Open Wounds: The last 5 years have left Bluetooth to bleed

A survey like no other!

Xeno Kovah
OpenSecurityTraining2 &
Dark Mentor LLC

Modified from https://www.maskworld.com/english/news/diy-blood-effects ;)

About Me

- 75% of my time is spent making free (as in beer), open access, and *open source* (CreativeCommons licensed) classes for a non-profit I started, **OpenSecurityTraining2 (ost2.fyi)**

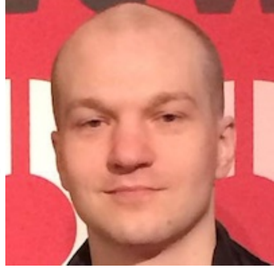
About Me

- 75% of my time is spent making free (as in beer), open access, and *open source* (CreativeCommons licensed) classes for a non-profit I started, **OpenSecurityTraining2 (ost2.fyi)**

- 25% of my time doing consulting and research for **Dark Mentor LLC**
  - The research is for fun, but is *also a trojan horse* to get me into conferences to tell you about OST2 ;)

OST2 Crew

In order of appearance



Xeno Kovah

Arch1001 ~28 hours, Arch2001 ~27 hours
Arch4001 ~14 hours, HW1101 ~6 hours
Vulns1001 ~15 hours, Vulns1002 ~23 hours

Piotr Król

Arch4021 ~6 hours
Arch4031 ~6 hours

Michał Żygowski

Arch4021 ~6 hours

Thaís Moreira Hamasaki

RE3201 ~6 hours

Gal Zaban

RE3011 ~6 hours

Kc Udonsi

Vulns1001 ~15 hours

Cedric Halbronn

Dbg3011 ~6 hours
Arch2821 ~5 hours
Exp4011 ~33(!) hours

Sina Karvandi

Dbg3301 ~16 hours

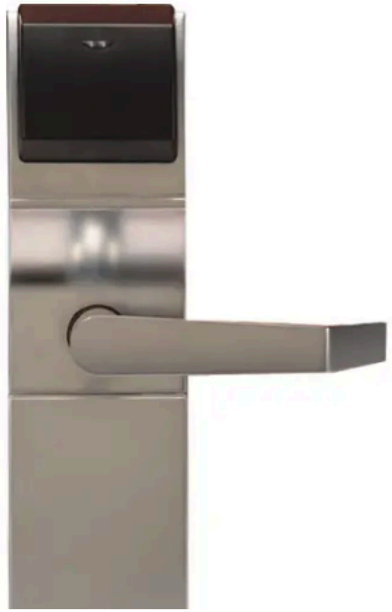
What I Want To Know:

What Bluetooth Chip Is Inside Any Device


DARK MENTOR


DARK MENTOR

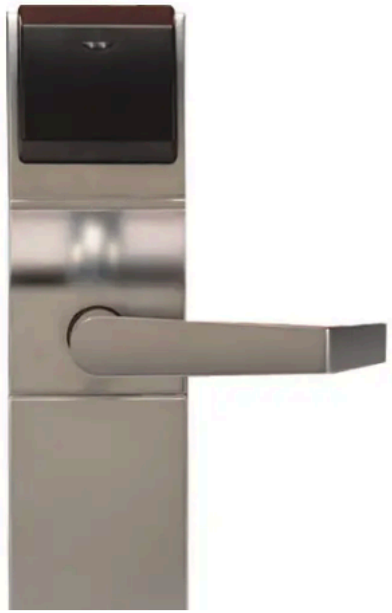

nest



TEXAS INSTRUMENTS

BROADCOM®

SILICON LABS



?

Why I Want To Know It:

So I Know If It's Vulnerable To A Firmware-Level Exploit

TEXAS INSTRUMENTS

SILICON LABS

BROADCOM®

ARMIS®

DST2.FYI

Texas Instruments

SILICON LABS®

BROADCOM®

ARMIS®

DST2.FYI

TEXAS INSTRUMENTS

SEMO
SECURE MOBILE NETWORKING

SILICON LABS®

BROADCOM®

ARMIS®

DARK MENTOR

TEXAS INSTRUMENTS

SEEMO

SECURE MOBILE NETWORKING

SILICON LABS®

BROADCOM®

ARMIS®

DARK MENTOR

TEXAS INSTRUMENTS

SEEMO

SECURE MOBILE NETWORKING

BROADCOM®

DARK MENTOR

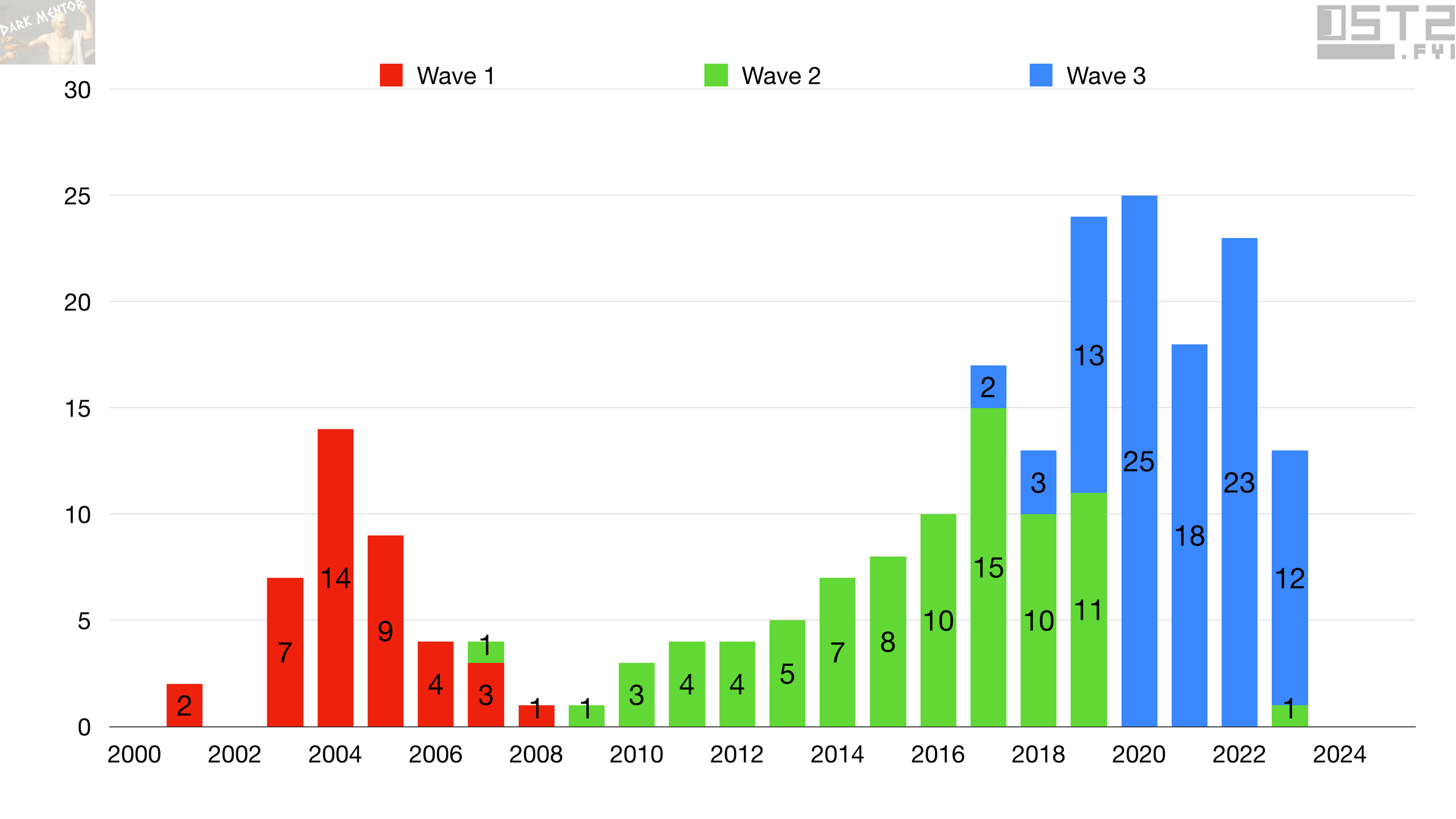
SILICON LABS®

How Am I Going To Figure It Out?

1) Know That I Know Nothing 😏
2) Do A Bunch Of Naive BT Data Collection
3) Find Out What I Don't Know

How Am I Going To Figure It Out?

1) Know That I Know Nothing 😏
2) Do A Bunch Of Naive BT Data Collection
3) Find Out What I Don't Know
4) Read Related Work!


Wave 1
Wave 2
Wave 3
30
25
20
15
10
5
0
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
2020
2022
2024
2
7
14
9
4
3
1
1
1
3
4
4
5
7
8
10
15
2
10
3
11
13
25
18
23
1
12

Dark Mentor LLC

Corrupting the young and old alike.

About Us

Dark Mentor LLC was started in 2018 as a low-level-security consultancy, specializing in full-stack bluetooth security, firmware security, and security education.

DarkMentor.com/bt.html

Dark Mentor LLC Home Publications Bluetooth Security Timeline PC/Server Firmware Security Timeline

Dark Mentor LLC

Corrupting the young and old alike.

Us

LLC was started in 2018 as a low-level-security consultancy, specializing in
etooth security, firmware security, and security education.

DarkMentor.com/bt.html

Bluetooth Security Timeline

6th September 2023 at 2:32pm

Submit additions or corrections via Merge Requests at https://gitlab.com/XenoKovah/bluetooth_security_timeline

Read this *Contributor Guidance* before sending Merge Requests

Key:
⬆️🥊 = High Impact!
🥷 = PoC exploit available
🛠️ = Tool

2023

08

Snoop on to them as they snoop on to us 🛠️

Turning my phone into a skimming device: MPos Solutions 🛠️

Intel BIOS Advisory – Memory Corruption in HID Drivers

It Was Harder to Sniff Bluetooth Through My Mask During the Pandemic...

07

Blind My - An Improved Cryptographic Protocol to Prevent Stalking in Apple's Find My Network

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Android

20 matches

Title matches:

A Story About Three Bluetooth Vulnerabilities in Android
A Tale of Reversing the Android-based Snow2 HUD
Android Bluetooth Vulnerabilities in the March 2018 Security Bulletin
BadBluetooth: Breaking Android Security Mechanisms via Malicious Bluetooth Peripherals
Deep into Android Bluetooth Bug Hunting: New Attack Surfaces and Weak Code Patterns
Env: Android
Inside Job: Understanding and Mitigating the Threat of External Device Mis-Bonding on A
Nearby Threats: Reversing, Analyzing, and Attacking Google's 'Nearby Connections' on A
No need to ask the Android: Bluetooth-Low-Energy scanning without the location permiss
Stealthily Access Your Android Phones: Bypass the Bluetooth Authentication
Tricking Android Smart Lock with Bluetooth

All matches:

A Story About Three Bluetooth Vulnerabilities in Android
A Tale of Reversing the Android-based Snow2 HUD
Android Bluetooth Vulnerabilities in the March 2018 Security Bulletin
BadBluetooth: Breaking Android Security Mechanisms via Malicious Bluetooth Peripherals
BlueBorne
Bluetooth Security Timeline
BrokenMesh: New Attack Surfaces of Bluetooth Mesh
Deep into Android Bluetooth Bug Hunting: New Attack Surfaces and Weak Code Patterns
Env: Android
Exploit Millions of Pebble Smartwatches for Fun and Profit
Extracting the painful (blue)tooth
Fingerprinting Bluetooth-Low-Energy Devices Based on the Generic Attribute Profile
Happy MitM – Fun and Toys in Every Bluetooth Device
Inside Job: Understanding and Mitigating the Threat of External Device Mis-Bonding on A

Bluetooth Security Timeline

6th September 2023 at 2:32pm

Submit additions or corrections via Merge Requests at https://gitlab.com/XenoKovah/bluetooth_security_timeline

Read this *Contributor Guidance* before sending Merge Requests

Key:
⬆️🧲 = High Impact!
🥷 = PoC exploit available
🛠️ = Tool

2023

08

Snoop on to them as they snoop on to us 🛠️

Turning my phone into a skimming device: MPos Solutions 🛠️

Intel BIOS Advisory – Memory Corruption in HID Drivers

It Was Harder to Sniff Bluetooth Through My Mask During the Pandemic...

07

Blind My - An Improved Cryptographic Protocol to Prevent Stalking in Apple's Find My Network

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Damien 7 matches

Title matches:

All matches:

BtleJuice: the Bluetooth Smart Man In The Middle Framework
Defeating Bluetooth Low Energy 5 PRNG for Fun and Jamming
ESPwn32: Hacking with ESP32 System-on-Chips
Mass-pwning with a small IoT spy bug
Sniffing BTLE with the Micro:Bit
Weaponizing the BBC Micro Bit
You'd better secure your BLE devices or we'll kick your butts!

BtleJuice: the Bluetooth Smart Man In The Middle Framework

8th May 2023 at 4:56pm

AttackSurface: MitM | Author: Damien Cauquil | Conference: DEF CON | Conference: Hack.lu | Month: 08 | Month: August | Org: CERT UBIK | Type: Attack | Type: Tool | Year: 2016

2016-08

This was originally presented at the DEF CON IoT Village, not the main conference.

