



# nRootTag

## Exploiting Find My and Transforming Computers Into Unauthorized Trackers

Junming Chen, Xiaoyue Ma, Lannan Luo, Qiang Zeng

---

33<sup>th</sup> DEFCON  
August 7-10, 2025  
Las Vegas, NV, USA



# Techniques for Tracking

## ■ IP based Tracking

Accuracy	State
Latency	Hard to tell
Accessibility	ISP
Coverage	Determine by ISP

Geolocation data from **IP2Location** Product: DB6, 2025-3-1

IP: [REDACTED]	ISP: Verizon Business
ADDRESS:	ORGANIZATION: Not available
COUNTRY: United States	LATITUDE: 39.0395
REGION: Virginia	LONGITUDE: -77.4918
CITY: Ashburn	

Incorrect location? [Contact IP2Location](#) [view map](#)

Geolocation data from **ipinfo.io** Product: API, real-time

IP: [REDACTED]	ISP: Not available
ADDRESS:	ORGANIZATION: AS701 Verizon Business
COUNTRY: United States	LATITUDE: 38.7934
REGION: Virginia	LONGITUDE: -77.2716
CITY: Burke	

Incorrect location? [Contact ipinfo.io](#) [view map](#)

Geolocation data from **DB-IP** Product: API, real-time

IP: [REDACTED]	ISP: Verizon Business
ADDRESS:	ORGANIZATION: Verizon Business
COUNTRY: United States	LATITUDE: 38.9072
REGION: District of Columbia	LONGITUDE: -77.0369
CITY: Washington D.C.	

Incorrect location? [Contact DB-IP](#) [view map](#)

Geolocation data from **IPRegistry.co** Product: API, real-time

IP: [REDACTED]	ISP: Verizon Business
ADDRESS:	ORGANIZATION: Verizon Business (verizonenterprise.com)
COUNTRY: United States	LATITUDE: 38.7918
REGION: Virginia	LONGITUDE: -77.28017
CITY: Burke	

Incorrect location? [Contact IPRegistry.co](#) [view map](#)



# Techniques for Tracking

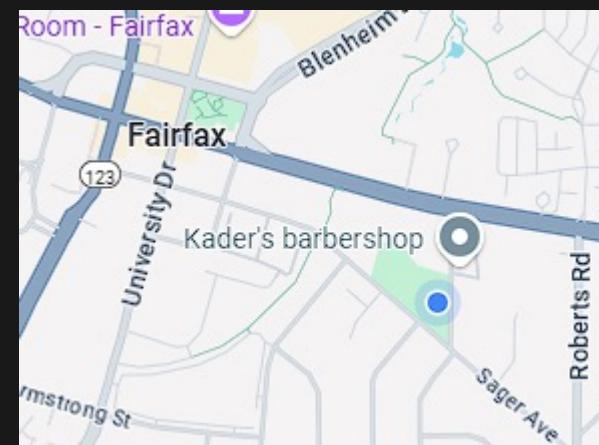
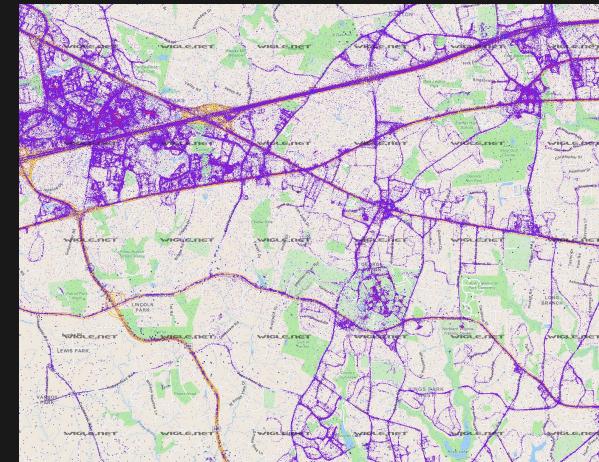
- Beacon / Wi-Fi based tracking

Accuracy Street Blocks

Latency Days, Months

Accessibility Everyone (wigle.net)

