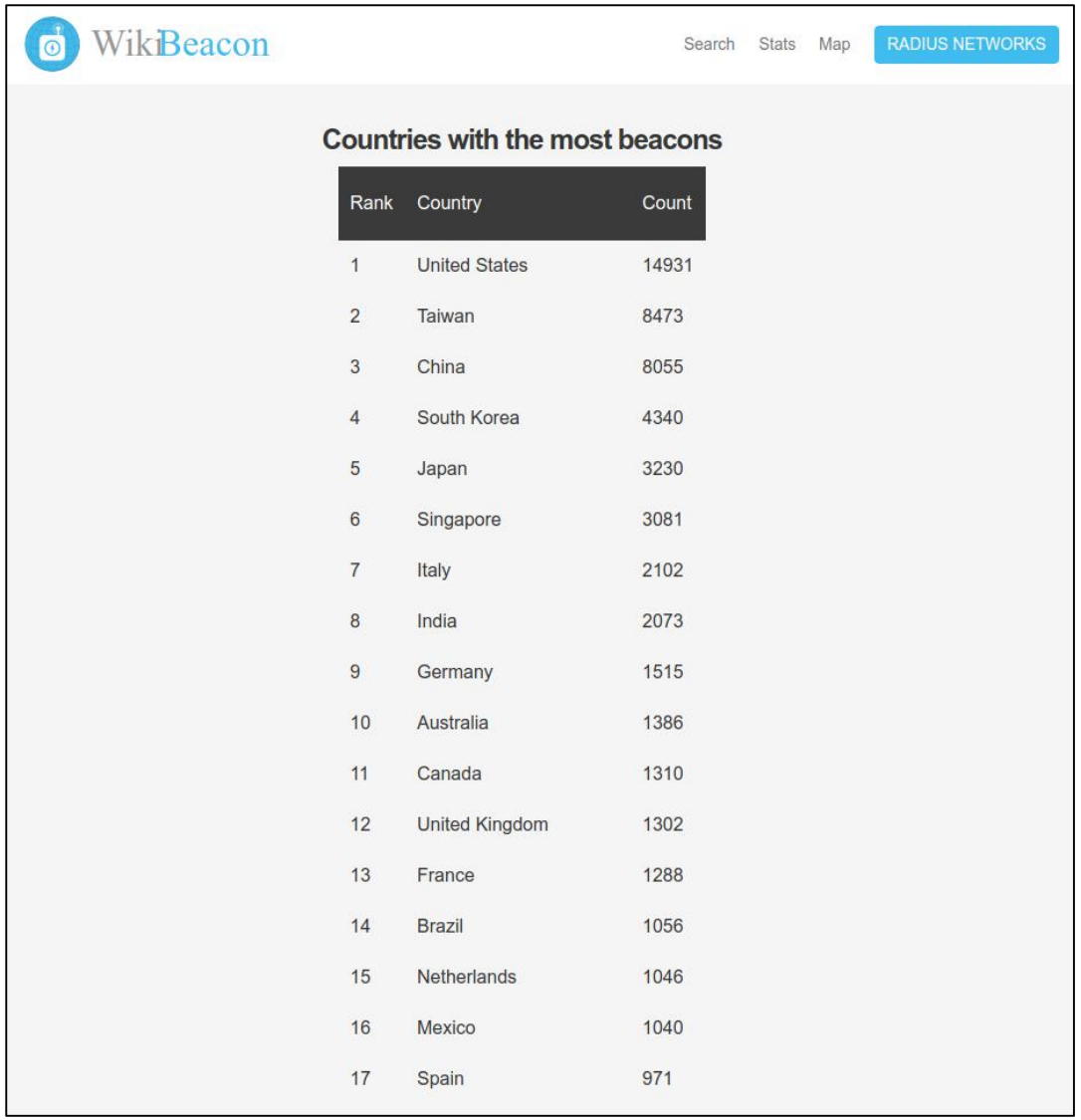
Forging fake iBeacon frames

WikiBeacon (1/5)

- Community resource providing crowd-sourced information (smartphone app) about proximity beacon usage
- Maps, stats and search tools
- http://www.wikibeacon.org/
- See also https://openuuid.net/

