InGuardians™

APPLE AIRTAG DETECTION AND SIMULATION

Larry Pesce

Director of Research & Development,

Principal Managing Consultant

July 2022

What are AirTags?

Apple developed "tracking" devices useful for placing with often misplaced objects

- Keys, kids, backpacks, pets
- Battery powered, water resistant

Connects to Apple's Find My network

- Distributed monitoring and location tracking
- No location data stored on device, triangulated through observation by other macOS/iOS devices

Anonymous and encrypted, privacy built in*

What is the purpose AirTags?

Useful for tracking the location of your stuff

Also useful for others to track the location of….you.

Global coverage with iOS devices, other systems lacking

Device Identifier rotation, short observation period

- Tracking still possible for up to 24 hours

Privacy Issues

Affix a registered tag to victim

- Surreptitiously drop into bag, pocket
- Affix to underside of a vehicle
- Insert other evil action here…

No built-in detection availability with the Android ecosystem

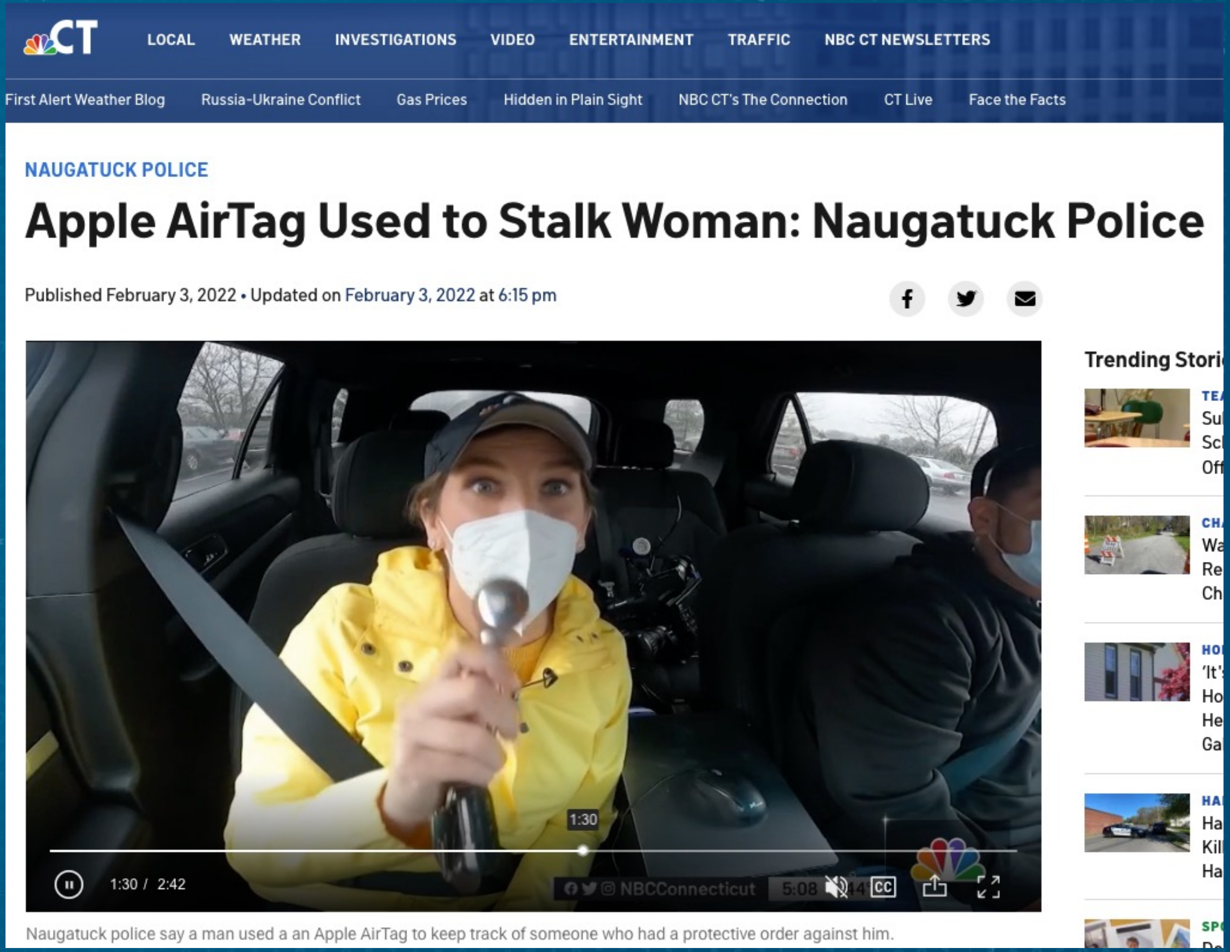
- “Tracker Detect” app from the Google Play Store, others
- Not ”always on” and detection inconsistent as a result