Coverage  $\leq$  Wi-Fi Coverage





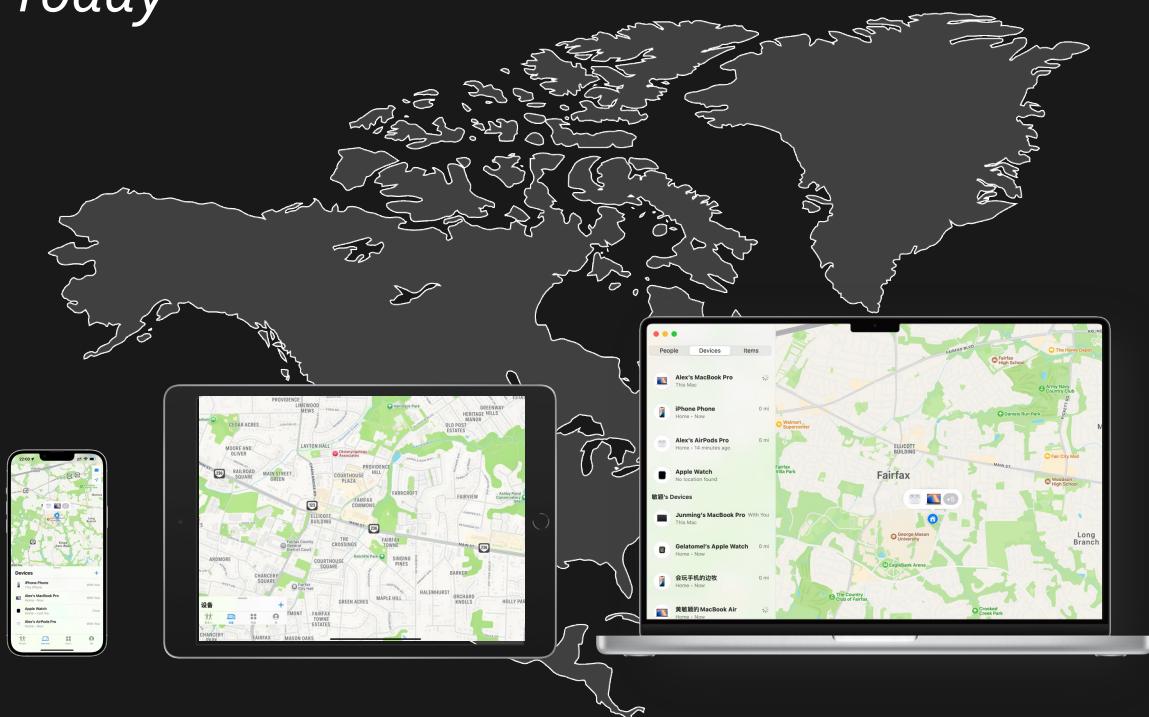
# Techniques for Tracking

- Crowdsourcing tracking



*Protagonist of Today*

Over **1.5 billion** iPhones forming  
a global tracking network



# Media Coverage

---



“ A major flaw in Apple's Find My service

that lets hackers turn pretty much any device—even  
Android phones—into a fake AirTag

*Yahoo Tech*

”