WikiBeacon (2/5)

WikiBeacon (3/5)



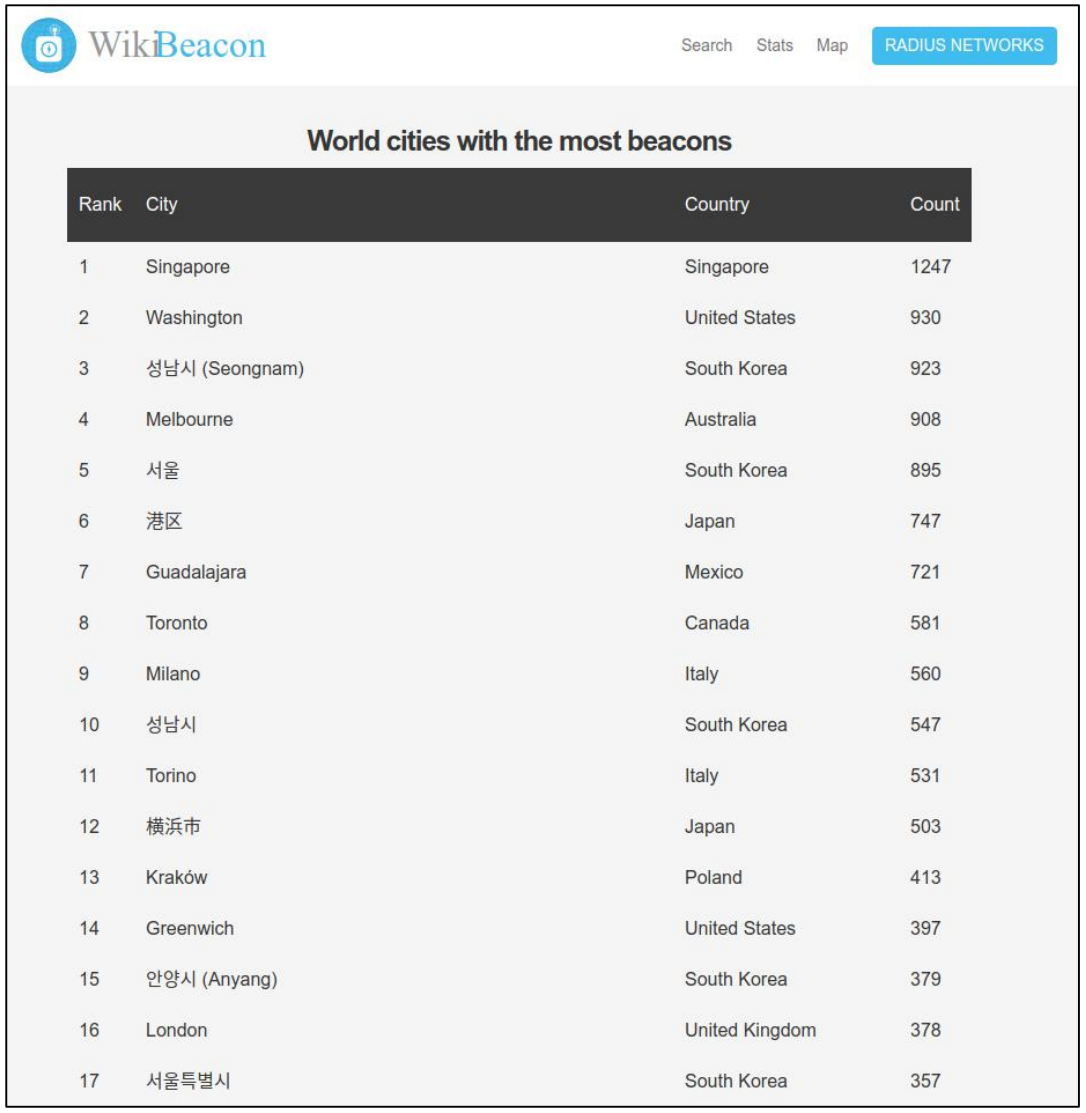
WikiBeacon

Search Stats Map RADIUS NETWORKS

Countries with the most beacons

Rank	Country	Count
1	United States	14931
2	Taiwan	8473
3	China	8055
4	South Korea	4340
5	Japan	3230
6	Singapore	3081
7	Italy	2102
8	India	2073
9	Germany	1515
10	Australia	1386
11	Canada	1310
12	United Kingdom	1302
13	France	1288
14	Brazil	1056
15	Netherlands	1046
16	Mexico	1040
17	Spain	971