Slides:
https://files.speakerdeck.com/presentations/668f2e5bee01416c94fb0a6ee107c3a6/BtleJuice__the_Bluetooth_Smart_Man_In_The_Middle_Framework.pdf

Mirror:
https://web.archive.org/web/20220427223245/https://files.speakerdeck.com/presentations/668f2e5bee01416c94fb0a6ee107c3a6/BtleJuice__the_Bluetooth_Smart_Man_In_The_Middle_Framework.pdf

Mirror: https://gitlab.com/XenoKovah/bluetooth-security-timeline/-/blob/main/paper-mirror/Cauquil_slides_BtleJuice_the_Bluetooth_Smart_Man_In_The_Middle_Framework.pdf

Code: https://github.com/DigitalSecurity/BtleJuice

Code: https://github.com/DigitalSecurity/btlejuice-node-bindings

Code: https://github.com/DigitalSecurity/btlejuice-python-bindings

Twitter (Damien Cauquil): @virtualabs, Mastodon: @virtualabs@mamot.fr

---

This was subsequently presented at Hack.lu 2016-10-20

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Damien 7 matches

Title matches:

All matches:

- BtleJuice: the Bluetooth Smart Man In The Middle Framework
- Defeating Bluetooth Low Energy 5 PRNG for Fun and Jamming
- ESPwn32: Hacking with ESP32 System-on-Chips
- Mass-pwning with a small IoT spy bug
- Sniffing BTLE with the Micro:Bit
- Weaponizing the BBC Micro Bit
- You'd better secure your BLE devices or we'll kick your butts!

BtleJuice: the Bluetooth Smart Man In The Middle Framework

8th May 2023 at 4:56pm

AttackSurface: MitM | Author: Damien Cauquil | Conference: DEF CON

AttackSurface: MitM

A Billion Open Interfaces for Eve and Mallory: MitM, DoS, and Tracking Attacks on iOS and macOS Through Apple Wireless Direct Link
Access Your Tesla without Your Awareness: Compromising Keyless Entry System of Model 3
BLE injection-free attack: a novel attack on bluetooth low energy devices
BLESA: Spoofing Attacks against Reconnections in Bluetooth Low Energy
Blue Picking: Hacking Bluetooth Smart Locks
BtleJuice: the Bluetooth Smart Man In The Middle Framework
Disrupting Continuity of Apple's Wireless Ecosystem Security: New Tracking, DoS, and MitM Attacks on iOS and macOS Through Bluetooth Low Energy, AWDL, and Wi-Fi
Extrapolating Formal Analysis to Uncover Attacks in Bluetooth Passkey Entry Pairing
GATTacking Bluetooth Smart Devices
Hacking Bluetooth Low Energy Based Applications
Happy MitM – Fun and Toys in Every Bluetooth Device
Man-in-the-middle attacks on Secure Simple Pairing in Bluetooth standard V5.0 and its countermeasure
Men-in-the-Middle Attack Simulation on Low Energy Wireless Devices using Software Define Radio
Security Weaknesses in Bluetooth
Turning my phone into a skimming device: MPos Solutions
You'd better secure your BLE devices or we'll kick your butts!

k.pdf

Code: https://github.com/DigitalSecurity/BtleJuice

Code: https://github.com/DigitalSecurity/btlejuice-node-bindings

Code: https://github.com/DigitalSecurity/btlejuice-python-bindings

Twitter (Damien Cauquil): @virtualabs, Mastodon: @virtualabs@mamot.fr

---

This was subsequently presented at Hack.lu 2016-10-20

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Damien 7 matches

$:/TagManager

Colour	Tag	Count	Icon	Info
	AttackSurface: ADV_EXT_IND	1		
	AttackSurface: ADV_IND	2		
	AttackSurface: BIAS	1		
	AttackSurface: BLE LL	4		
	AttackSurface: BNEP	2		
	AttackSurface: Brute Force	7		
	AttackSurface: BT Mesh	1		
	AttackSurface: Co-located Apps on Paired Device	2		
	AttackSurface: CONNECT_IND	1		
	AttackSurface: Cryptography	2		
	AttackSurface: Design	1		
	AttackSurface: Downgrade Attacks	1		
	AttackSurface: Extended Inquiry Response	1		
	AttackSurface: Fault Injection	3		
	AttackSurface: GATT	4		
	AttackSurface: HCI	2		
	AttackSurface: HID over BT	2		
	AttackSurface: Impersonation	2		
	AttackSurface: JTAG	1		
	AttackSurface: Key Agreement Protocols	11		

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Damien 7 matches

Open Recent Tools More

- BtleJuice: the Bluetooth Smart Man In The Middle Framework
- Bluetooth Security Timeline
- **$:/TagManager**

close all

Conference: RuxCon	1	
Conference: ShmooCon	4	
Conference: SSTIC	2	
Conference: Summercon	1	
Conference: TCHES	1	
Conference: THOTCON	1	
Conference: ToorCon	1	
Conference: USENIX ATC	1	
Conference: USENIX NSDI	1	
Conference: USENIX Security	10	
Conference: VirusBulletin	1	
Conference: WiCon	1	
Conference: WiSec	16	
Conference: WOOT	8	
Env: Android	14	
Env: Apple	8	
Env: BlueZ	5	
Env: Broadcom Firmware	3	
Env: Cambridge Silicon Radio (CSR) Firmware	1	
Env: Cypress Firmware	2	
Env: Fitbit Firmware	2	
Env: Fuchsia	1	
Env: Industry-wide	4	
Env: iOS userspace	2	
Env: Linux kernelspace	1	

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Open Recent Tools More

- BtleJuice: the Bluetooth Smart Man In The Middle Framework
- Bluetooth Security Timeline
- **$:/TagManager**

close all

Org: University of Oxford	4	
Org: University of Padua	2	
Org: University of Patras	1	
Org: University of Science and Technology of China	1	
Org: University of Toulous	4	
Org: University of Twente	1	
Org: University of Wollongong	1	
Org: Virginia Tech	2	
Org: Ziften Technologies	1	
Protocol: BLE	50	
Protocol: BT Classic	51	
Protocol: BT Mesh	2	
Tech: Fingerprinting	11	
Tech: Fuzzing	11	
Tech: Jamming	2	
Tech: SDR	2	
Tech: Sniffing	29	
Tech: VariantAnalysis	1	
TestTarget: Acer Liquid Leap	1	
TestTarget: Apple 339S00056	1	
TestTarget: Apple 339S00199	1	
TestTarget: Apple 339S00397	1	

Protocol: BT Mesh

BlueMirror: Reflections on Bluetooth Pairing and Provisioning Protocols
BrokenMesh: New Attack Surfaces of Bluetooth Mesh

Bluetooth Security Timeline

By @XenoKovah of @DarkMentorLLC

Open | Recent | Tools | More

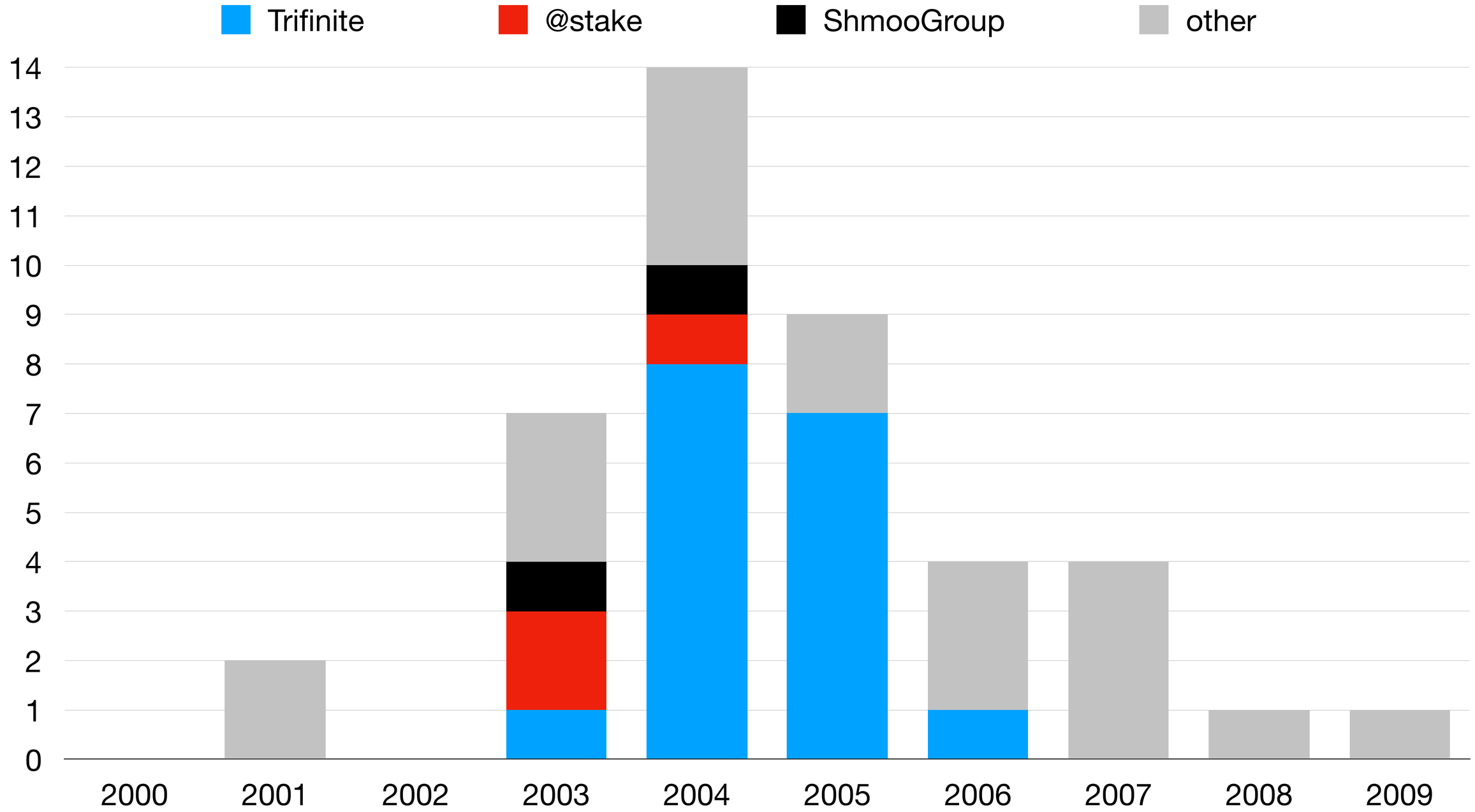
- BtleJuice: the Bluetooth Smart Man In The Middle Framework
- Bluetooth Security Timeline
- **$:/TagManager**

close all

Open Access || GTFO

- There are papers explicitly left off the timeline because they are not openly available
  - I'm happy to *not* read closed publications, and not have anyone else read them either

Wave 1


Trifinite
@stake
ShmooGroup
other
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009

Trifinite

Martin Herfurt, Adam Laurie, Collin Mulliner, Marcel Holtmann

- Used named-logo bugs before it was cool! Eat your heart out Heartbleed!
- First Wave 1 publication: 2003-11-??, last Wave 1 publication: 2006-01-??