also featured in [Dailymail](#), [Forbes](#), [TechRadar](#), [ACM](#), and more

# Who May Be Interested?

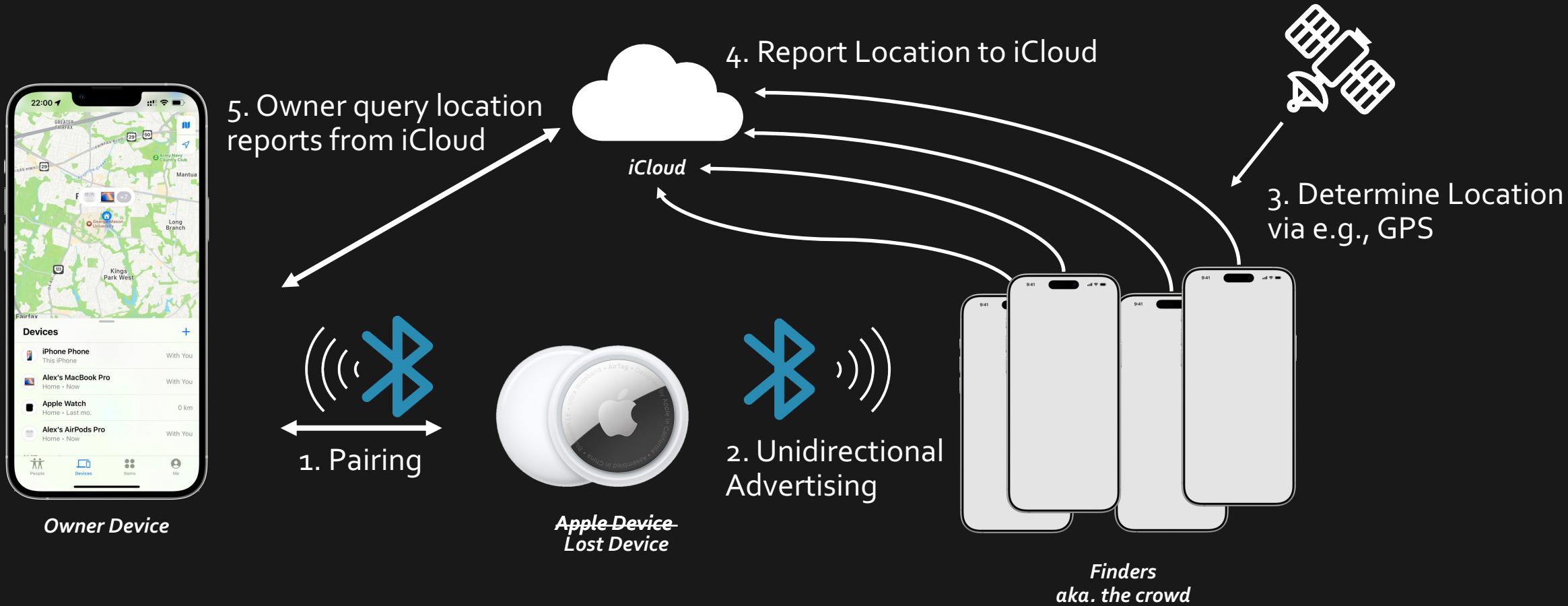


- Intelligence Departments
- Law Enforcement Agencies
- Advertising-Focused Mobile Apps and Software
- Cybercriminals





# How Offline Finding Works?



# The Explore Begins



**Design:** End-to-End Encryption, data exclusive to owner

**Challenge:** Put cryptographic key into limited payload space

**Solution:** Borrow extra space from BLE address

## Addr. Type (2bits)

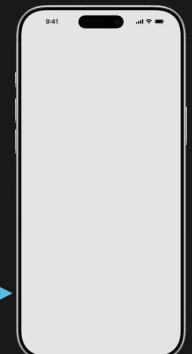
**MAC address**  
6 bytes

11 2465B8942D55

Secp 224R1  
Public Key

**BLE Payload**  
31 bytes

... 1234567812345678 ...



# The Explore Begins



## BLE Address Types Overview (*from Internet*)

Types	MSB	LSB	
Public	OUI	Manufacturer Specific	Managed by IEEE
NRPA	0 0	Random part of NRPA	
RPA	0 1	Random part of prand	Randomly generated by device
Random Static	1 1	Random part of static address	Apple's choice

Only the public address can identify its assigned vendor!



# The Explore Begins

---

## Address Type Definition

The Find My network advertising payloads replaces the AdvA field of the advertising PDU defined by the BT SIG with the first 0 to 5 bytes of the current key. The nearby or separated state of the accessory determines the current key. Most significant bits of byte 0 shall be 0b11, indicating a static device address.

*Apple Inc. Find My Network Accessory Specification*

# The Explore Begins



## OpenHayStack

- Pioneer research
- Based on reverse engineering
- Cheap and cool project for DIY

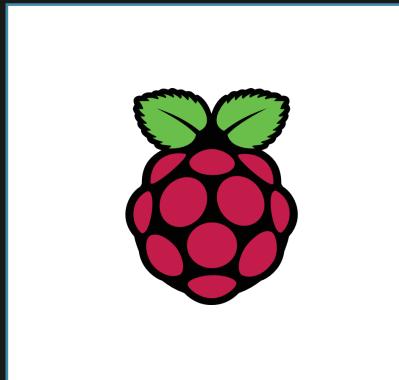
[seemoo-lab / openhaystack](#) Public

Build your own 'AirTags' 📺 today! Framework for tracking personal Bluetooth devices via Apple's massive Find My network.

[owlink.org](#)

AGPL-3.0 license

11.2k stars   532 forks   Branches   Tags   Activity



# The Explore Begins

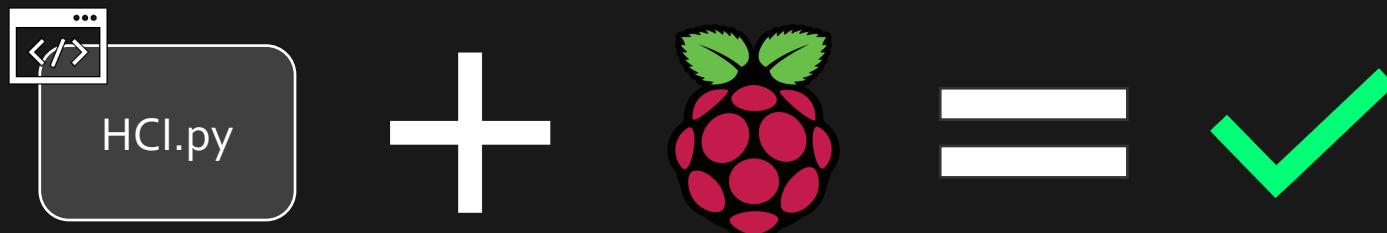


## Replicating OpenHayStack

```
# ./hci.py --key 3q2+7xI0v52mn5n0uM7lgUooCh3JcbYMfN/JVQ==  
key      (28) deadbeef1234 ...
```