> *"We take customer safety very seriously and are committed to AirTag’s privacy and security. AirTag is designed with a set of proactive features to discourage unwanted tracking — a first in the industry — that both inform users if an unknown AirTag might be with them, and deter bad actors from using an AirTag for nefarious purposes. If users ever feel their safety is at risk, they are encouraged to contact local law enforcement who can work with Apple to provide any available information about the unknown AirTag.” - Apple*

Privacy Case Study

Reports of Airtags used in domestic stalking in Naugatuck, CT

Many, many other cases reported to LE in various scenarios



CT

LOCAL WEATHER INVESTIGATIONS VIDEO ENTERTAINMENT TRAFFIC NBC CT NEWSLETTERS

First Alert Weather Blog Russia-Ukraine Conflict Gas Prices Hidden in Plain Sight NBC CT's The Connection CT Live Face the Facts

NAUGATUCK POLICE

Apple AirTag Used to Stalk Woman: Naugatuck Police

Published February 3, 2022 • Updated on February 3, 2022 at 6:15 pm

Naugatuck police say a man used a an Apple AirTag to keep track of someone who had a protective order against him.

https://www.nbcconnecticut.com/news/local/apple-airtag-used-to-stalk-woman-naugatuck-police/2709492/

AirTag Technical Det

Assembled from mostly commodity components,

- nRF52832 is used for BLE and NFC
- Apple’s U1 for UWB

Enters “lost mode” 3 days away from its owner’s device

Makes noise once every **6 hours** while in lost mode and movement is detected

BLE advertisement every 2 seconds when away

BLE connection interval of 1 second when near

What RF technology?

BLE (Bluetooth 5.1)

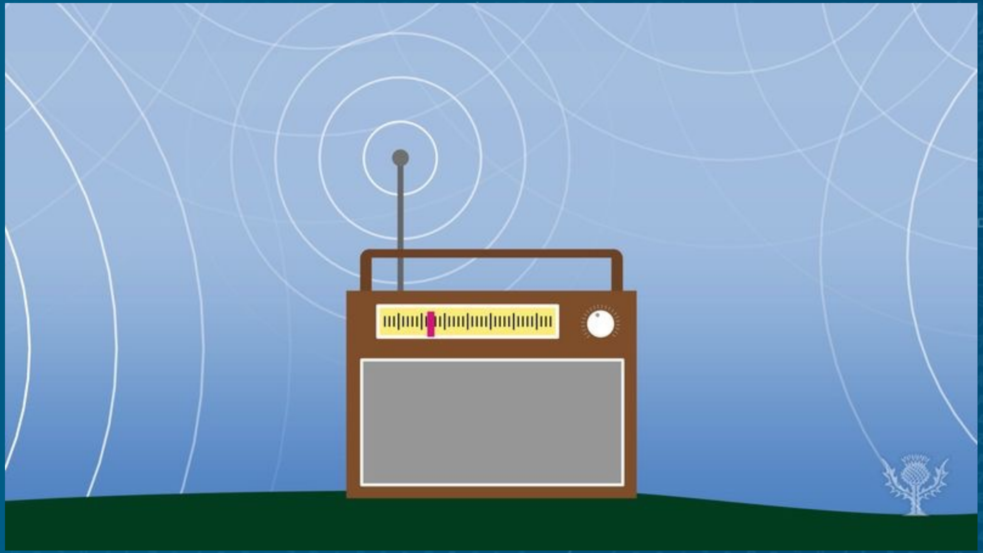
- Unregistered and registered advertisements
- Local notification of presence