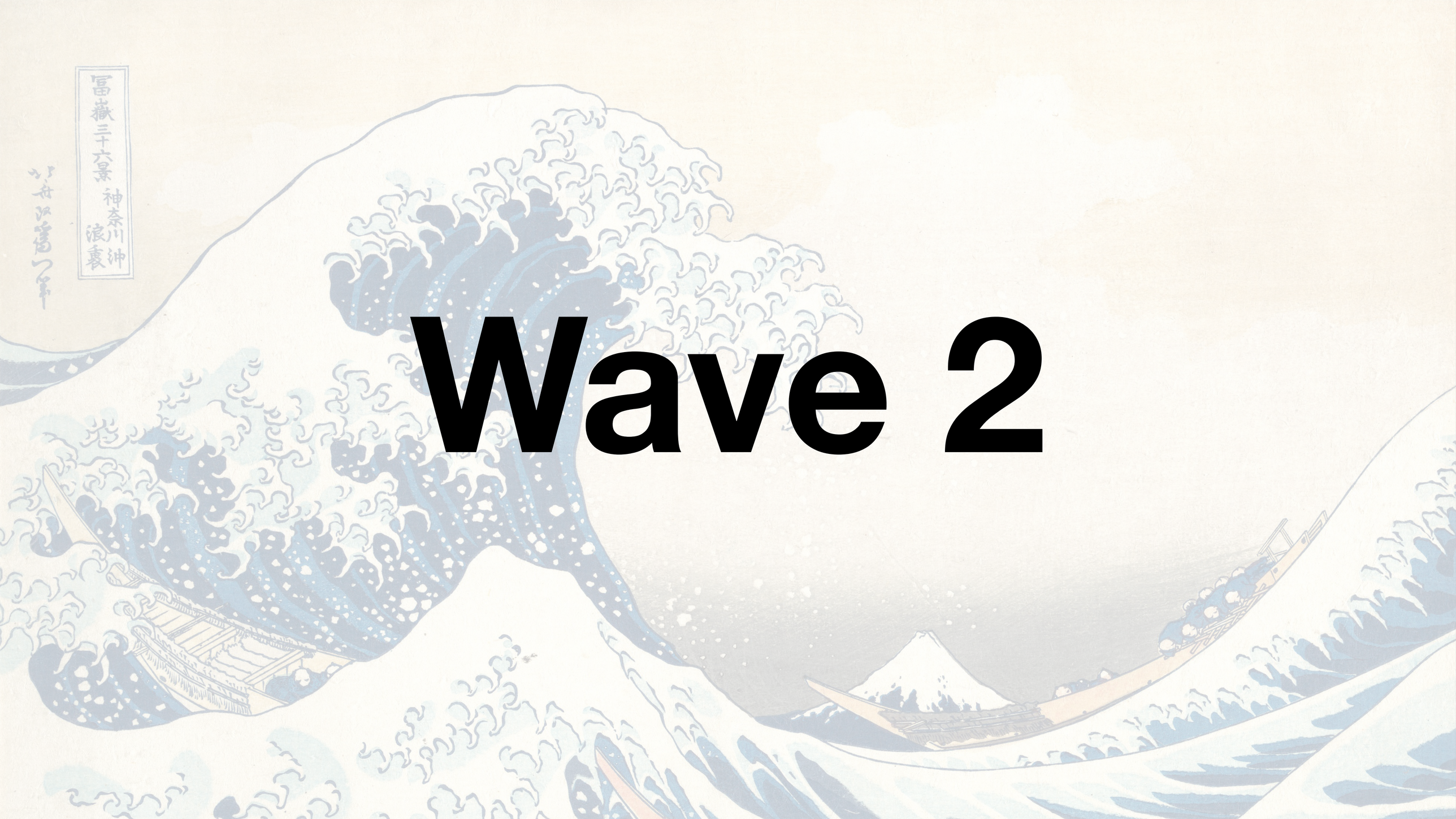
Trifinite

Martin Herfurt, Adam Laurie, Collin Mulliner, Marcel Holtmann

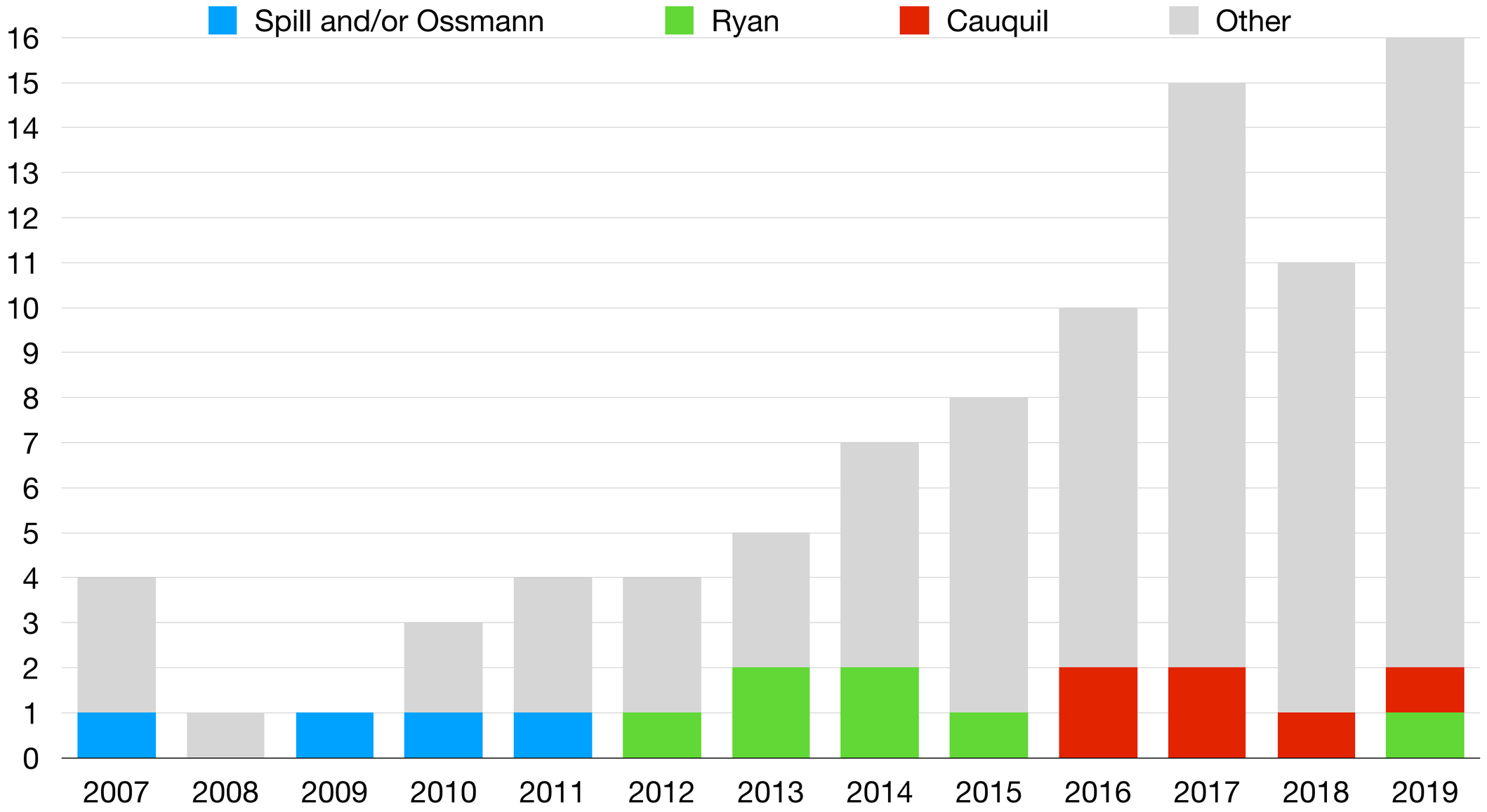
- Used named-logo bugs before it was cool! Eat your heart out Heartbleed!
- First Wave 1 publication: 2003-11-??, last Wave 1 publication: 2006-01-??

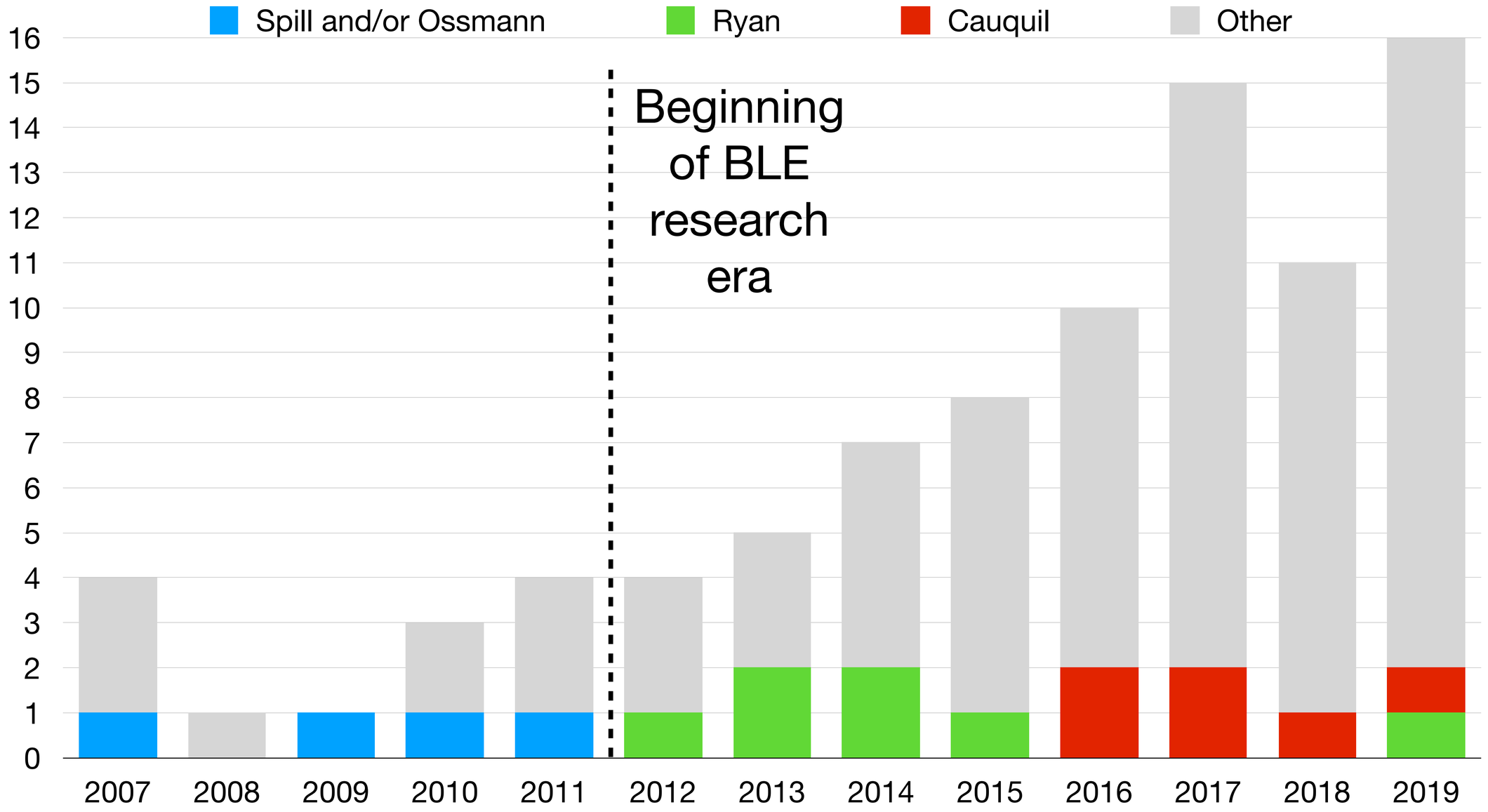
The Tools of Wave 1

- The Trifinite PoCs
- Redfang by @stake



Wave 2


Spill and/or Ossmann
Ryan
Cauquil
Other
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019


Spill and/or Ossmann
Ryan
Cauquil
Other
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
Beginning of BLE research era
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019

Wave 2

Tools

Tools of Wave 2: Ubertooth

BT Classic: Dominic Spill & Mike Ossmann, BLE: Mike Ryan

- 2011: Uses wireless signal aliasing trick to behave like it's listening on many BT channels at once, and decode the packets it sees
- 2013: BLE support added by Mike Ryan
- Support ended in Dec 2023*

* https://greatscottgadgets.com/2022/12-22-ubertooth-retirement/

Tools of Wave 2: CrackLE

Mike Ryan

- 2013: Bluetooth LE < spec v4.0 & v4.1 only had support for 6-digit-pin-code based pairing*
- This tool can brute force a temporary key, and then decrypt traffic
- BT spec v4.2 improved pairing by adding Elliptic Curve Diffie Hellman

* And also Out-Of-Band (OOB) pairing which is for vendors to make up themselves

Tools of Wave 2: GATTack

Sławomir Jasek

- 2016: BLE MitM, with an emphasis on manipulation of GATT data

Tools of Wave 2: BTLEjuice & BTLEjack

Damien Caquil

- 2016: BTLEjuice = BLE MitM
- 2018: BTLEjack = BLE MitM w/ connection hijacking & hop-along jamming
- Both use a custom firmware for the BBC Microbit (or equivalent Nordic chips)