### Using Raspberry Pi

Addr: DE:AD:BE:EF:12:34 ← Expected Result





# The Explore Begins

---

When checking address of adapter, we noticed something

```
# ./hci.py --key 3q2+7xI0v52mn5n0uM7lgUooCh3JcbYMfN/JVQ==  
key      (28) deadbeef1234 ...
```

Using PC and Dongle

Addr: **30:05:05:EC:A6:02** ← Unexpected Result





# The Explore Begins

- Issues in OHS Code (*first*)

```
print(f"payload ({len(adv):2}) {adv.hex()}"  
  
# Set BLE address  
run_hci_cmd(["0x3f", "0x001"] + bytes_to_strarray(addr,  
subprocess.run(["systemctl", "restart", "bluetooth"])  
time.sleep(1)  
  
# Set BLE advertisement payload  
run_hci_cmd(["0x08", "0x0008"] + [format(len(adv), "x")])
```

## Configure address

*this is a cmd for changing Cypress adapter address*



# The Explore Begins

## ■ Issues in OHS Code (*second*)

```
# Set BLE advertising mode
interval_enc = struct.pack("<h", interval_ms)
hci_set_adv_params = ["0x08", "0x0006"]
hci_set_adv_params += bytes_to_strarray(interval_enc)
hci_set_adv_params += bytes_to_strarray(interval_enc)
hci_set_adv_params += ["03", "00", "00", "00", "00", "00", "00", "00", "00"]
hci_set_adv_params += ["07", "00"]
run_hci_cmd(hci_set_adv_params)
```

*Own\_Address\_Type:*

Value	Parameter Description
0x00	Public Device Address ( <b>default</b> )
0x01	Random Device Address
0x02 – 0xFF	Reserved for future use

*Bluetooth Specification*



It conflicts with Apple's Spec!  
Public != Static



# The Explore Begins

- What's the *address type* for E4:65:B8:94:2D:55?

Hint, E4 in binary is **11100100**

Types	MSB	LSB
Public	OUI	Manufacturer Specific
NRPA	0 0	Random part of NRPA
RPA	0 1	Random part of prand Hash
Random Static	1 1	Random part of static address



# The Explore Begins

- Did you get it right?

Wireshark says

Results  
E4:65:B8 Espressif Inc.

Types	MSB	LSB
Public	1 1	OUI      Manufacturer Specific
Random Static	1 1	Random part of static address



# The Explore Begins

- BLE Address Types Overview (*the correct version*)

Types	TxAdd	MSB	LSB
<b>Public</b>	0	OUI	Manufacturer Specific
<b>NRPA</b>	1	0   0	Random part of NRPA
<b>RPA</b>	1	0   1	Random part of prand Hash
<b>Random Static</b>	1	1   1	Random part of static address

**TxAdd is required to determine address type!**

# The Explore Begins

---



Find My network accepts **more address types!**

Apple, OpenHayStack, and derived works all missed this.