WikiBeacon (4/5)

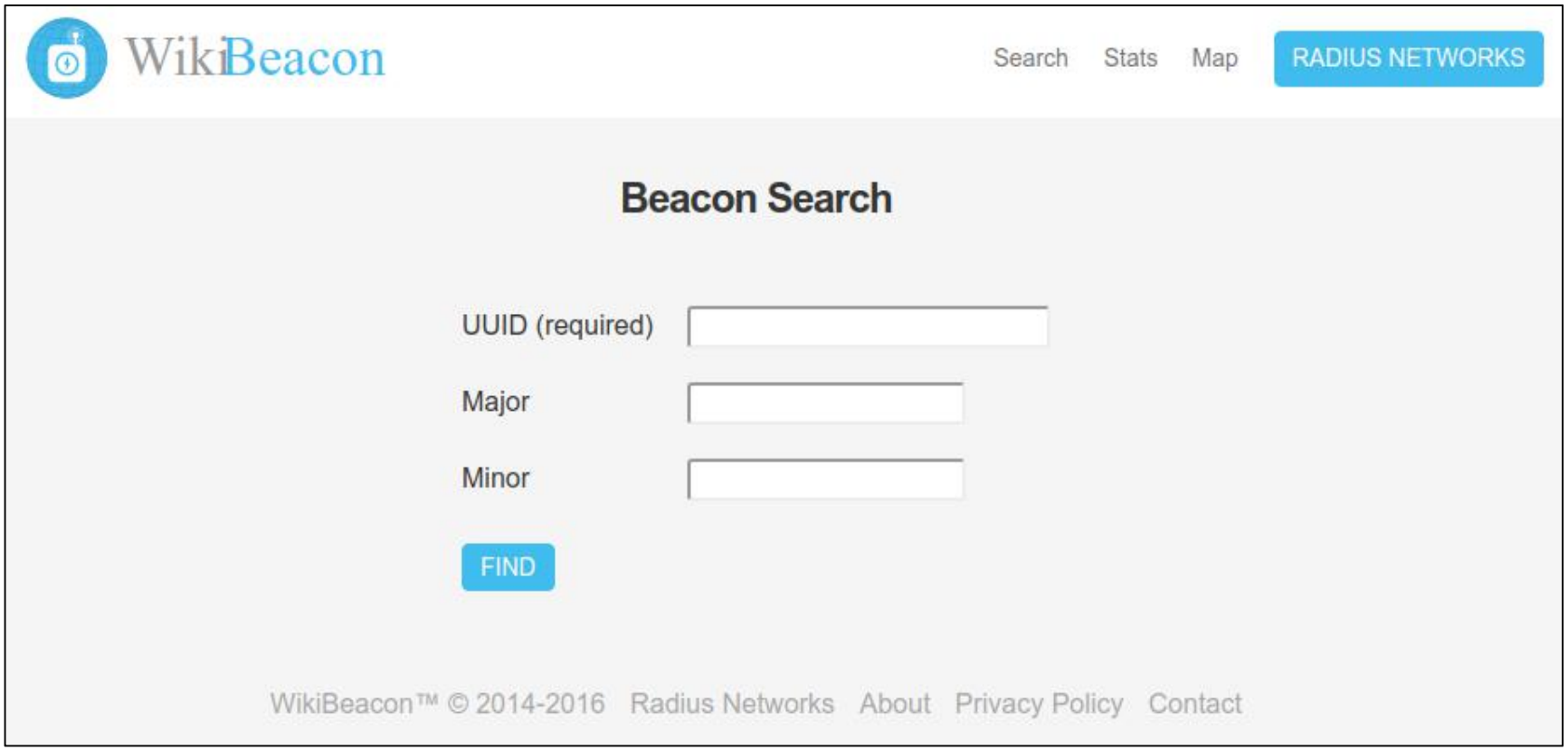


WikiBeacon Search Stats Map RADIUS NETWORKS

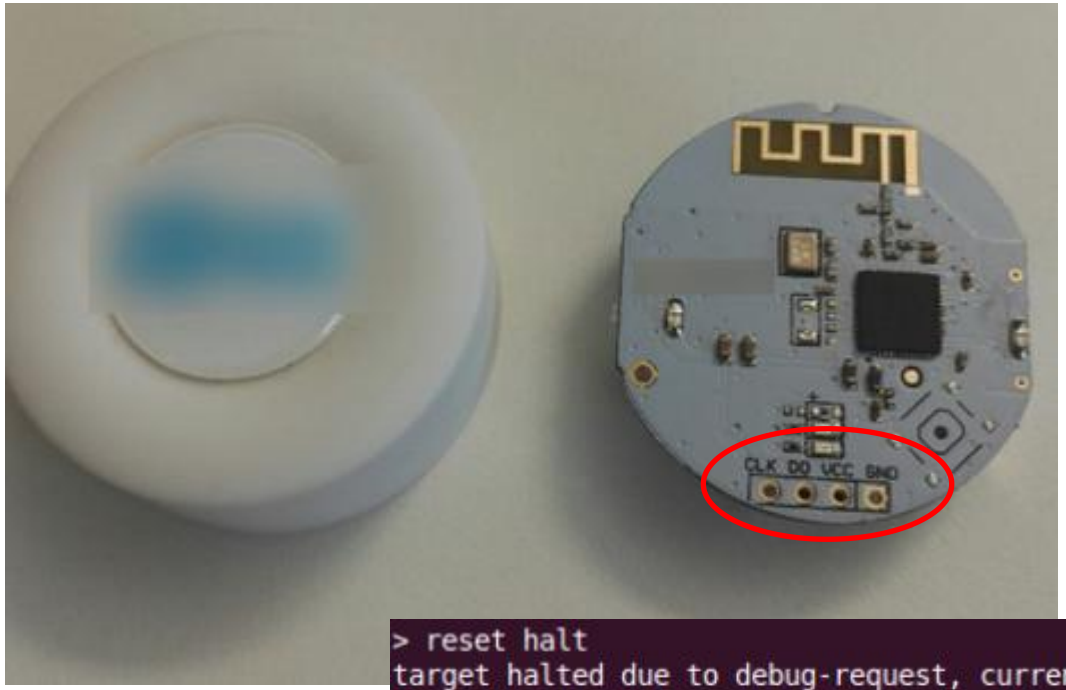
World cities with the most beacons

Rank	City	Country	Count
1	Singapore	Singapore	1247
2	Washington	United States	930
3	성남시 (Seongnam)	South Korea	923
4	Melbourne	Australia	908
5	서울	South Korea	895
6	港区	Japan	747
7	Guadalajara	Mexico	721
8	Toronto	Canada	581
9	Milano	Italy	560
10	성남시	South Korea	547
11	Torino	Italy	531
12	横浜市	Japan	503
13	Kraków	Poland	413
14	Greenwich	United States	397
15	안양시 (Anyang)	South Korea	379
16	London	United Kingdom	378
17	서울특별시	South Korea	357

WikiBeacon (5/5)



Attack scenarios: physical access



- Test points or flash memory access
- Dump with OpenOCD and a suitable adapter
- Access to all secrets & perfect cloning!

```
> reset halt
target halted due to debug-request, current mode: Thread
xPSR: 0xc1000000 pc: 0x000006d0 msp: 0x000007c0
> flash banks
#0 : nrf51.flash (nrf51) at 0x00000000, size 0x00000000, buswidth 1, chipwidth 1
#1 : nrf51.uicr (nrf51) at 0x10001000, size 0x00000000, buswidth 1, chipwidth 1
> dump_image nrf51.flash 0x00000000 0xffffff
```

```
0001f800  42 48 de 49                                 00 20 0c 9a
0001f810  66 00 02 00                                 33 09
0001f820  00 00 05 89                                 6f 6e
```

Attack scenarios

- Spoofing beacons can cause:
  - Location spoofing for applications
  - Fake data uploaded to cloud
  - Fraudulent profit (ex: game at CES 2015)
- iBeacon with weak configurations (DFU/FOTA) or passwords (PIN & passwords are usually sent... plaintext):
  - RCE
  - Advertisements for competitors
  - DoS
- UUID harvesting (app store or open database):
  - Application spamming
- Tracking / motion detection
- Vulnerabilities involving hooked mobiles applications: remote code execution?