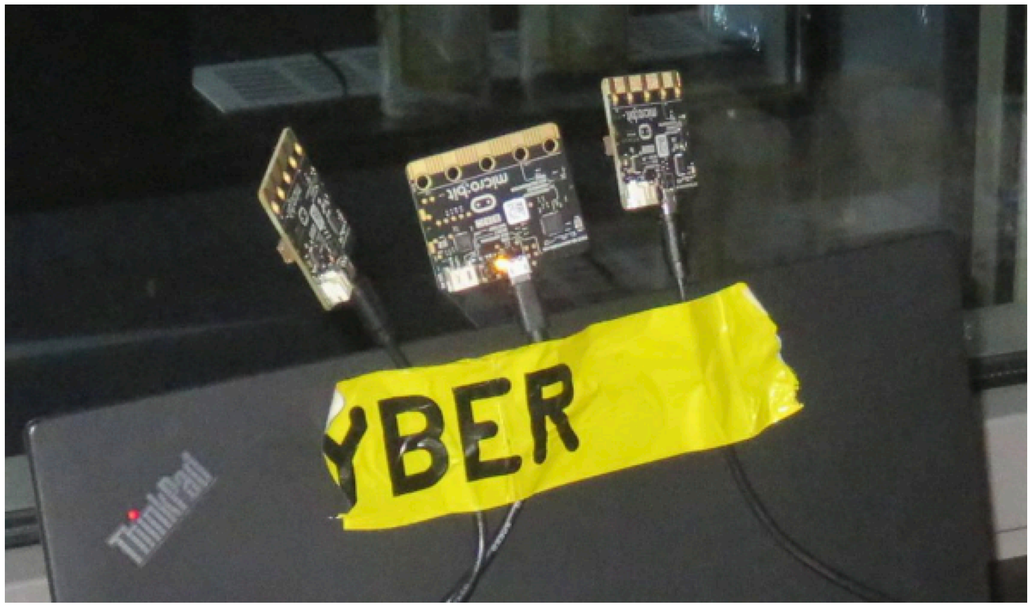
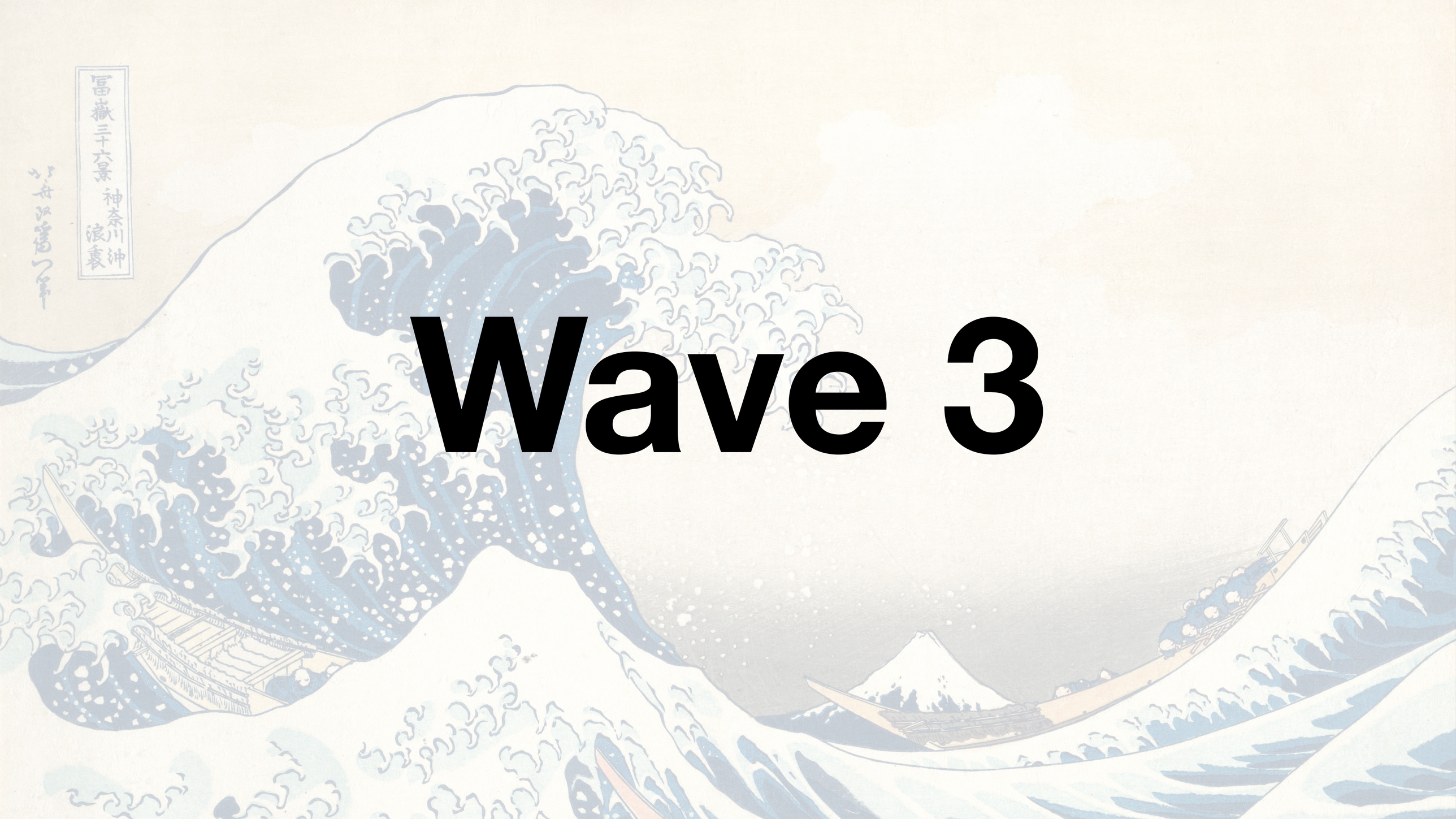
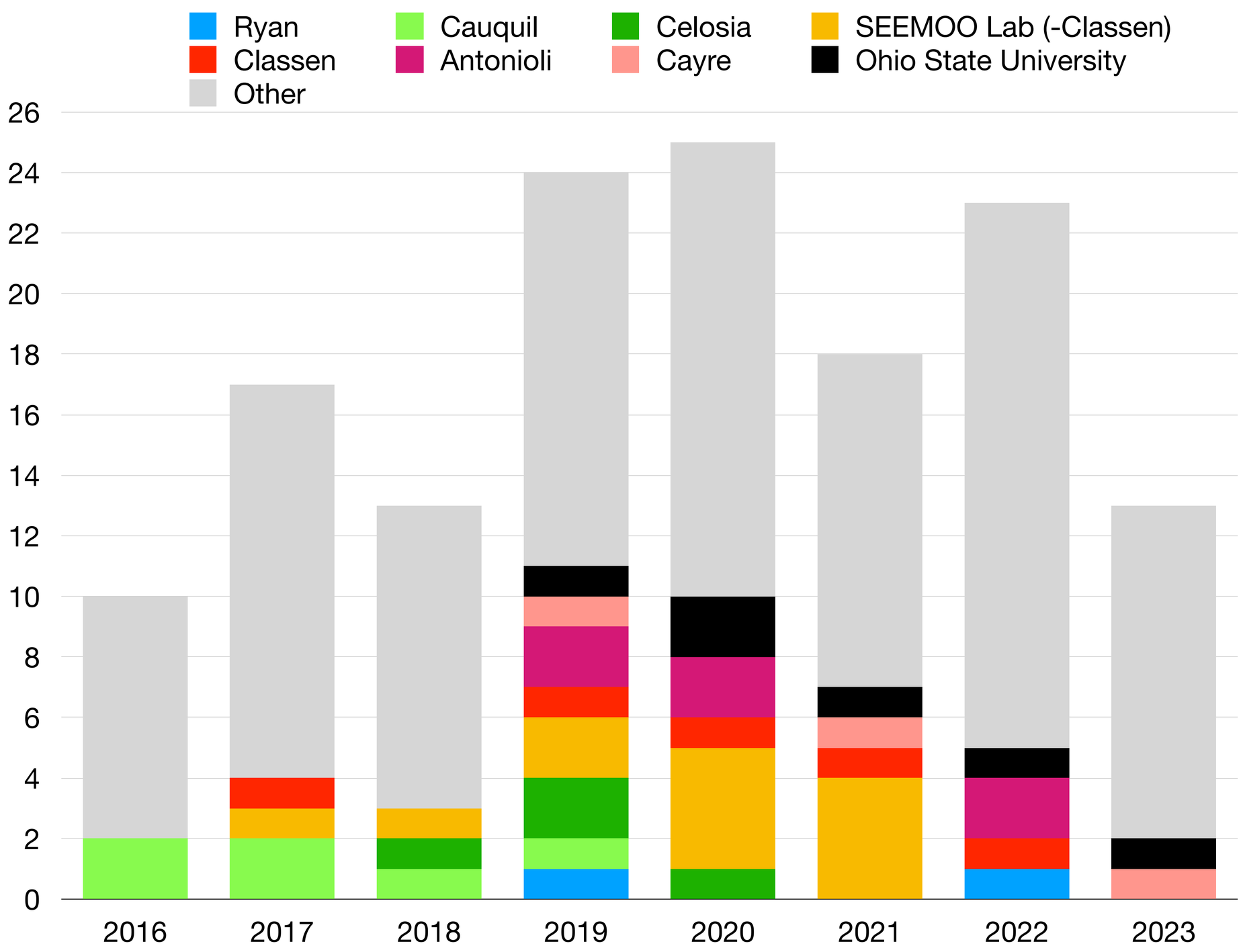


Image from https://i.blackhat.com/USA-19/Thursday/us-19-Ray-Moving-From-Hacking-IoT-Gadgets-To-Breaking-Into-One-Of-Europes-Highest-Hotel-Suites.pdf



Wave 3


Ryan
Classen
Other
Cauquil
Antonioli
Celosia
Cayre
SEEMOO Lab (-Classen)
Ohio State University
26
24
22
20
18
16
14
12
10
8
6
4
2
0
2016
2017
2018
2019
2020
2021
2022
2023

https://www.bluetooth.com/learn-about-bluetooth/key-attributes/bluetooth-security/reporting-security/

Bluetooth Security Notices

Vulnerability	Publication Date	Details	Specifications Affected	CVE [NVD]
Pairing Mode Confusion in BLE Passkey Entry	12/09/2022	SIG Security Notice	Core Spec v4.0 to 5.3	CVE-2022-25836
Pairing Mode Confusion in BR/EDR	12/09/2022	SIG Security Notice	Core Spec v1.0B to 5.3	CVE-2022-25837
InjectaBLE: Injecting malicious traffic into established Bluetooth Low Energy connections	06/21/2021	SIG Security Notice	Core Spec, v4.0 to 5.2	CVE-2021-31615
Bluetooth Mesh Profile AuthValue leak	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26559
Malleable commitment in Bluetooth Mesh Profile provisioning	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26556
Predictable Authvalue in Bluetooth Mesh Profile provisioning leads to MITM	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26557

legacy-pairing protocol				
Impersonation in the Passkey entry protocol	05/24/2021	SIG Security Notice	Core Spec, v2.1 to 5.2	CVE-2020-26558
Exploiting Cross-Transport Key Derivation	09/09/2020	SIG Security Notice	Core Spec, v4.2 to 5.0	CVE-2020-15802
Pairing Method Confusion	05/18/2020	SIG Security Notice	Core Spec, v2.1 to v5.2	CVE-2020-10134
Bluetooth Impersonation Attacks	05/18/2020	SIG Security Notice	Core Spec, v2.1 to v5.2	CVE-2020-10135
Key Negotiation of Bluetooth	08/13/2019	SIG Security Notice	Core Spec, v4.2, v5.0 and v5.1	CVE-2019-9506
Validation of Elliptic Curve Parameters	07/23/2018	SIG Security Notice	Core Spec, v2.1 to v5.0	CVE-2018-5383

Wave 3

Tools

Tools of Wave 3: InternalBlue

By Dennis Mantz

- 2018: Masters Thesis
  - Custom patches to Broadcom BT firmware to send arbitrary packets
    - Worked on same platforms as "Nexmon" which was an existing Broadcom WiFi patching tool from SEEMOO
- Opened the door to much of the subsequent SEEMOO Lab work

Tools of Wave 3: Sniffle

By Sultan Qasim Khan

- 2019: Reliable BT sniffer w/ support for new PHY encodings added in BT spec v5.0

Tools of Wave 3: SwyenTooth & BrakTooth

By Matheus E. Garbelini

- 2020: SweynTooth provides customized Nordic BT firmware for sending arbitrary BLE packets, and then built a fuzzer on top and found a bunch of crashes
- 2022: BrakTooth provides customized Espressif BT firmware for sending arbitrary BT Classic packets, and then built a fuzzer on top and found a bunch of crashes
    - These are what I use in my current work

Wave 3

Talks

ARMIS®

DARK MENTOR

TEXAS INSTRUMENTS

SEEMO

SECURE MOBILE NETWORKING

BROADCOM®

DARK MENTOR

SILICON LABS

Wave 3 High Impact Talks

Over-the-air, pre/no-auth, Bluetooth controller, arbitrary code execution exploits

- **CVE-2018-1698** - “BleedingBit” - Seri et al. - TI CC2640, CC2650
- **CVE-2019-11516** - Ruge et al. - BCM20702, BCM4335C0, BCM4345B0, BCM4358A3, BCM4345C1, BCM20707, BCM4347B0/1, CYW20735B1, CYW20819A1
- **CVE-2019-15948** - Veronica Kovah - TI CC256*B, CC256*C, and WL18**
- **CVE-2020-15531** - Veronica Kovah - Silicon Labs EFR32* SoCs (EFR32BG*, EFR32FG*, EFR32MG*, & Silicon Labs Bluetooth Low Energy SDK before 2.13.3)

Wave 3 High Impact Talks

Over-the-air, pre/no-auth, Bluetooth controller, arbitrary code execution exploits

- **High impact..and yet…the full impact is unknown…**
- 😒

Silicon Vendors

TEXAS INSTRUMENTS

Silicon Vendors

Texas Instruments

Module-Makers (IDEK how many)

LG Innotek

JINGXUN

ACKme NETWORKS

Blue Radios A Wireless World

Rayson

Laird

MOKO BLUE

BNCOM

WNC

partron

Panasonic INDUSTRY

stollmann

muRata INNOVATOR IN ELECTRONICS

Silicon Vendors

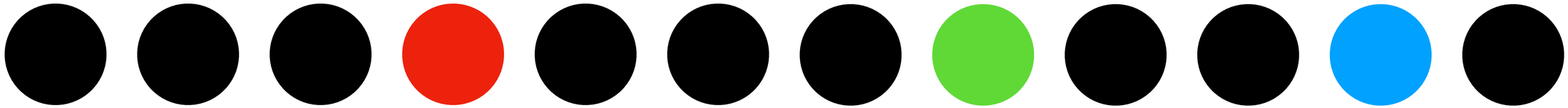
Texas Instruments

Module-Makers (IDEK how many)

LG Innotek, JINGXUN, ACKme Networks, Blue Radios A Wireless World, Rayson, Laird, MOKO BLUE, BNCOM, WNC, partron, Panasonic INDUSTRY, stollmann, muRata INNOVATOR IN ELECTRONICS

Product-Makers

3330 *registered* with Bluetooth SIG as of the time of writing!