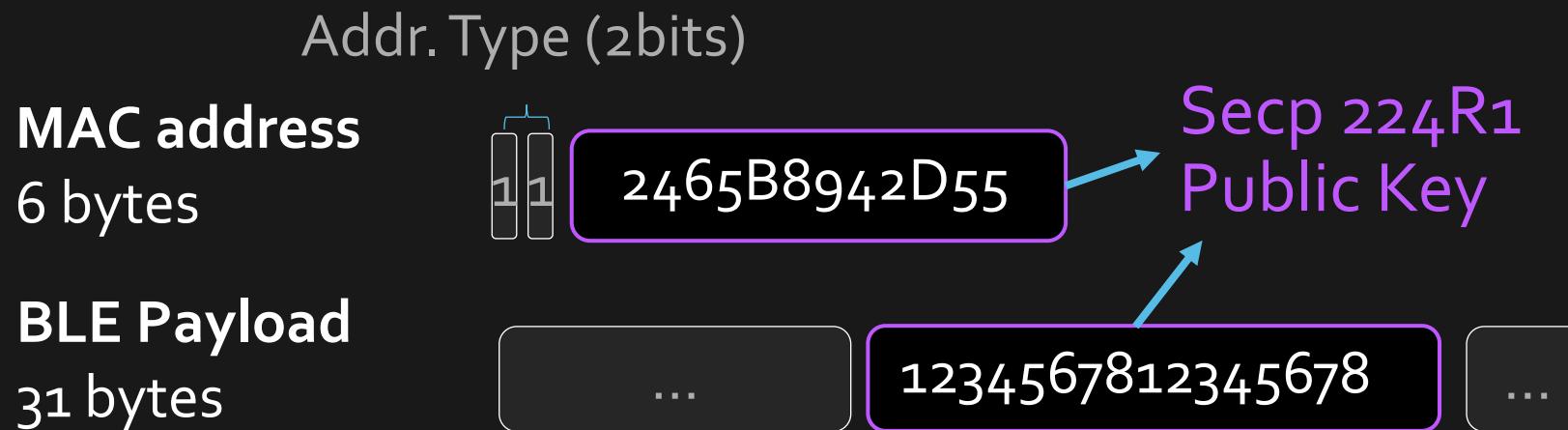
Operating System	Address Type
Linux	Public Address
Android	RPA
Windows	NRPA

Unprivileged Address Types

# The Explore Begins



## Splitting Public Key and Use as MAC Address



# Exploit Deep Dive

---



## Tailored Attacks

- Attack for *Linux*
- Attack for *Everything*

# Exploit Deep Dive

---



➤ Attack for *Linux*

Linux uses BlueZ → BlueZ Adv. with Public Addr. → Only 47,054 OUI

**Rainbow Table!**

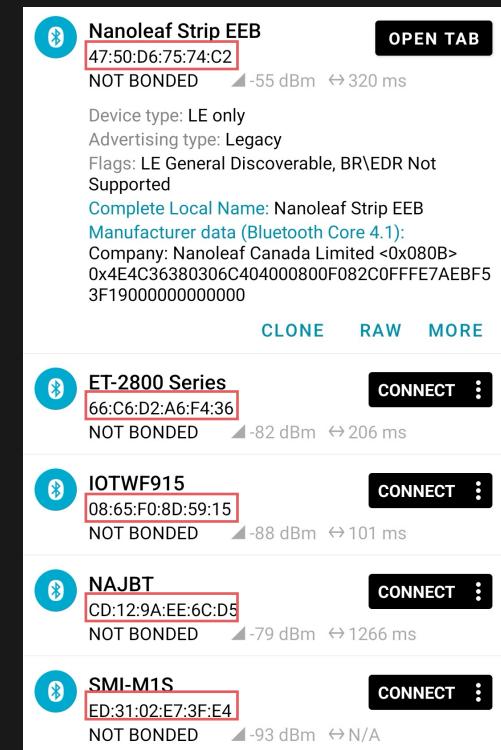
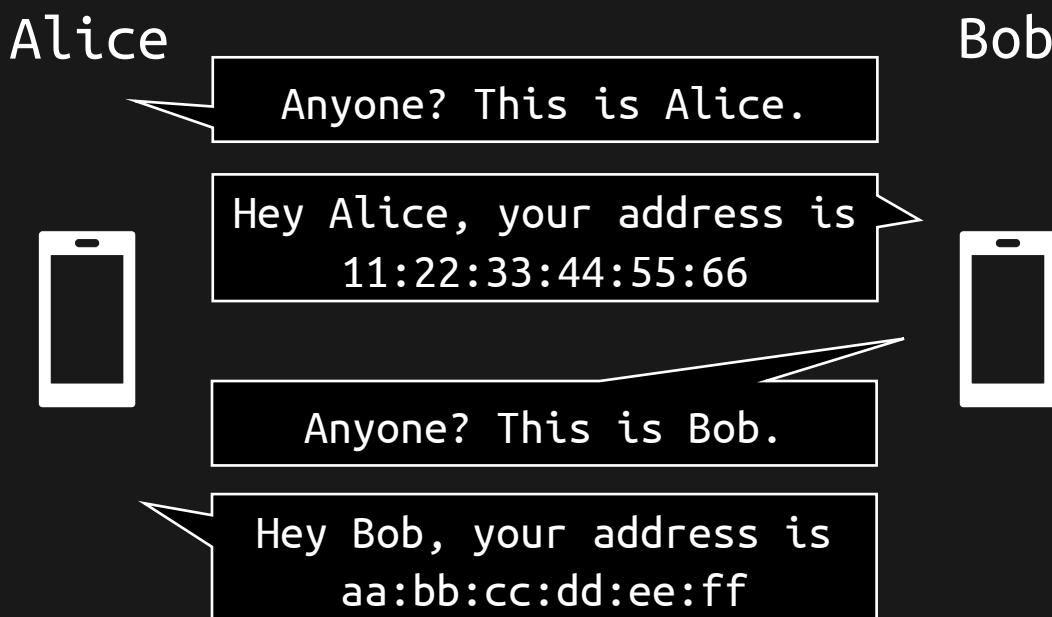
Total space: 20.x TB