Physical Web security

digital security

Payload formats

- URL scheme prefix and TLD are encoded for compression purposes:

Frame Specification

Byte offset	Field	Description
0	Frame Type	Value = `0x10`
1	TX Power	Calibrated Tx power at 0 m
2	URL Scheme	Encoded Scheme Prefix
3+	Encoded URL	Length 1-17

URL Scheme Prefix

Decimal	Hex	Expansion
0	0x00	`http://www.`
1	0x01	`https://www.`
2	0x02	`http://`
3	0x03	`https://`

Eddystone-URL HTTP URL encoding

The HTTP URL scheme is defined by RFC 1738, for example `https://goo.gl/S6zT6P`, and is used to designate Internet resources accessible using HTTP (HyperText Transfer Protocol).

The encoding consists of a sequence of characters. Character codes excluded from the URL encoding are used as text expansion codes. When a user agent receives the Eddystone-URL the byte codes in the URL identifier are replaced by the expansion text according to the table below.

Decimal	Hex	Expansion
0	0x00	.com/
1	0x01	.org/
2	0x02	.edu/
3	0x03	.net/
4	0x04	.info/
5	0x05	.biz/
6	0x06	.gov/
7	0x07	.com
8	0x08	.org
9	0x09	.edu
10	0x0a	.net
11	0x0b	.info
12	0x0c	.biz
13	0x0d	.gov
14..32	0x0e..0x20	Reserved for Future Use
127..255	0x7F..0xFF	Reserved for Future Use

- Full specification: https://github.com/google/eddystone/tree/master/eddystone-url

Physical web

- Uses Eddystone URL protocol
- Straightforward to implement:
  Beacon Toy (Android), PyBeacon (Python)

```
$ sudo pip install PyBeacon
$ sudo PyBeacon -u https://twitter.com/nono2357
Advertising: url : https://twitter.com/nono2357
```

- But some limitations

Eddystone URL limitations and bypasses (1/2)

- Basic limitations:
  - Chrome and Nearby Notifications only support HTTPS URLs
  - URL length limited to 17 characters

☞ URL shorteners!

"Physical" phishing & tracking with URL shorteners

Eddystone URL limitations and bypasses (2/2)

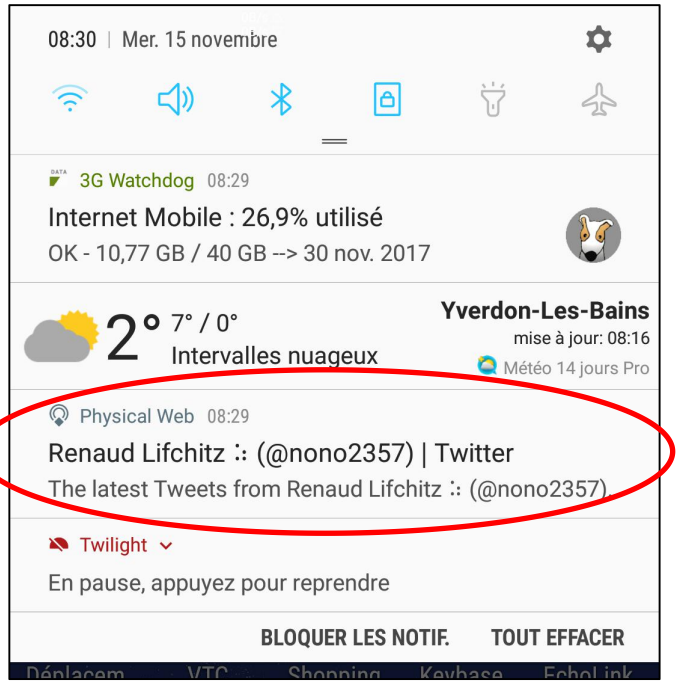
- Google Physical web service uses a proxy to preview links while protecting personal information and possibly filter spam

☞ Testing Google proxy could be fun!