Silicon Vendors

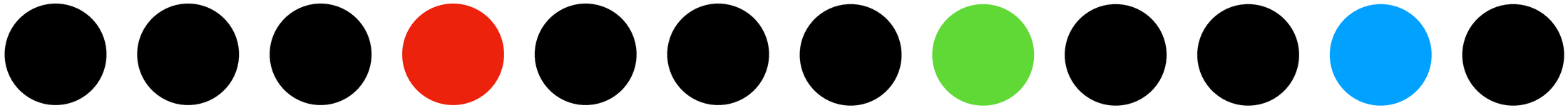
Texas Instruments

Module-Makers (IDEK how many)

LG Innotek, JINGXUN, ACKme Networks, Blue Radios A Wireless World, Rayson, Laird, MOKO BLUE, BNCOM, WNC, partron, Panasonic INDUSTRY, stollmann, muRata INNOVATOR IN ELECTRONICS

Product-Makers

3330 *registered* with Bluetooth SIG as of the time of writing!



Products

Silicon Vendors

Texas Instruments

Module-Makers (IDEK how many)

LG Innotek

晶讯 JINGXUN

ACKme NETWORKS

Blue Radios A Wireless World

Rayson

Laird

MOKO BLUE

BNCOM

WNC

partron

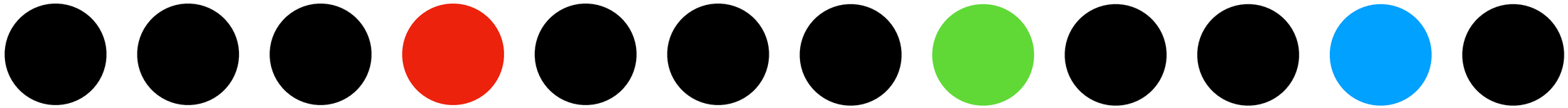
Panasonic INDUSTRY

stollmann

muRata INNOVATOR IN ELECTRONICS

Product-Makers

3330 *registered* with Bluetooth SIG as of the time of writing!



Products

Silicon Vendors

Texas Instruments

Module-Makers (IDEK how many)

LG Innotek

JINGXUN

ACKme Networks

Blue Radios — A Wireless World

Rayson

Laird

MOKO BLUE

BNCOM

WNC

partron

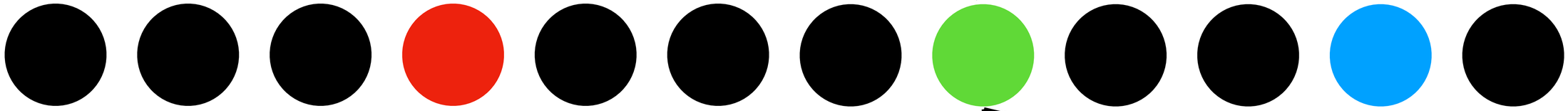
Panasonic INDUSTRY

stollmann

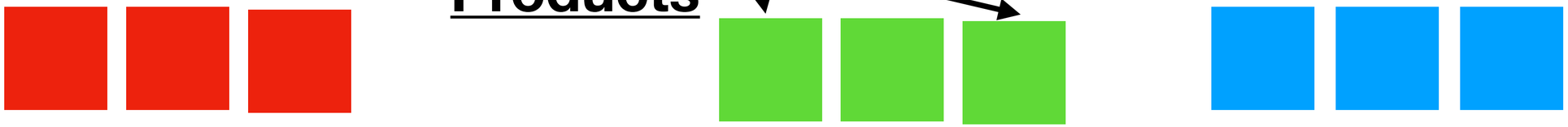
muRata — INNOVATOR IN ELECTRONICS

Product-Makers

3330 *registered* with Bluetooth SIG as of the time of writing!



Products



Silicon Vendors

Texas Instruments

Module-Makers (IDEK how many)

LG Innotek

MOKO BLUE

BNCOM

WNC

partron

JINGXUN

ACKme NETWORKS

Blue Radios — A Wireless World

Rayson

Laird

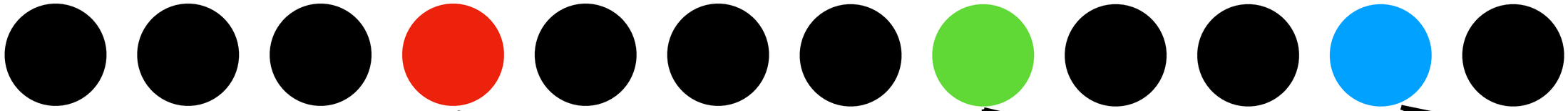
Panasonic INDUSTRY

stollmann

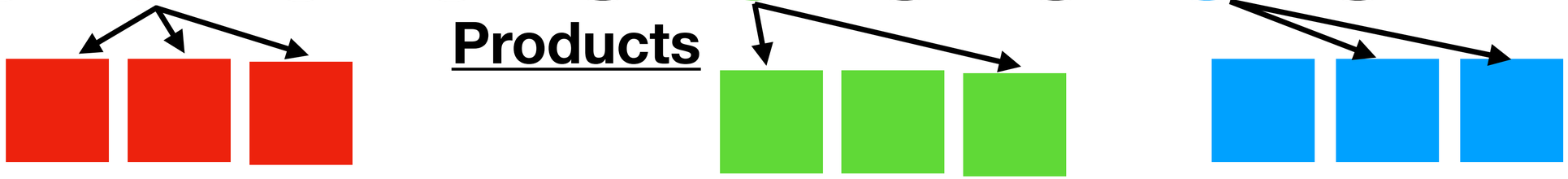
muRata — INNOVATOR IN ELECTRONICS

Product-Makers

3330 *registered* with Bluetooth SIG as of the time of writing!



Products



Silicon Vendors (>20)

BROADCOM
Qualcomm CSR
Texas Instruments
bluegiga SILICON LABS
Realtek
NORDIC SEMICONDUCTOR
SAMSUNG SEMICONDUCTORS
intel
MEDIATEK
ESPRESSIF
CYPRESS EMBEDDED IN TOMORROW
Infineon
NXP freescale semiconductor
Telink
RENESAS
MICROCHIP
em microelectronic A COMPANY OF THE SWATCH GROUP
YiCHiP
oppo
MARVELL
ISSC ISSC Technologies Corp.
BES TECHNIC

Module-Makers (IDEK how many)

LG Innotek
晶讯 JINGXUN
ACKme NETWORKS
BlueRadios A Wireless World
Rayson
Laird
MOKO BLUE
BNCOM
WNC
partron
Panasonic INDUSTRY
stollmann
muRata INNOVATOR IN ELECTRONICS

Product-Makers

3330 *registered* with Bluetooth SIG as of the time of writing!

What does Bluetooth pairing mean in practice?

And therefore what do successful attacks on pairing imply?

- Pairing establishes shared keys between two devices, so that traffic between them can be encrypted and/or authenticated
    - Therefore the absence of a requirement for pairing, or the capability to MitM the pairing, implies an ability to capture the data passing between devices
        - There are pairing modes (e.g. “just works”) which explicitly and intentionally don’t defend against MitM
- Some attack surfaces, such as the read/write interfaces of the GATT protocol, can be restricted to requiring encrypted and/or authenticated connections before they can be accessed

Wave 3 High Impact Talks

"Breaking the Bluetooth Pairing – Fixed Coordinate Invalid Curve Attack"
Biham & Neumann, Aug. 2018

- The Y coordinate in the Elliptic Curve Diffie-Hellman (ECDH) key exchange is not verified, and the protocol didn't require verification that the computed public key was valid, by actually being on the elliptic curve
- Two types of attacks, both of which grant MitM capability
  - "Semi-passive", 25% chance to passively decrypt subsequent traffic
  - "Fully-Active", 50% chance to actively decrypt subsequent traffic (while re-encrypting with a different key while MitMing)

Validation of Elliptic Curve Parameters	07/23/2018	SIG Security Notice	Core Spec, v2.1 to v5.0	CVE-2018-5383

https://crypto.iacr.org/2019/affevents/wac/medias/Neumann-BreakingBluetoothPairing.pdf

Our Fixed Coordinate Invalid Curve Attack

- The Fixed Coordinate Invalid Curve Attack is a new variant of the Invalid Curve Attack in which we exploit the ability to forge low order ECDH public keys that preserve the x-coordinate of the original public-keys.
- It is based on the following observations:
  - Only the x-coordinate of each party is authenticated during the Bluetooth pairing protocol.
  - The protocol does not require its implementations to validate whether a given public-key satisfies the curve equation.
- We describe two versions of our attack:
  - Semi-Passive.
  - Fully-Active.

Wave 3 High Impact Talks

"The KNOB is Broken: Exploiting Low Entropy in the Encryption Key Negotiation Of Bluetooth BR/EDR"
Antonioli et al., Aug. 2019

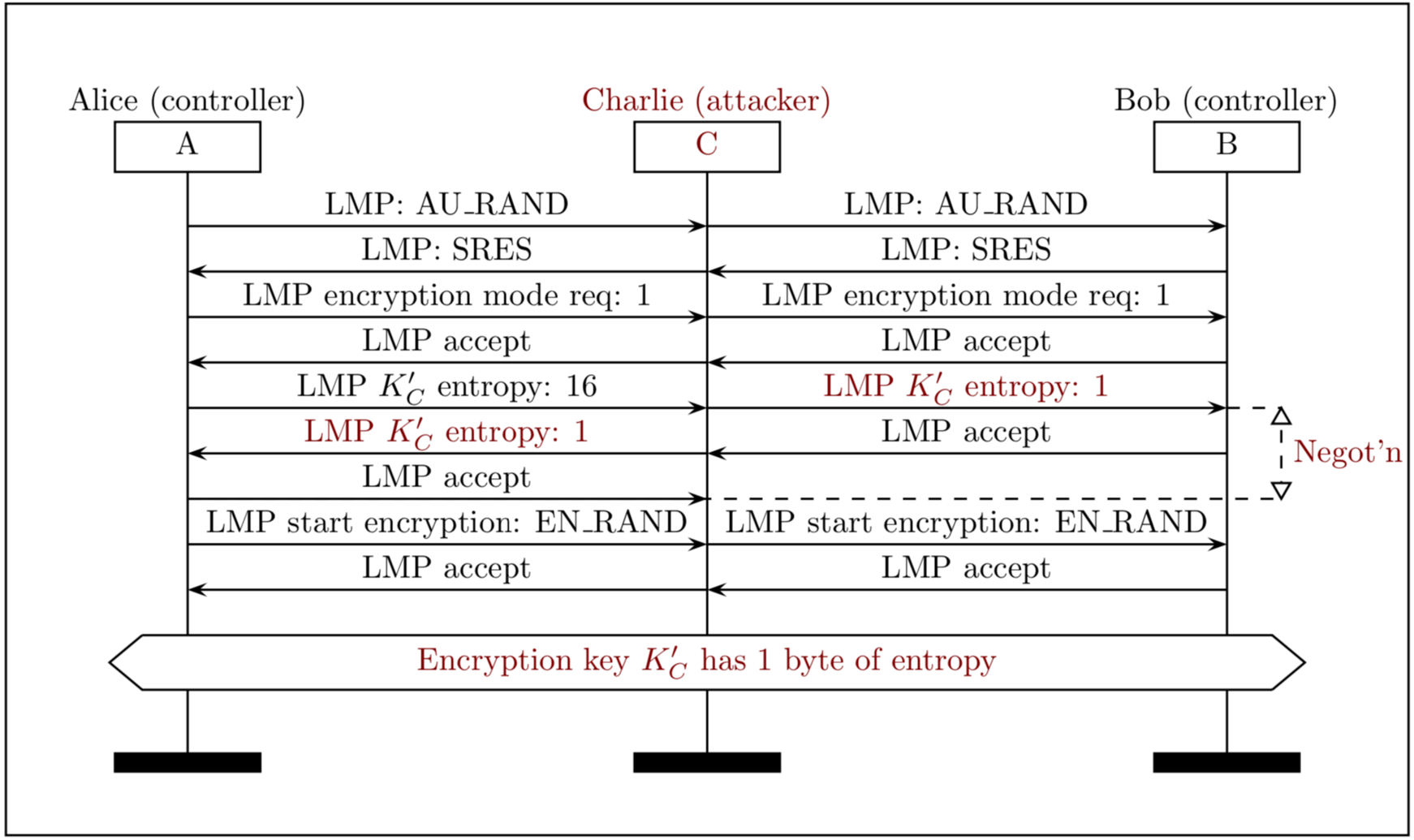
- Showed that BT Classic spec allowed for negotiating a link layer data encryption key entropy of ***8 bits*** instead of 128!
  - Trivially vulnerable to brute force attack
  - "To remedy the vulnerability, the Bluetooth SIG has updated the Bluetooth Core Specification to recommend a minimum encryption key length of 7 octets for BR/EDR connections."
    - 56 bits? DES anyone?