# Exploit Deep Dive

## ➤ Attack for *Everything*

- Android prevented Bluetooth tracking
- Allowed reading address from received Msg.



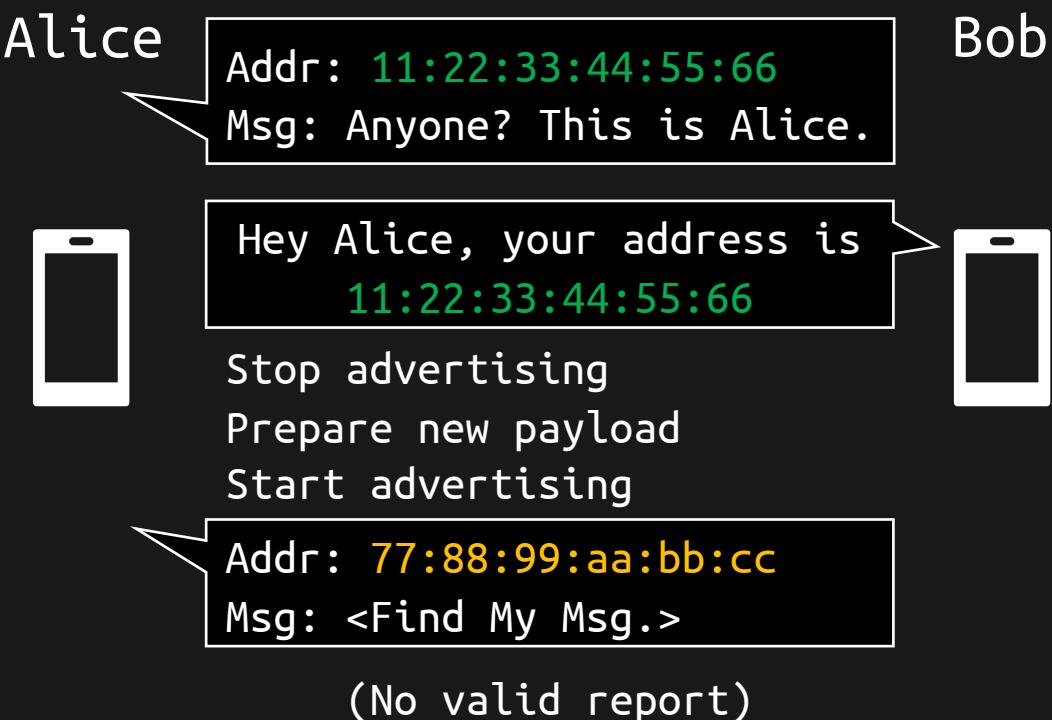
nRF connect shows address  
of surrounding devices