NFC (Near Field Communication)

- Unregistered and registered URLs
- Tap for “lost mode” access URL to locate owner

UWB (Ultra Wide Band)

- Precision Finding, distance detection



NFC

Usable by anyone with a mobile device to scan an AirTag

NFC read and lead to owner ID, return of lost object

Read only MIFARE Plus with URL:

- Registered

```
https://found.apple.com/airtag?pid=5500&b=00&pt=004c&fv=00
100e10&dg=00&z=00&pi=793f8d9fccaa91c3c177f32acf47160656873
168d72f070cd925ce97
```

- Unregistered

```
https://found.apple.com/airtag?pid=5500&b=00&pt=004c&fv=00
100e10&dg=00&z=00&bt=A0B1C2D3E4F5&sr=ABCDEF123456&bp=0015
```

Parameter	Value	Description
pid	5500	AirTag Product ID?
b	00	Battery?
pt	004c	Precision Tracking version?
fv	00100e10	Firmware version?
dg	00	Diagnostic code?
z	00	Unknown
bt	XXXXXXXXXXXXXX	BDADDR (hex)
sr	XXXXXXXXXXXXXX	Serial Number (alpha)
bp	0015	Bluetooth version?
pi	Varies	Public Identity

UWB

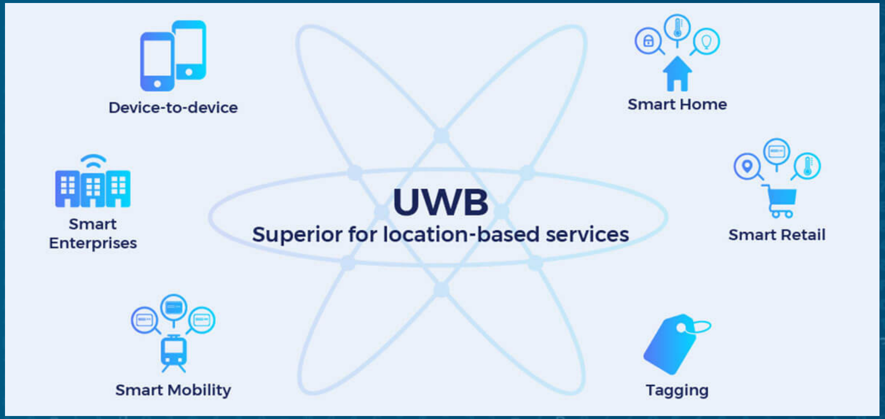
Implemented in Apple’s proprietary U1 chipset

- Channels 5 (6.5GHz) and 9 (8GHz), 500MHz wide
- BPSK modulation
- Single antenna

Used with Precision Finding for measuring distance (not direction)

- BLE (5.1) likely used for direction
  - BLE angle of arrival (AoA) and angle of departure (AoD)
  - BLE requires multiple antennas

Little known on the UWB implementation



BLE advertising

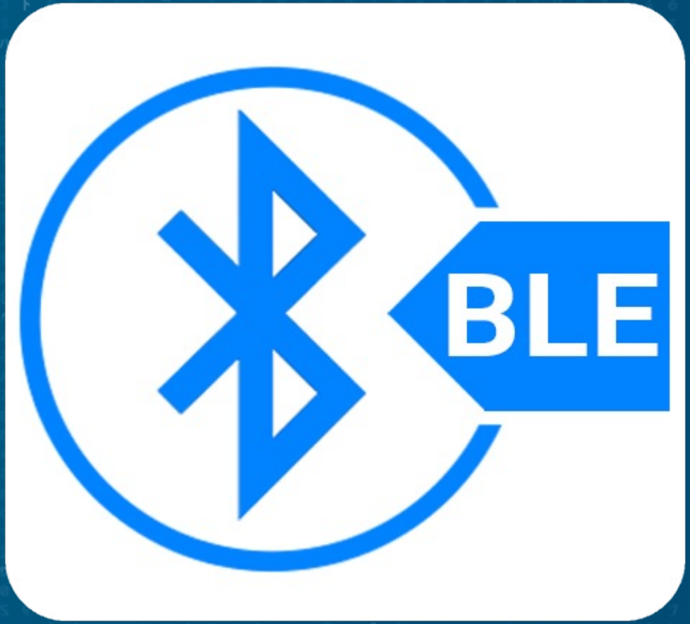
AirTags as BLE devices have a simplified channel hopping mechanism