Key Negotiation of Bluetooth | 08/13/2019 | SIG Security Notice | Core Spec, v4.2, v5.0 and v5.1 | CVE-2019-9506

https://github.com/francozappa/knob

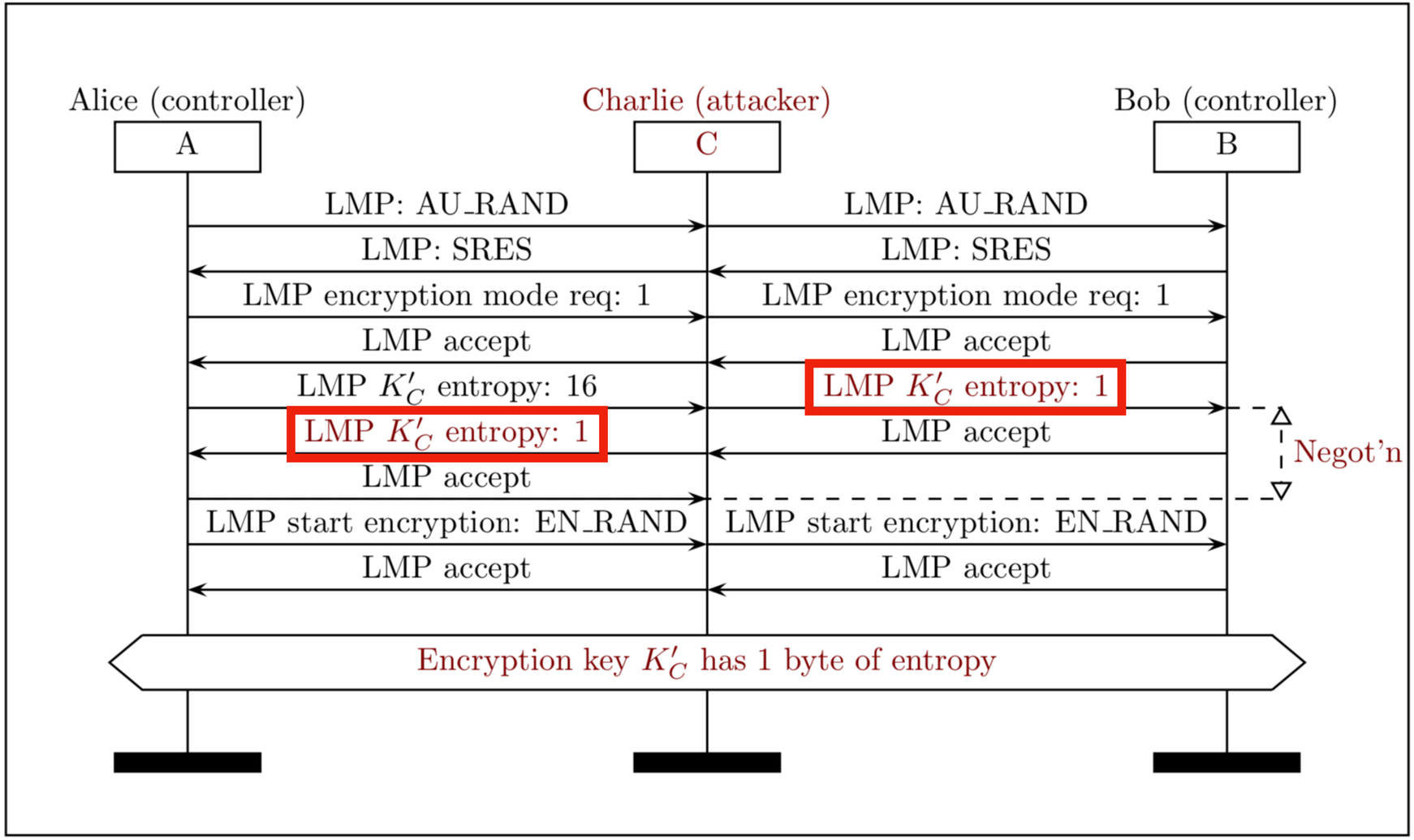
Adversarial Bluetooth Entropy Negotiation

- Charlie sets N=1 (K'_C's entropy), LMP is neither integrity protected nor encrypted



Adversarial Bluetooth Entropy Negotiation

- Charlie sets N=1 (K'_C's entropy), LMP is neither integrity protected nor encrypted



Wave 3 High Impact Talks

“Key Negotiation Downgrade Attacks on Bluetooth and Bluetooth Low Energy”
Antonioli et al., July 2020

- Subsequently showed that BLE spec allowed downgrading key entropy from 128 to 56 bits

https://github.com/francozappa/knob

Wave 3 High Impact Talks

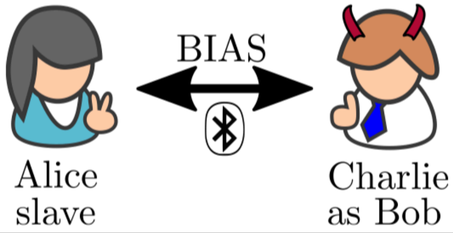
"BIAS: Bluetooth Impersonation Attack"
Antonioli et al., May 2020

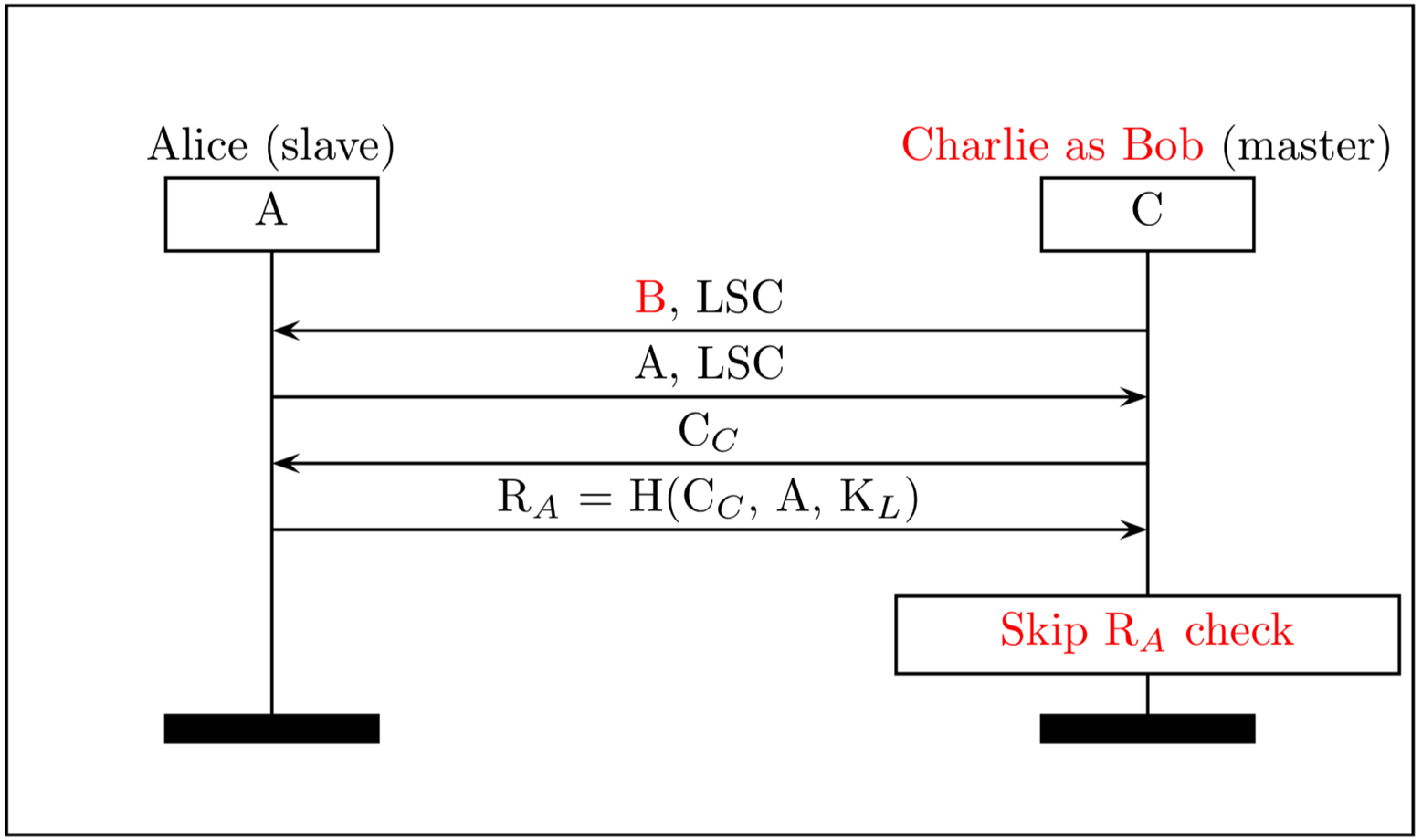
- Showed a standard-compliant way to get a MitM position, by utilizing the BT "*role switch*" operation, in which master and slave can reverse their roles, a lack of mutual authentication, and the capability to downgrade authentication protocols
  - Role switches can normally be used for instance to conserve power - e.g. a desktop initiates the connection to a keyboard, but then they role switch so that the keyboard can stay in lower power mode until it needs to send a key

Bluetooth Impersonation Attacks 05/18/2020 SIG Security Notice Core Spec, v2.1 to v5.2 CVE-2020-10135

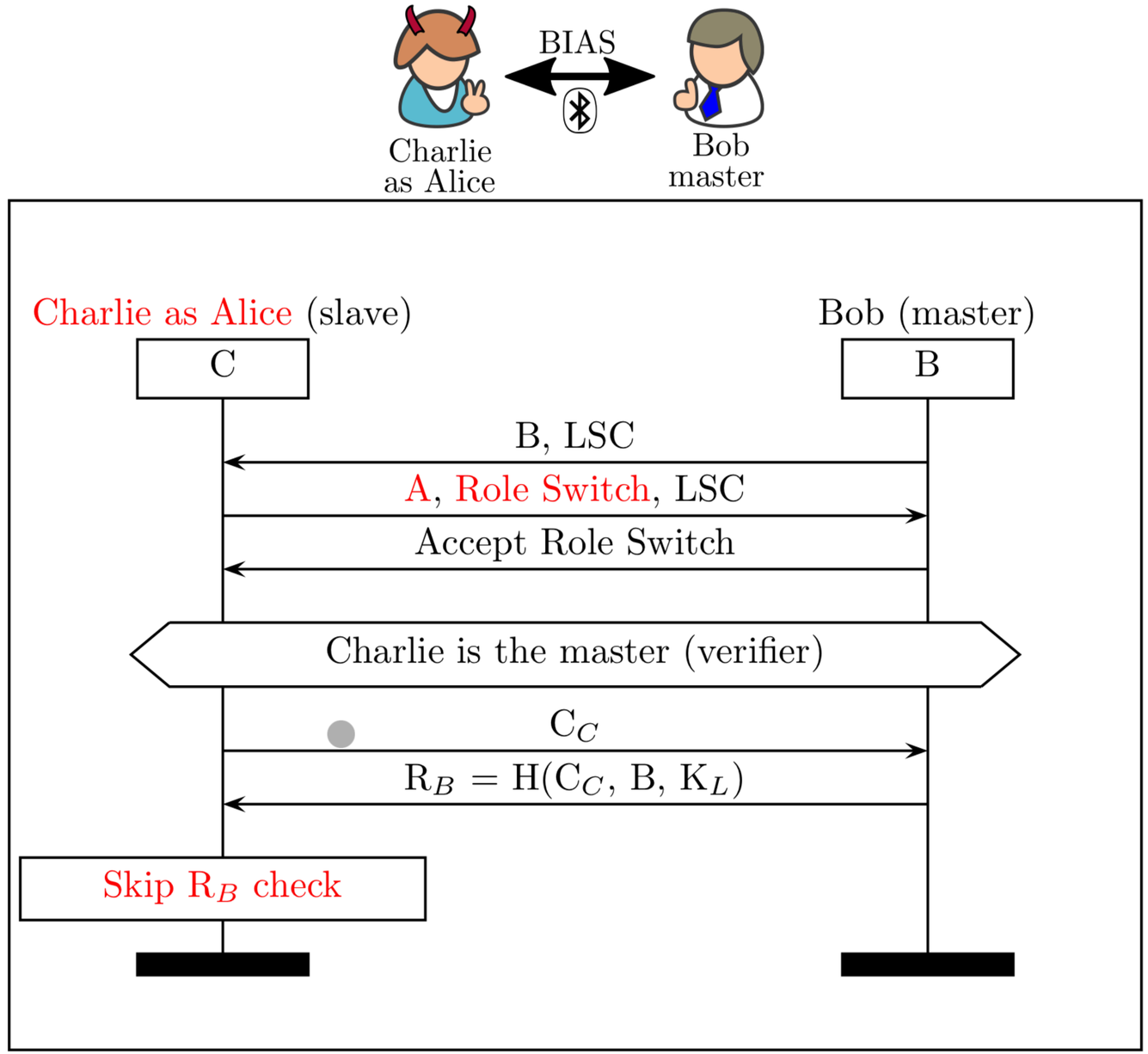
https://francozappa.github.io/publication/2020/bias/

BIAS Attack on LSC: Master Impersonation





BIAS Attack on LSC: Slave Impersonation



Wave 3 *Important* Talk

"On the Insecurity of Vehicles Against Protocol-Level Bluetooth Threats"
Antonioli & Payer, May 2022

- Evaluated devices against KNOB & BIAS, at a time when *ostensibly* everything should be patched, and found they were all still vulnerable
  - Detached vehicle infotainment units from KIA, Toyota, Mazda, Nissan and Subaru
  - In-situ car units in Suzuki IGNIS from 2021, a SKODA Fabia from 2020, and a SKODA Octavia from 2021

Wave 3 High Impact Talks

“BlueMirror: Reflections on Bluetooth Pairing and Provisioning Protocols”
Claverle & Esteves, May 2021

- Used cryptographic reflection attacks against pairing protocols to defeat almost all the authenticated key agreement protocols used during Bluetooth, Bluetooth Low Energy, and Bluetooth Mesh
  - The result of which would be an MitM attacker being able to complete pairing with a victim device, without knowing the key
    - Having successfully paired, the attacker could access additional attack surfaces on the device (such as GATT attributes which require an authenticated connection)
- Some of the first work on Bluetooth Mesh (mostly because it’s not super commonly used)

https://www.ieee-security.org/TC/SP2021/SPW2021/WOOT21/files/woot21-claverie-slides.pdf