# Exploit Deep Dive

## Device Address Rotation

- Address changes when updating Msg. payload

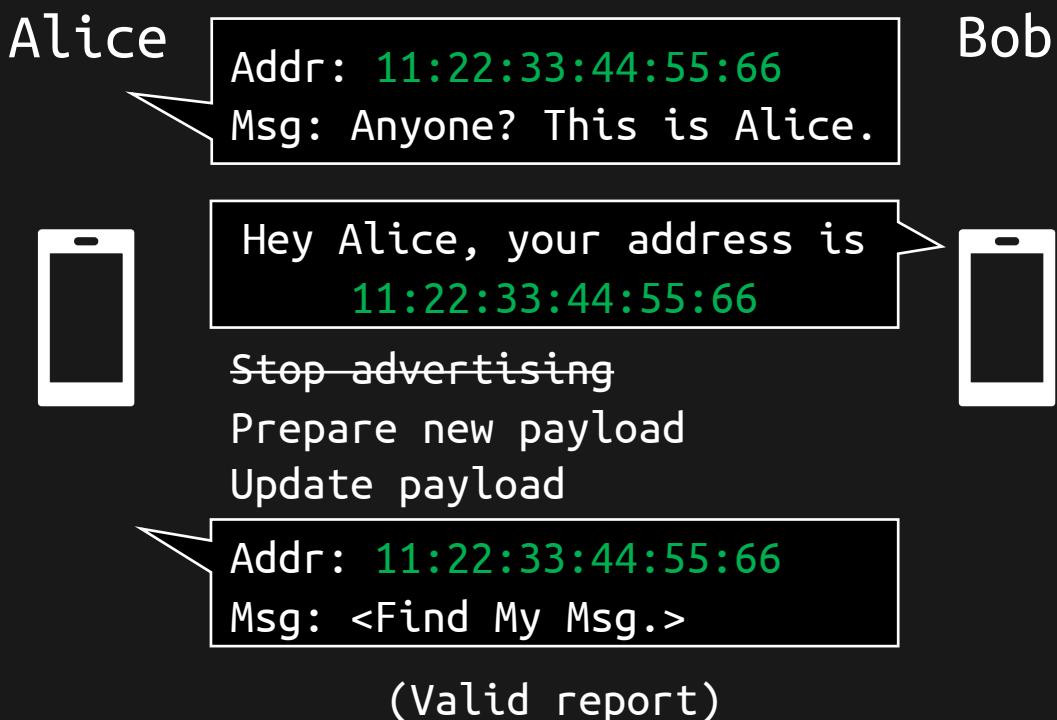




# Exploit Deep Dive

## Device Address Rotation (Cont.)

- Android ≥ 8 allows “hot update”



# Exploit Deep Dive

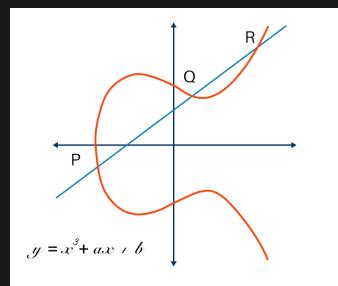


## Summary of the Exploit

1. Applications can advertise without root privilege
2. We can find a key for tracking



Generate Random #



Compute Public Key



```
{  
    If 1==1  
}
```

Compare



# Benchmark

Attacks should be practical in dimensions of time and cost

Model	KPS In Billion	Purchase Price	Price-to-KPS Ratio*	Rent Price	Rent-to-KPS Ratio*
RTX 3070	2.64	\$499	189	\$0.13	0.050
RTX 3080	4.90	\$699	143	\$0.22	0.045
RTX 4090	8.22	\$1,599	195	\$0.40	0.049
A100 80GB	9.07	\$13,224	1458	\$1.60	0.178
H100 80GB	11.91	\$48,200	4047	\$2.14	0.180

\*the smaller the more efficient (cheaper).



# Benchmark

**2.76**

Estimated Minutes

90% possibility finding a match  
using 200 RTX 3080

**2.20**

Estimated USD

for renting RTX3080

**66**

Measured Seconds

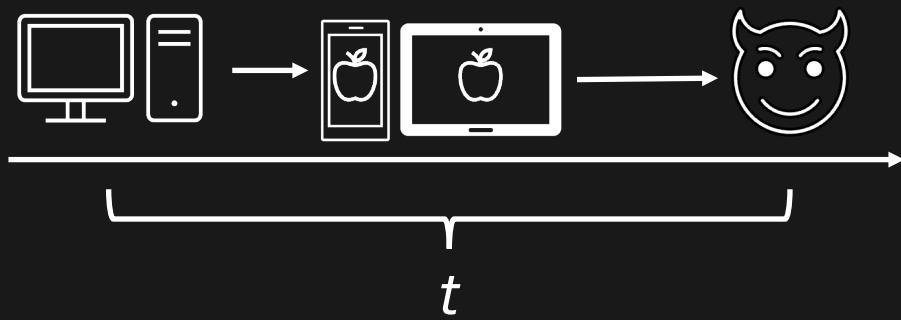
in reality using 200 units

For a few targets, Don't Store, Don't Buy, Just Rent!