- Simplified, as opposed to Bluetooth Classic

Frequency hopping spread spectrum (FHSS)

- 40 overall channels, 37 for data (if needed), 3 for advertisements
- Small 5khz wide channels, hops 1600 hops a second

Advertisements always available and continuous on all 3 advertising channels



Unregistered AirTag Advertisements

Specific static values for identification for unregistered (new) AirTags

33ms advertising interval

Adv Length 1 byte	Adv Type 1 byte	Company ID 2 bytes	Payload Type 1 byte	Payload Length 1 byte	Unknown 25 bytes
0x1E 31 bytes	0xFF Manufacturer specific	0X004C Apple's BLE ID	0x07 New FindMy broadcast	0x19 25 bytes	Varies

Registered AirTag Advertisements

Again, specific static values for AirTags registered to the FindMy network

- ECC P-224 Public key changes daily at 4AM, Counter (?) every 15 minutes

2000ms advertising interval

Some fields known; others static but unsure of actual functionality

Adv Length 1 byte	Adv Type 1 byte	Company ID 2 bytes	Payload Type 1 byte	Payload Length 1 byte	Status 1 byte	ECC P-224 Public Key 23 bytes	Upper 2 bits ECC key 1 byte	Counter? 1 byte
0x1E 31 bytes	0xFF Manufacturer specific	0X004C Apple's BLE ID	0x12 FindMy broadcast	0x19 25 bytes	0x10	Varies	0 - 3	Random

Overall AirTag Detection

Detection is trivial with iOS with built in tools

- Makes sense given the ecosystem…

Android apps exist, but are not real time

- Apps need to be open and at least in the background
- Need to remember to start the app, affects battery

Linux detection limited at best

- No apps, but can be cobbled together with BlueZ utilities

AirTag detection under Linux

Simple checks using *hcidump* and comparing the output to the following BLE beacon format (in regex format)

- Registered AirTag

```
^04\ 3E\ 2B\ 02\ 01\ .{26}\ 1E\ FF\ 4C\ 00\ 12\ 19
```

- Unregistered AirTag

```
^04\ 3E\ 2B\ 02\ 01\ .{26}\ 1E\ FF\ 4C\ 00\ 07\ 19
```

Automate with *AirTag-scan.sh*

```
# sudo ./AirTag-scan.sh
Registered AirTag Found!
04 3E 2B 02 01 00 01 E3 9A FE E5 9E C6 1F 1E FF 4C 00 07 19
05 00 55 10 00 00 01 6C 6E 0A 91 F9 B7 20 46 36 86 DA 43 6A
A1 F8 64 B7 80 E2
Unregistered AirTag Found!
04 3E 2B 02 01 00 01 CA 53 1F 56 0C C0 1F 1E FF 4C 00 12 19
10 CA 01 C1 53 C4 0E E2 A1 84 04 45 94 4E 31 0B 3A 55 E9 F3
A7 67 B3 02 09 C9
^C
```

AirTag Simulation

hciconfig for up/down and LE enable, *hcitool* to set BD_ADDR (*cmd 0x3f*) and send specific AirTag commands (*cmd 0x08*)

```
# sudo hcitool -i hci0 cmd 0x3f 0x001 0xec 0x3d 0xe2 0x92 0x52 0xd1
# sudo hcitool -i hci0 cmd 0x08 0x0008 1f 1e ff 4c 00 12 19 00 67 25 ac c7 7b 7e 7a 5c b8 b8 05 05 19
a2 f6 74 8a 5a c7 a0 0b 89 01 00
# sudo hcitool -i hci0 cmd 0x08 0x0006 d0 07 d0 07 03 00 00 00 00 00 00 00 00 07 00
# sudo hcitool -i hci0 cmd 0x08 0x000a 01
```