Impersonation attack in Bluetooth Mesh Profile provisioning	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26560
Impersonation in the BR/EDR pin-pairing protocol	05/24/2021	SIG Security Notice	Core Spec, v1.0B to 5.2	CVE-2020-26555
Authentication of the Bluetooth LE legacy-pairing protocol	05/24/2021	SIG Security Notice	Core Spec, v4.0 to 5.2	N/A
Impersonation in the Passkey entry protocol	05/24/2021	SIG Security Notice	Core Spec, v2.1 to 5.2	CVE-2020-26558
Bluetooth Mesh Profile AuthValue leak	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26559
Malleable commitment in Bluetooth Mesh Profile provisioning	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26556
Predictable Authvalue in Bluetooth Mesh Profile provisioning leads to MITM	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26557

Impersonation attack in Bluetooth Mesh Profile provisioning	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26560
Impersonation in the BR/EDR pin-pairing protocol	05/24/2021	SIG Security Notice	Core Spec, v1.0B to 5.2	CVE-2020-26555
Authentication of the Bluetooth LE legacy-pairing protocol	05/24/2021	SIG Security Notice	Core Spec, v4.0 to 5.2	N/A
Impersonation in the Passkey entry protocol	05/24/2021	SIG Security Notice	Core Spec, v2.1 to 5.2	CVE-2020-26558
Bluetooth Mesh Profile AuthValue leak	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26559
Malleable commitment in Bluetooth Mesh Profile provisioning	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26556
Predictable Authvalue in Bluetooth Mesh Profile provisioning leads to MITM	05/24/2021	SIG Security Notice	Mesh Profile Spec, v1.0 to v1.0.1	CVE-2020-26557

State of the Art

Passive attacks | Active attacks

[SW05] [Rya13] | [Lin08] [Ros13] [BN19] [vTPFG21] [ATR20b]

Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

Synthesis

- Authenticated key agreements
- Secure key agreements according to the specification [Blu19a, Blu19b]
- Successfully attacked key agreements in this study

Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

Wave 3 High Impact Talks

“InjectaBLE: Injecting malicious traffic into established Bluetooth Low Energy connections”
Cayre et al., Jun 2021

- Showed how inter-packet variance could be used to inject malicious packets into existing communication between devices
  - Can’t defeat encryption on existing sessions (*if* they’re encrypted), but can inject non-encrypted traffic such as link-layer control packets

InjectaBLE: Injecting malicious traffic into established Bluetooth Low Energy connections	06/21/2021	SIG Security Notice	Core Spec, v4.0 to 5.2	CVE-2021-31615

https://laas.hal.science/hal-03193297/file/injectable_final_version.pdf

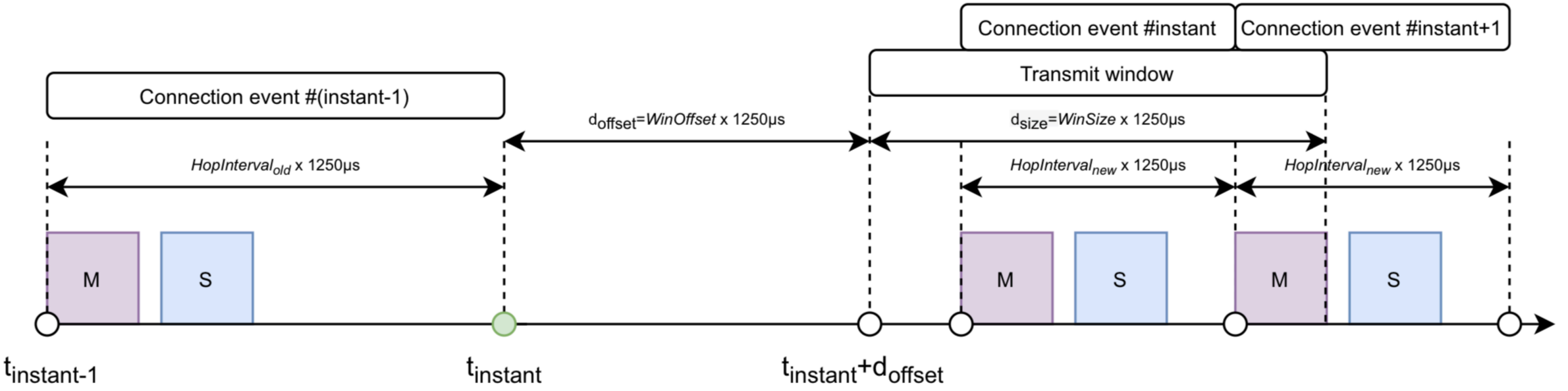


Fig. 2: Connection update procedure

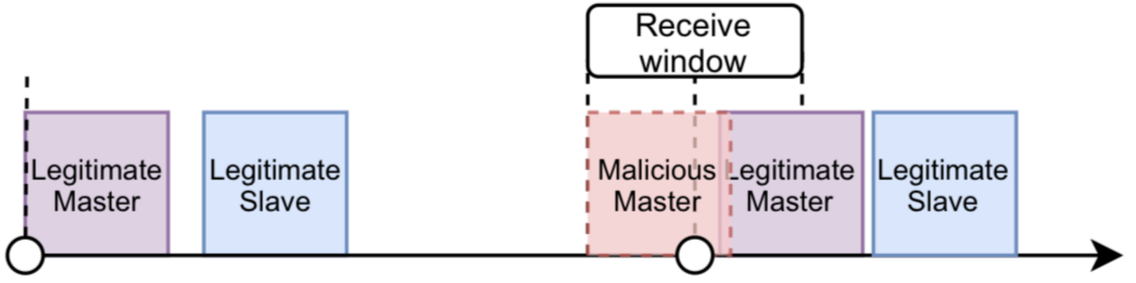


Fig. 3: Attack overview

injection point (challenge **C1** of Section IV). Subsection V-C describes how to inject the well-formed frame without altering the consistency of the connection state (challenge **C2**) and Subsection V-D describes how to check whether the injection is successful or not (challenge **C3**).

A. Clock (in)accuracy

Wave 3 High Impact Talks

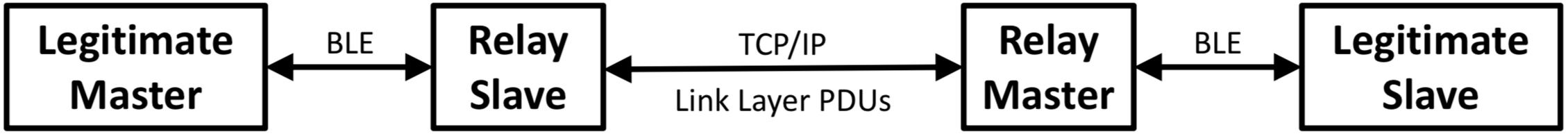
“Popping Locks, Stealing Cars, Breaking a Billion Other Things: Bluetooth LE Link Layer Relay Attacks”
Khan, Oct 2022

- Built a BLE link-layer relay device, and showed it could be used to defeat proximity (Received-Signal-Strength-Indicator - RSSI) based authentication on Teslas and BLE locks
  - Showed the feasibility of relaying auth challenges even over cellular networks
  - *Tool not released*, but the results generalize to any system that says “OK, I can tell the right person is in the right location, because I got a strong signal , and a fast response, from them during the authentication challenge”
    - Tesla *hasn’t fixed* because it is a known limitation of their threat model (that they state in their bug bounty and Pwn2Own contest rules for instance)
      - Almost certainly most other devices remain vulnerable to this

https://research.nccgroup.com/2022/05/15/technical-advisory-ble-proximity-authentication-vulnerable-to-relay-attacks/

https://hardwear.io/netherlands-2022/presentation/bluetooth-LE-link-layer-relay-attacks.pdf

Link Layer BLE Relaying Process



1. Relay master captures advertisements of legitimate slave
2. Relay slave mimics advertisements of legitimate slave
3. Legitimate master connects to relay slave
4. Relay slave notifies relay master of incoming connection
5. Relay master connects to legitimate slave
6. Relay master and slave forward link layer PDUs

Wave Goodbye

The (Piconet of Things) Patch Management Conundrum

- Even once protocol-level vulnerabilities are “patched”, the patches will flow downstream to (the *billions* of) devices very slowly, *if ever*
  - (Dozens of) Silicon vendor updates development frameworks
    - (Dozens of) Module makers adopt silicon makers’ updates
  - (Dozens of) OS/RTOS vendors update OSes
    - OS makers deploy software updates
  - (Thousands of) Device makers adopt silicon and/or module makers’ and/or RTOS vendor’s updates
    - Device makers deploy software/firmware updates … ***How?***

Update ETA wen?

- Some vendors push device firmware updates via associated mobile phone apps
  - Non-updated phone app most likely means non-updated firmware
  - Many devices (e.g. my gaming controller, bathroom scale, blood pressure monitor, and bed), don’t require phone app connectivity for normal usage
  - Devices may use architecturally-insecure firmware update channels[1] that themselves constitute vulnerabilities
- Many hundreds of millions of devices are *end-of-life* or from companies that are out of business and will never get updates
  - Many more are from simply irresponsible companies that will chose not to deploy updates to fix vulnerabilities

[1] https://www.armis.com/research/bleedingbit/

Call To Action!

Conclusion

- Bluetooth *vulnerability assessment* **is not yet a thing you can really do**!
- This is an active research topic I'm working on, but it needs more researchers working on it
- The starting point, as always, is to read related work
- I've organized the related work into a TiddlyWiki that I will continue to update over the coming years, and which others can contribute to via github PRs
  - https://darkmentor.com/bt.html

Fin

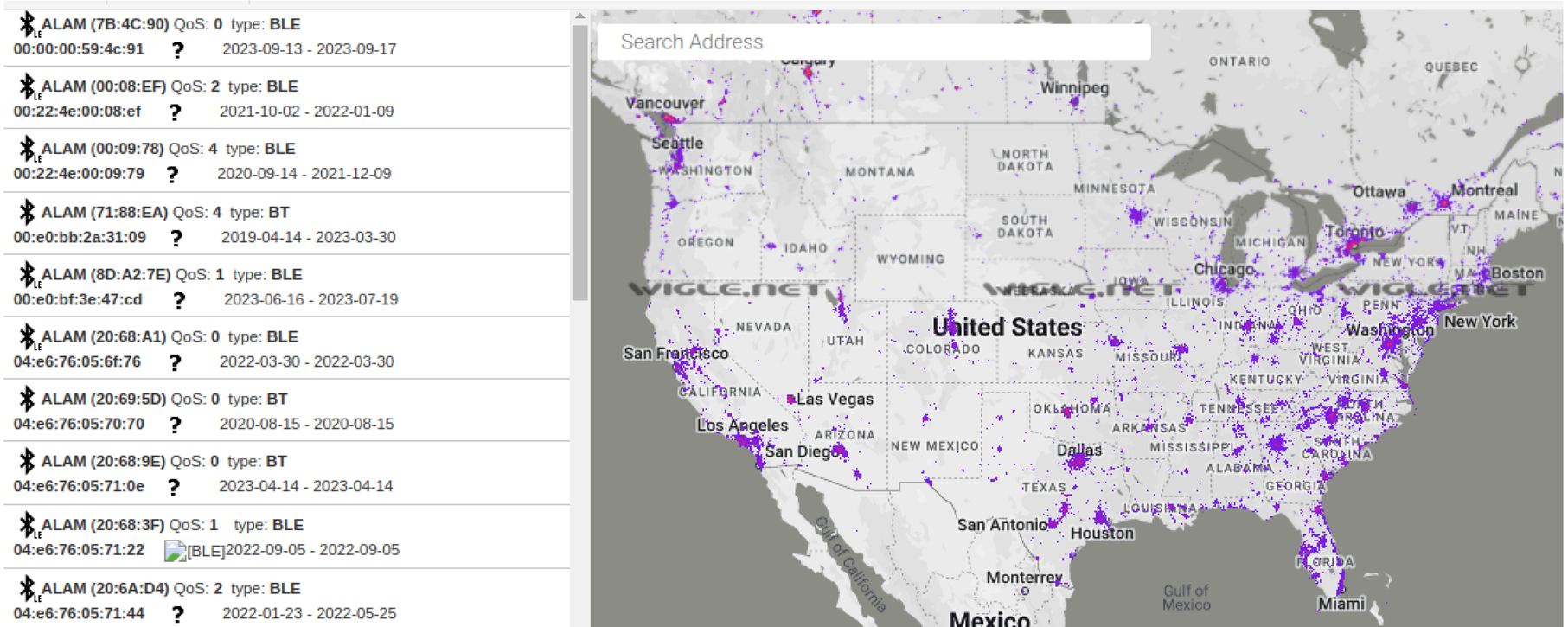
- BT research is cool
- But *OpenSecurityTraining2 (https://ost2.fyi , @OpenSecTraining) is cooler!*
  - We'll have BT classes eventually, but in the meantime there's so much other stuff to learn! Reverse Engineering, Vulnerabilities, Firmware, System Architecture!
- You should take a class, or *teach* a class!