- User agent cloaking
- Recursive redirections
- Allowed content types

What about other web services?

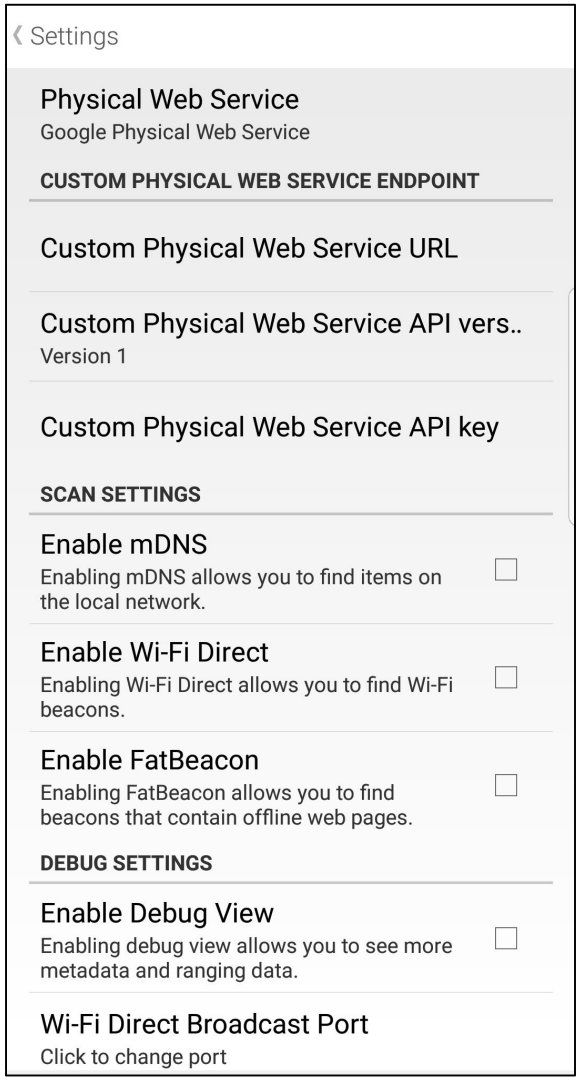
- Once link is clicked, the user is no more protected against fingerprinting (IP, MAC, user agent, OS, browser...), tracking and exploits



mDNS, Wi-Fi Direct, SSDP and FatBeacon support

- mDNS & SSDP: discovery of physical web services throught Wi-Fi and IP
- Wi-Fi Direct: serves content via P2P Wi-Fi and HTTP (device name: PW-<title>-<port>)
- FatBeacon: sends full content over BLE

- These features need to be carefully tested for security before use



Eddystone security

- Eddystone can provide beacon security (requires internet connection)
- Beacons should also rotate their BDADDR for privacy
- Eddystone cryptographic features (based on AES-EAX), extended features (mDNS, Wi-Fi Direct, SSDP, FatBeacon) and implementations should be thoroughly audited...