# Benchmark

## Latency Measurement



**6.75**  
Avg. minutes  
In University Campus

**8.09**  
Avg. minutes  
At Single Family House

Performs just like a normal AirTag



# Benchmark

## Accuracy Measurement

Confidence	Spread
1	886 ft
2	184 ft
3	88.6 ft





# Benchmark

## Accuracy Measurement



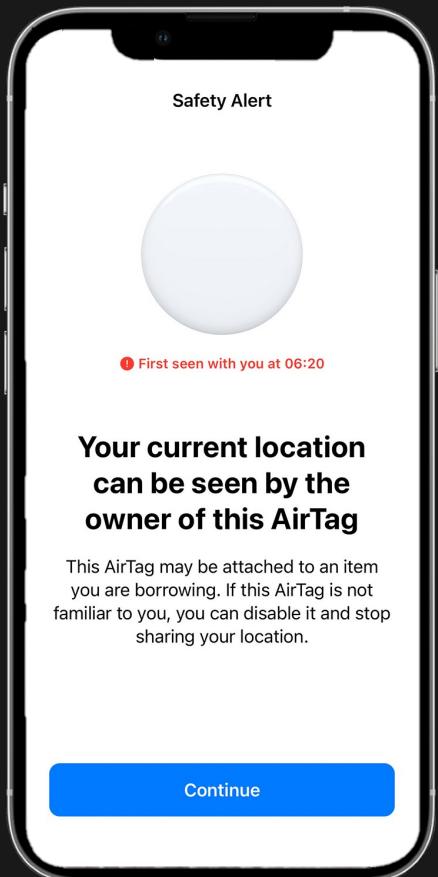
Avg. **9.8** feet  
At single family house

**229.8** feet  
During a flight from DC to LAX  
at 536.243 mph

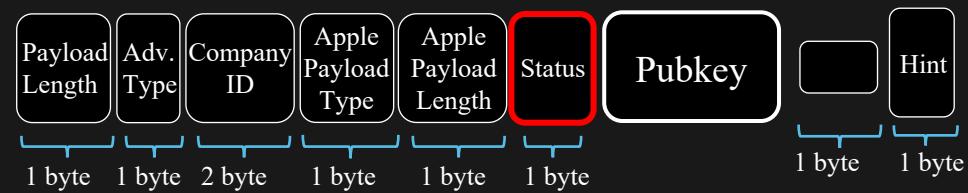


# Apple Anti-Stalking Strategy

iPhone user will receive notification when being tracked



**BLE Payload**  
31 bytes



*Mayberry et al.* revealed it can be bypassed!

# Stealth Enable

---



- Bluetooth can be disabled by system or user
- Linux, Android, Windows allow enabling Bluetooth quietly
- No user interaction or awareness

# Closing Remarks

---



## Security Adversary

Apple acknowledged our contribution and released patches in December 2024 to fix this vulnerability. Including,

iOS	18.2	tvOS	18.2
visionOS	2.2	macOS Ventura	13.7.2
iPadOS	17.7.3 & 18.2	macOS Sonoma	14.7.2
watchOS	11.2	macOS Sequoia	15.2

We speculate the vulnerability will take years to eventually fade out.

# Demo Time



The image shows two side-by-side browser windows. The left window is titled "nRootTag Control Center" and displays system status: Configured Replicas (200), Current State (running), and Runtime (10m). Below this is a "Search Management" section with a search bar containing "aaaaaa or 1eadbe112233", a green "Search Address" button, and a red "Clear All Tasks" button. A "Current Tasks" section below shows "No active tasks". At the bottom is a "Search Results" section with a loading message "0 results found". The right window is titled "TIME.IS" and shows the current time in Las Vegas, Nevada, United States: **12:00:51 am** on Friday, August 8, 2025. It also displays the International Cat Day event. Below the main time are time zone boxes for London (08:00am), Paris (09:00am), Beijing (03:00pm), Tokyo (04:00pm), and New York (03:00am). A large section at the bottom provides "Time zone info for Las Vegas" with the text **UTC -7** and **Pacific Time (PT)**.

localhost

nRootTag Control Center

Configured Replicas: 200  
Current State: running  
Runtime: 10m

TIME.IS

Time in **Las Vegas**, Nevada, United States now

**12:00:51 am**

Friday, August 8, 2025

Sun: ↑ 05:52AM ↓ 07:40PM (13h 48m) - More info - Make Las Vegas time default - Add to favorite locations

London 08:00am    Paris 09:00am    Beijing 03:00pm    Tokyo 04:00pm

New York 03:00am

Time zone info for **Las Vegas**

**UTC -7**

**Pacific Time (PT)**

# Closing Remarks

---



## Our Attack: nRootTag

- Turns a Bluetooth device into an "AirTag"
- Applies to Linux, Android and Windows devices
- **No Root Privileges are required**

# Closing Remarks

---



## Prerequisites

- Malicious App Installed
- BLE Capability
- Surrounded by FindMy Network

# Acknowledgments

---



- NSF and CCI fundings
- OpenHayStack, VanitySearch
- Apple Product Security
- GMU ORC