cmd 0x08 0x0008 sets the payload

cmd 0x08 0x0006 sets the advertising interval

cmd 0x08 0x000a starts the advertisement

AirTag key 515292e23dec6725acc77b7e7a5cb8b8050519a2f6748a5ac7a00b89

AirTag BD_ADDR d15292e23dec

AirTag payload 1eff4c001219006725acc77b7e7a5cb8b8050519a2f6748a5ac7a00b890100

Automate with Airtag-create.sh*

```
# sudo ./AirTag-create.sh {-u,-r}
```

Going Further

OpenHaystack!

- FOSS implementation from seemoo-labs to add your own stuff to Apple's Find My network
- BBC Micro:bit, ESP32, nRF51, linux HCI
- https://github.com/seemoo-lab/openhaystack

Making *actual* clones with a voltage glitching attack, also from seemoo-labs

- Glitch to recover firmware and restore it to a new AirTag 800 miles away
- https://raw.githubusercontent.com/seemoo-lab/airtag/main/woot22-paper.pdf

Wrapping up

Yay, new tools! (Look Ma, I did it!)

https://github.com/haxorthematrix/AirTag-tools

Pull requests HAPPILY accepted

- jwright approach to tool dev on this one!

Thanks!

@haxorthematrix (Twitter, IG, *all the things)

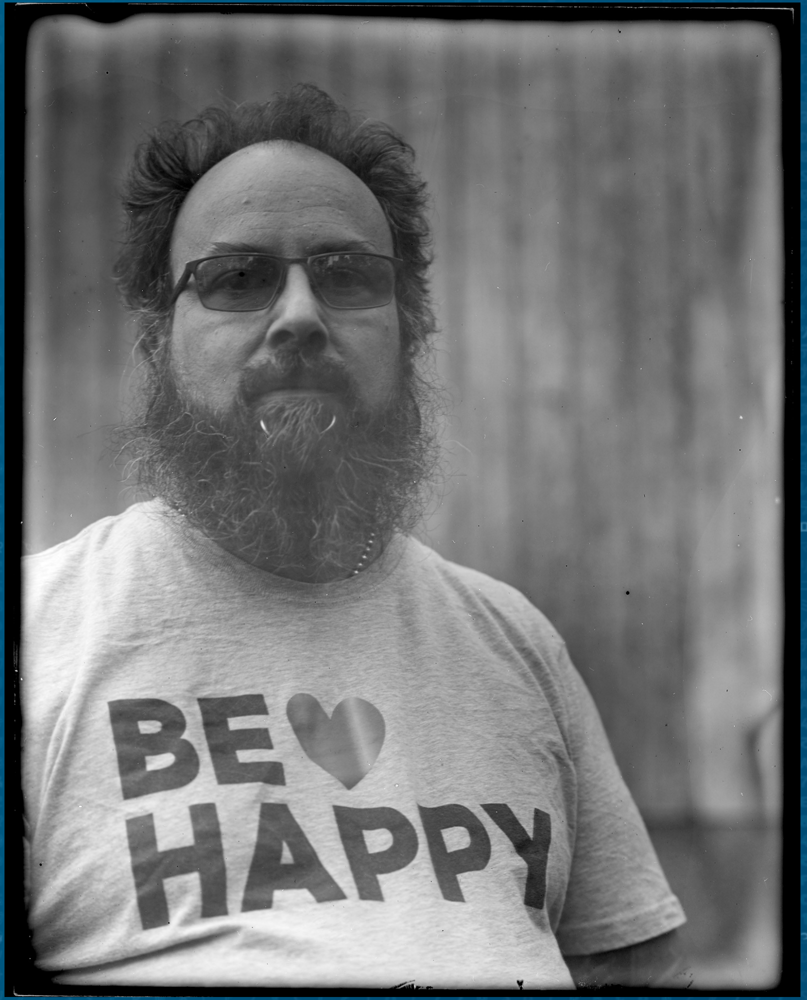


Photo Credit: Joshua Wright

@joswr1ght