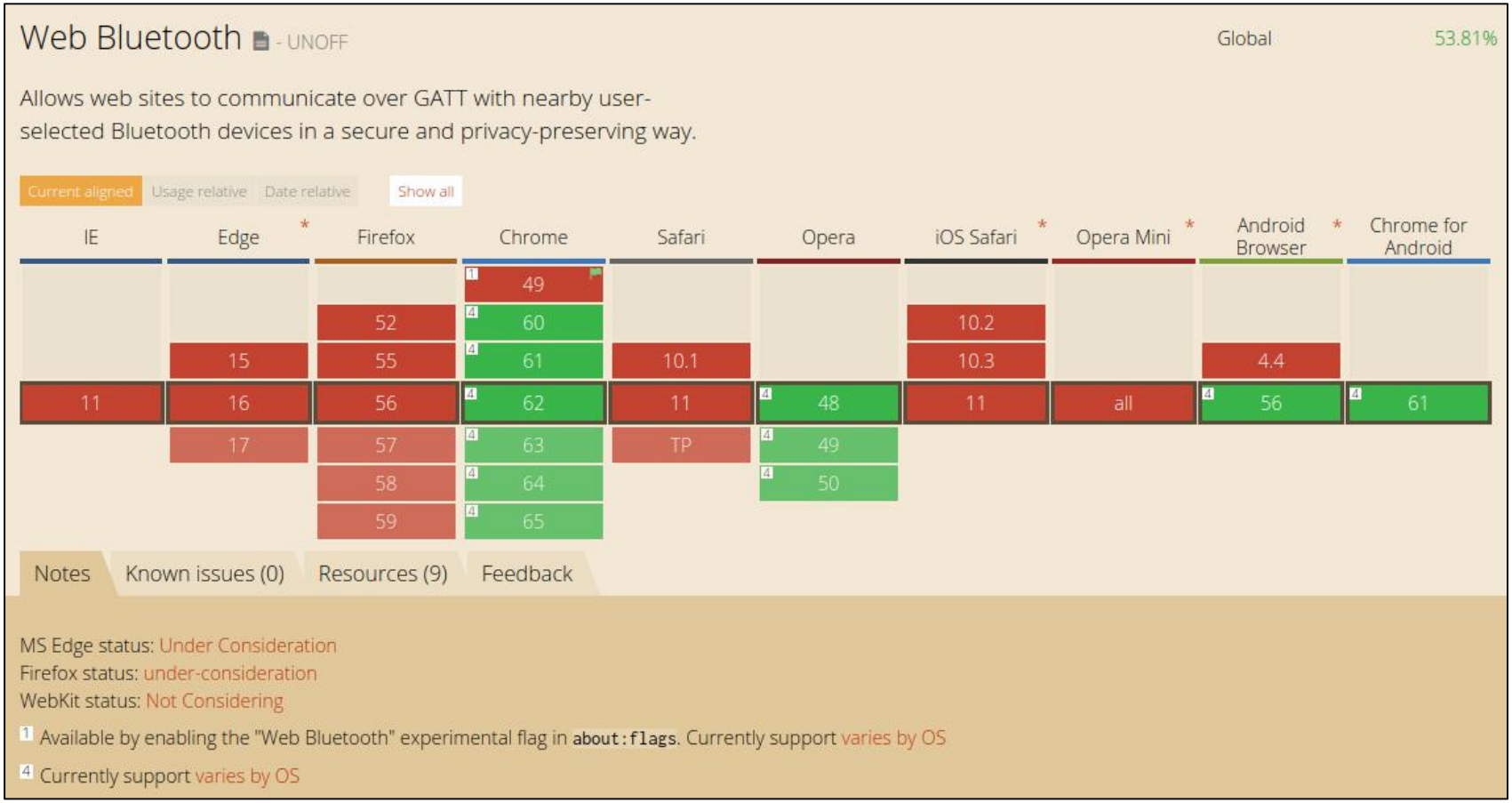
Web Bluetooth security

digital security

Introduction to the specification

- W3C open specification: https://webbluetoothcg.github.io/web-bluetooth/
- Allows a desktop/mobile browser to directly query BLE devices
- Provides a Javascript API: https://developer.mozilla.org/fr/docs/Web/API/Web_Bluetooth_API

Compatibility



Security

- A web page can scan devices and read or write GATT characteristics
- Web Bluetooth extends IoT RF short range attacks to very long range: typically a web page can query your smartwatch for your phone book or your heart rate!
- Harmless web sites can be attacked with XSS to relay BLE attacks...
- Web Bluetooth allows combinations of logical and physical attacks, even remotely!
- Security nightmare

Querying a BLE device using a web page:

chrome --enable-web-bluetooth

chrome://flags/ -> "Experimental Web Platform"

https://googlechrome.github.io/samples/web-bluetooth/device-info.html?allDevices=true

digital security

IT & IoT Security

Thanks!

Questions?

Contact:
renaud.lifchitz@digitalsecurity.fr
info@digitalsecurity.fr

Follow us on Twitter!: @iotcert