Backup

Tools of Wave 3

Honorary mention: WiGLE.net

- Added BT scanning in 2019
- Crowdsourced data allows gathering some limited insights that would not otherwise be available to researchers

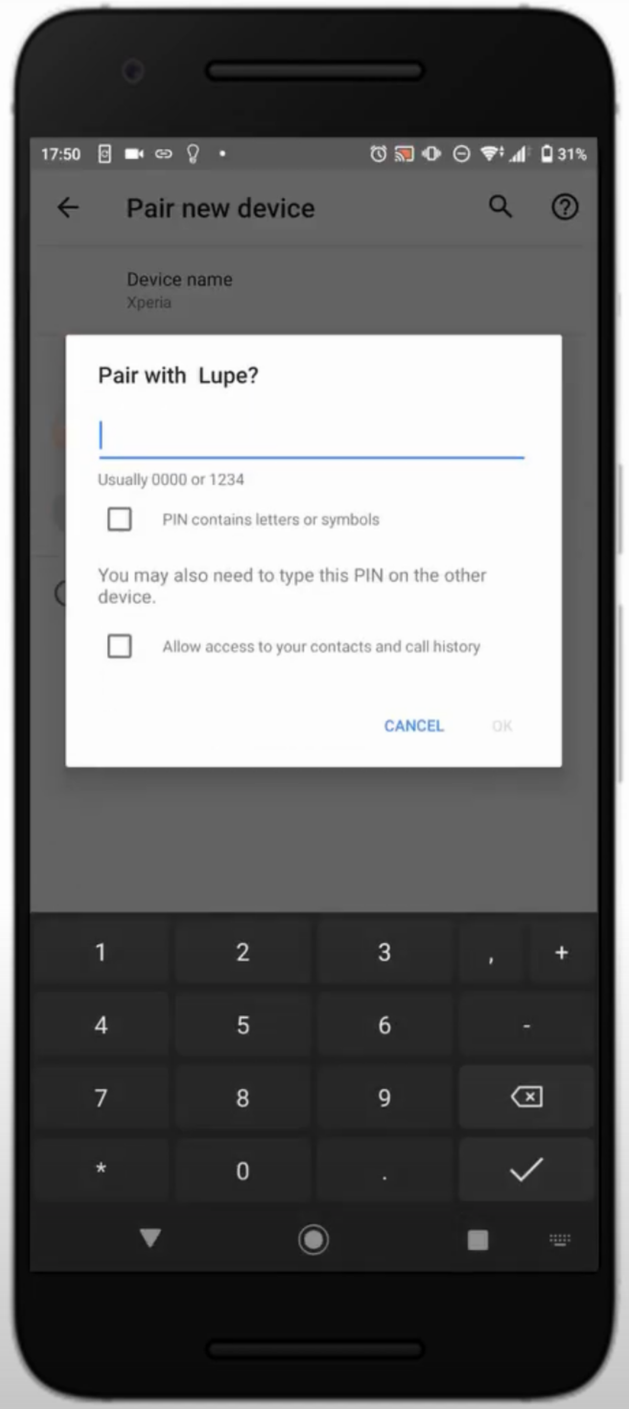


Wave 3 High Impact Talks

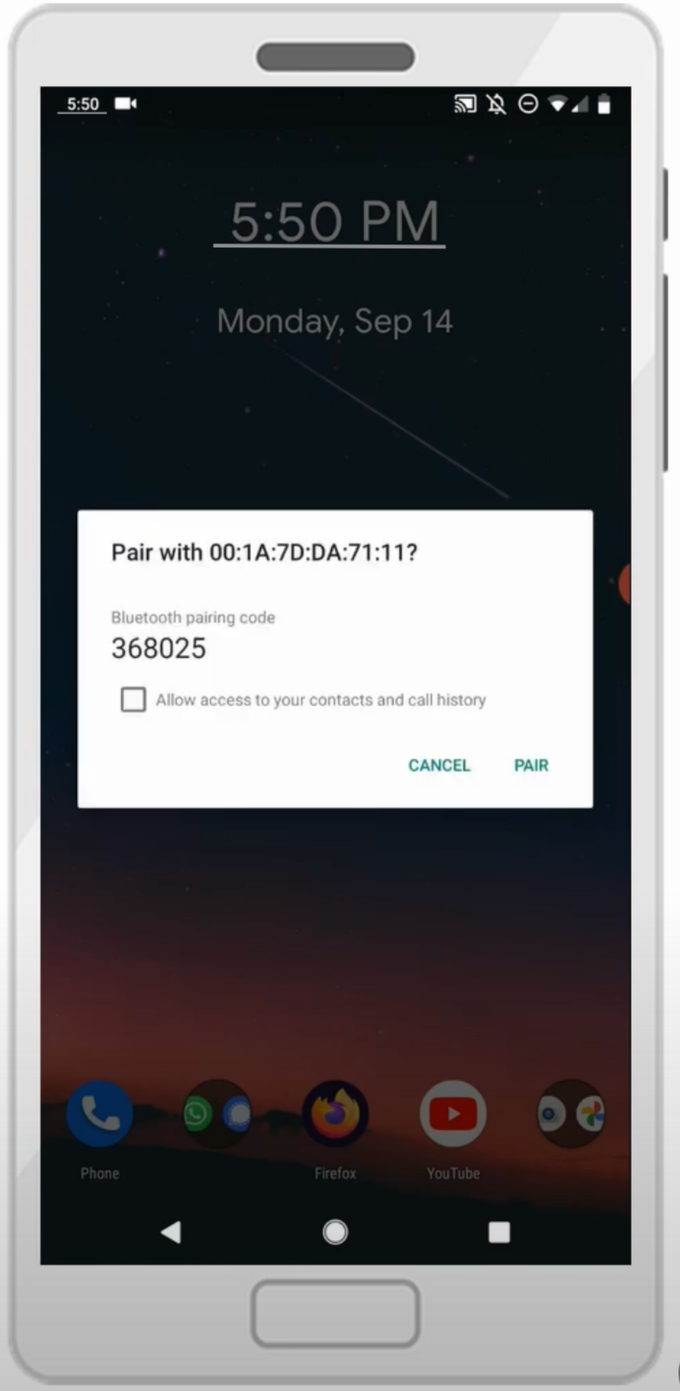
“Method Confusion Attack on Bluetooth Pairing”
von Tschirschnitz & Peuckert et al., May 2021

- Uses selective jamming (adapted from Cayre 2020) to prevent an Initiator device from connecting to a Responder device, and then impersonates the Responder
- The MitM Responder lies about its capabilities, to cause a “Passkey Entry” pairing to occur
- Then a MitM Initiator connects to the original target Responder, and lies about its capabilities, to cause a “Numeric Comparison” pairing to occur
- The human sees the initiator saying “Enter this 6 digit code on the other device”, and the responder saying “Enter the 6 digit code”, and they’re tricked into entering the code, which then gives it to the MitM, which uses it to complete the other pairing
    - Fundamentally this is about tricking the human, but they did a study and found people were easily confused into completing the pairing to the MitM’s advantage

Pairing Mode Confusion in BLE Passkey Entry	12/09/2022	SIG Security Notice	Core Spec v4.0 to 5.3	CVE-2022-25836
Pairing Mode Confusion in BR/EDR	12/09/2022	SIG Security Notice	Core Spec v1.0B to 5.3	CVE-2022-25837




```
lupe@kos /home/lupe/Files/BThack master ● sudo ./attack.py
-i 51 -r 52 -a auto -p Lupe
[i] Detected sniffers:
 > Sniffer #0: fw version 1.4
 > Sniffer #1: fw version 1.4
 > Sniffer #2: fw version 1.4
[i] Starting advertisements sniffing on channel 37 ...
b'\x11\x01%J\x00\x00\x00\x00\x00@\x0foN5yeX\x02\x01\x02\x05\tLup
e'
58:65:79:35:4e:6f
b'\x11\x01&?\x00\x00\x00\x00\x00@\x0foN5yeX\x02\x01\x02\x05\tLup
e'
58:65:79:35:4e:6f
Error occured when attempted to disable radios sniffing mode
Jamming all advertisements of 58:65:79:35:4e:6f
jamming b'Lupe' at 13
[i] Detected sniffers:
 > Sniffer #0: fw version 1.4
 > Sniffer #1: fw version 1.4
 > Sniffer #2: fw version 1.4
[i] Starting advertisements reactive jamming on channel 37 ...
b'Forking Initiator process\n'
b'Packet Log: /tmp/hci_dump_responder.pklg\n'
b'Forking Initiator process\n'
b'Packet Log: /tmp/hci_dump_initiator.pklg\n'
b'RESP: registering callbacks\n'
b'INIT: registering callbacks\n'
b'dev: 52\n'
b'using: 52\n'
b'Found Specified USB device\n'
b'dev: 51\n'
b'using: 51\n'
b'Found Specified USB device\n'
b'USB Path: 01-04\n'
b'RESP: Initialized\n'
b'RESP BThack up and running on 00:AA:00:AA:00:AA.\n'
b'USB Path: 01-01\n'
b'INIT: Initialized\n'
b'RESP: Identity resolving failed\n'
b'RESP: Connection complete\n'
Target has connected to MitM responder -> Stop jamming
b'RESP: victim iocap KEYBOARD_DISPLAY -> waiting for victim responde
r\n'
b'INIT: Responder is waiting for victim responder iocap -> resolving
\n'
b'INIT: Start scanning!\n'
b'INIT: Found targeted remote (58:65:79:35:4E:6F) with UUID 1111\n'
b'INIT: Connecting to attack target responder\n'
b'INIT: Connection complete\n'
b'INIT: NoP selected\n'
b'RESP: performing NoP [DISPLAY_ONLY]\n'
b'RESP: old tk 227021\n'
```



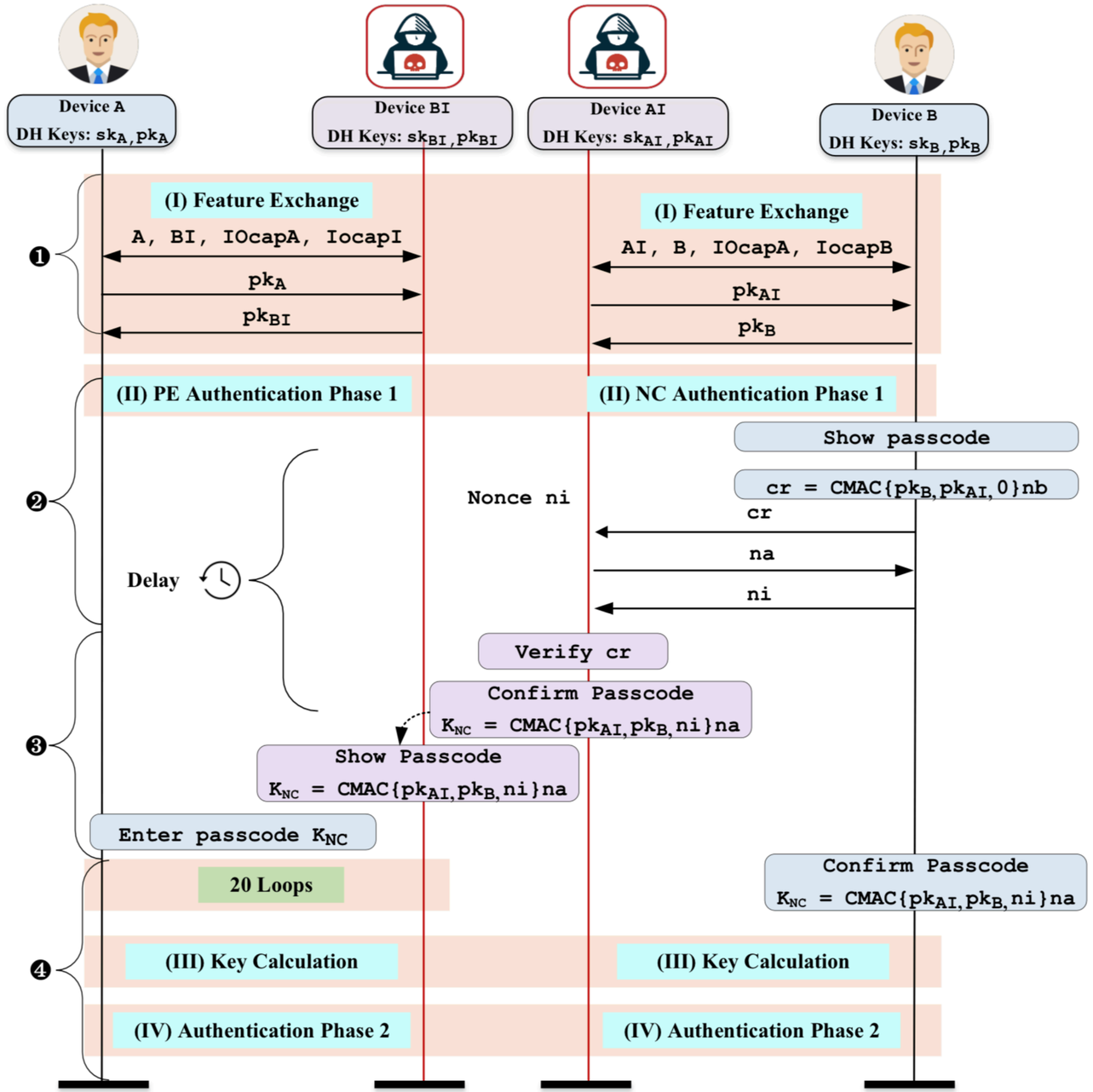


Fig. 3: Illustration of the Method Confusion attack.

Image from “Extrapolating Formal Analysis to Uncover Attacks in Bluetooth Passkey Entry Pairing”